

According to Smart Syllabus 2020
Accelerated Learning Programme (ALP)

F.Sc
12

AZEEM

10 BOARDS

SOLVED PAST PAPERS SERIES

COMPUTER SCIENCE

LAHORE

GUJRANWALA

MULTAN

FAISALABAD

RAWALPINDI

BAHAWALPUR

SARGODHA

D.G. KHAN

SAHIWAL

A.J.K.



According to Smart Syllabus 2020
Accelerated Learning Programme (ALP)

Azeem SOLVED PAST PAPERS
**COMPUTER
SCIENCE** **I.Cs.
Part-II**
2012 - 2019

*Lahore, Gujranwala, Rawalpindi, Faisalabad, Sargodha, Multan
Bahawalpur, Sahiwal, D.G. Khan and Mirpur (AJK) Boards*

- ☞ MCQs with solutions from exercises of PTBB.
- ☞ Short Questions with solutions from exercises of PTBB.
- ☞ Chapter wise MCQs with solutions from Past Papers (2014 - 2019) of all Boards of Punjab.
- ☞ Chapter wise Long questions from Past Papers (2014 - 2019) of all Boards of Punjab.
- ☞ Chapter wise Short questions with solutions from Past Papers (2014 - 2019) of all Boards of Punjab.

Azeem Academy
PUBLISHERS & BOOKSELLERS

22-Urdu Bazar Lahore.

Ph: 042-37231448

042-37232129

Website: www.azeemgroups.com E-mail: info@azeemgroups.com

CONTENTS

Ch.#	SUBJECT	Pg.#
Ch. 1	Data Basics	01
Ch. 2	Basic Concept & Terminology of Data Base	10
Ch. 3	Database Design Process	16
Ch. 4	Data Integrity and Normalization	26
Ch. 5	Introduction to Microsoft Access	35
Ch. 6	Table and Query	38
Ch. 7	Microsoft Access Forms and Reports	44
Ch. 8	Getting Started with C	47
Ch. 9	Elements of C	59
Ch. 10	Input/Output	77
Ch. 11	Decision Constructions	110
Ch. 12	Loop Constructs	136
Ch. 13	Functions	152
Ch. 14	File Handling in C	159

DATA BASICS

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

1. Which of the following represents a collection of concepts that are used to describe the structure of a database? (FSD-14)/ (LHR-14)
a) data warehouse b) data model
c) data structure d) data type
2. Which of following data model is more flexible? (SGD-14)(FSD-19)
(a) network data model (b) hierarchical data model
(c) relational data model (d) Object data model
3. Which of the following type of file require largest processing time?
(a) sequential file (b) random file
(c) Indexed sequential file (d) direct access file
4. Which of the following may be a temporary file?
(a) Master file (b) transaction file
(c) backup file (d) none of these

From Punjab Board:

1. A collection of related fields is called: (AJK-19)
(a) Character (b) Record
(c) Database (d) File
2. A type of file that contains data is called: (SGD-19)
(a) data file (b) Program file
(c) image file (d) Query file
3. Which of the following is also known as data set: (SWL-19)
(a) record (b) field
(c) file (d) module
4. The column of a table correspond to _____ (GRW-19) (D.G.K – 19)
(a) table (b) record
(c) attribute / Field (d) cell
5. Data that causes inconsistency lack in: (D.G.K – 19)
(a) Good data (b) Data redundancy
(c) Data integrity (d) Data anomaly

6. The process of arranging data in a logical sequence is called. (BWP-15) (SWL-18)
(a) sorting (b) summarizing
(c) data capturing (d) classifying
7. A set of related files created and managed by a DBMS is called. (SGD-18)
(a) Field (b) Record
(c) Database (d) Module
8. A collection of raw facts and figures is called. (SWL/G1-15) (MTN/G1-15)
(a) Data (b) Information
(c) Processing (d) None
9. Storage and retrieval of data is related to: (RWP/G1-15)
(a) Data capturing (b) Data manipulation
(c) Managing (d) Data finding
10. A set of related characters that represents a unit of data is called: (AJK/G1-14)
(a) Record (b) File
(c) Field (d) None key
11. Which of the following database model is also referred as inverted tree; (AJK-17)
(a) Hierarchal (b) Network
(c) Relational (d) Object
12. Data can be recovered in case of loss by using: (D.G.K-14)
(a) Master File (b) Transaction File
(c) Backup File (d) Data File
13. SQL is used for: (FSD-14)/ (LHR-14)/ (SWL-14)
(a) Data definition (b) Data manipulation
(c) Data definition & manipulation (d) Searching records
14. SQL stands for: (SWL-14) (AJK/G1-15) (LHR-19) (MTN-19) (GRW-19)
(a) Structured query language (b) Sort query language
(c) Self query language (d) Seek query language
15. Data base development process involves mapping of conceptual data model into: (SGD-14)
(a) Object oriented data model (b) Network data model
(c) Implementation model (d) Hierarchical model

SECTION II

SHORT QUESTIONS ANSWERS

From PTB Exercise:

Q1. Define file, record and field in details?

Ans. Records are composed of fields, each of which contains one item of information. A set of records constitutes a file. For example, a personnel file might contain records that have three fields: a name field, an address field, and a phone number field. In relational database management systems, records are called tuples.

Q2. Describe the file types from usage point of view and functional point of view?

(FSD-14), (FSD/G1-15) (RWP-19)

Ans. File Types (Usage Point of View):

Master File:- These file are latest updated files and never become empty. When the information in records is changed it is updated in master files.

Transaction File:- In these files data is kept before processing. These can be temporary files. Data in these files retained till the master files is updated.

Backup File:- These are permanent like master files. These files are created using some software utilities. These files are used to protect data. Used to store copy of actual data.

File Types (functional Point of View)

Program Files These files contain software instruction.

Example: Source program files and executable files. The extension of source and executable files are .com and .exe respectively.

Data File These files contain data. These files are created by the software being used.

Example:

Files having extensions like .dat, .doc, .xls, .mdb etc.

Software	File types
Word Processor	.doc, .rtf
Spread sheet	.xls, .wks
Database	.dat, .dbf, and .mdb
ASCII	.txt
Image files	.tif, .jpg, .eps, .gif, .bmp
Audio files	.wav, .mid
Video files	.avi, .mpeg

Q3. How do we organize the files on storage media?

Ans. File Organization (Storage Point of View)

Sequential Files: -These files store data as it arrives one after another in the sequence. These files take more time to store data. The data stored in these files are accessed sequentially. If you want to get to the last line in a sequential file of 23,000 lines, will have to read the preceding 22,999 lines. The best reason for using sequential files is their degree of portability to other Program. The drawback to sequential files is that you only have sequential access to your data. You access one line at a time, starting with the first line.

Direct or Random Files: Record in this type of file is stores on a calculated address. Each record in accessed directly. In random files the data is stored exactly as it appears in memory, thus saving processing time. Random access files can be wasteful of disk space because space is allocated for the longest possible field in every record. For example, a 100- byte comment field forces every record to use an extra 100 bytes of disk space, even if only one in a thousand actually uses it. At the other extreme, it records are consistent in length, especially if they contain mostly numbers, random files can save space over the equivalent sequential form.

Example:

In a database of 23,000 alumni, a program can go straight to record number 11,663 or 22,709 without reading any of the other records. This capability makes it the only reasonable choice for large files and probably the better choice for small ones, especially those with relatively consistent record lengths.

Indexed Sequential Files: The data in this type of file can be accessed sequentially as well as randomly based on a key value. As records are stored in the form of key pointer pair in the index file, therefore, it requires more space on the disk as compare to random files. Its processing is as fast as random files.

Q4. In general, what activities are to be performed on the databases? Discuss in details?

Ans. DBMS Functions

- Data Dictionary Management.
- Data Storage Management.
- Data Transformation and Presentation.
- Security Management.
- Multiuser Access Control.
- Backup and Recovery Management.
- Data Integrity Management.
- Database Access Languages and Application Programming Interfaces.

Q5. What are the four major components of the database systems? Write in details.

Ans. Components of DBMS

- Software. This is the set of programs used to control and manage the overall database....
- 1. Hardware.
- 2. Data.
- 3. Procedures.
- 4. Database Access Language.
- 5. Query Processor.
- 6. Run Time Database Manager.
- 7. Data Manager.

Q6. Discuss the objectives of the databases in your own words?

Ans. Various Objectives of Database Management System

- Mass Storage. DBMS can store a lot of data in it.
- Removes Duplicity. If you have lots of data then data duplicity will occur for sure at any instance.
- Multiple Users Access. No one handles the whole database alone.
- Data Protection.
- Data Back up and recovery.
- Everyone can work on DBMS.
- Integrity.
- Platform Independent.

Q7. Describe the different database models. (FSD-19)

Ans. Types of database models

1. Hierarchical database model.
2. Relational model.
3. Network model.
4. Object-oriented database model.
5. Entity-relationship model.
6. Document model.
7. Entity-attribute-value model.
8. Star schema.

Q8. Discuss the objectives and features of the dbms.

Ans. The functions of a DBMS include concurrency, security, backup and recovery, integrity and data descriptions. Database management systems provide a number of key benefits but can be costly and time-consuming to implement.

Q9. What are the advantages and disadvantages of the dbms?

Ans. Advantages of Database Management System (DBMS)

1. Better data sharing
2. Better data security
3. Improved data integration
4. Minimized data inconsistency
5. Improved decision making
6. Increased end-user productivity

Disadvantage of Database Management System (DBMS)

1. Increased costs
2. Management intricacy
3. Maintaining currency
4. Frequent upgrade/replacement cycles

From Punjab Board:

1. Define attribute with example. (RWP-19)

Ans. In a database management system (DBMS), an **attribute** may describe a component of the database, such as a table or a field, or may be used itself as another term for a field.

2. Distinguish between DMBS and data base. (SWL-19)

Ans. The database contains data in a well-structured format. DBMS is a kind of software that helps you to retrieve, edit and store structured data in the database. In the case of a relational database, related data are stored in multiple tables. And each table has data stored in rows and columns.

3. Write the purpose of index file? (AJK-17), (AJK/G1-16) (AJK-19)

Ans. An **indexed file** is a computer file with an **index** that allows easy random access to any record given its file key. The key must be such that it uniquely identifies a record. If more than one **index** is present the other ones are called alternate **indexes**. The **indexes** are created with the file and maintained by the system.

4. List different objectives of Databases. (BWP-14)

Ans. Various Objectives of Database Management System:

- Mass Storage. DBMS can store a lot of data in it.
- Removes Duplicity. If you have lots of data then data duplicity will occur for sure at any instance.
- Multiple Users Access. No one handles the whole database alone.
- Data Protection.
- Data Back up and recovery.
- Everyone can work on DBMS.
- Integrity.
- Platform Independent.

5. Define data processing? (AJK/G1-14), (D.G.K/G1-15), (D.G.K-14), (LHR-14)

Ans. The carrying out of operations on data: especially by a computer, to retrieve, transform, or classify information. **Data processing**. Manipulation of **data** by a **computer**. It includes the conversion of raw **data** to machine-readable form, flow of **data** through the CPU and memory to output devices, and formatting or transformation of output. Any use of **computers** to perform defined operations on **data** can be included under **data processing**.

6. What are program files? (AJK/G1-14) (FSD-14)

Ans. The files that contain the set of instructions in machine code to perform various tasks on data are known as program files. These are executable files. These files have the file extensions exe. or com.

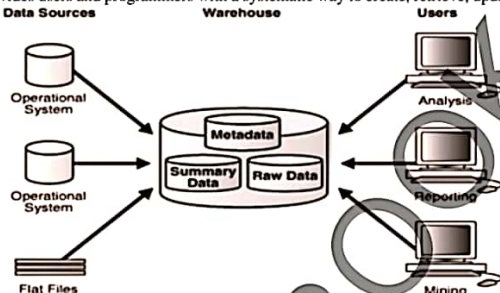
(OR) When used as a singular noun, a **program file** is a description of a file that stores the code to run a **program**. This type of file is commonly referred to as an **executable file**. When referring to more than one **file**, **program files** are all of the different files that allow a software program to run on your computer. Most programs have different types of files (e.g., **DLL** files, data files, etc.) that are used in conjunction with the executable file.

7. Define data dictionary? (AJK/G1-14) (SWL-19)(D.G.K – 19)

Ans. A set of information describing the contents, format, and structure of a database and the relationship between its elements, used to control access to and manipulation of the database. A **data dictionary** is a file or a set of files that contains a database's metadata. The **data dictionary** contains records about other objects in the database, such as **data** ownership, **data** relationships to other objects, and other **data**.

8. Define DBMS.

Ans. A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data.



9. What is a file?

Ans. We can think of a traditional database as an electronic filing system, organized by fields, records, and files. A field is a single piece of information; a record is one complete set of fields; and a file is a collection of records. For example, a telephone book is analogous to a file.

10. Define the term DDL (Data definition language).

Ans. **Data Definition language (DDL)** in DBMS with Examples: Data Definition Language can be defined as a standard for commands through which data structures are defined. It is a computer language that is used for creating and modifying the structure of the database objects, such as schemas, tables, views, indexes, etc.

11. State the purpose of backup and recovery.

Ans. This feature is available in almost all the DBMS software. By using this feature, you can take the backup of your important data. In case of data failure or loss of data, the data can be recovered by using the backup copy of the data.

12. Describe consistency constraints.

Ans. **Constraints** are the rules enforced on the data columns of a table. These are used to limit the type of data that can go into a table. This ensures the accuracy and reliability of the data in the database. **Constraints** could be either on a column level or a table level.

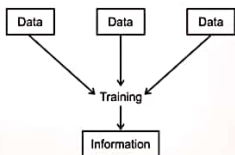
Simply we can say **Consistency** in database systems refers to the requirement that any given database transaction must change affected data only in allowed ways. Any data written to the database must be valid according to all defined rules, including **constraints**, cascades, triggers, and any combination thereof.

13. Differentiate between data and information.

Ans. The major difference between data and information is that data is raw material that is to be processed and information is the processed data. Data: ... It is unorganized data or facts that are to be processed. Data is plain fact and it has to be processed for further information.

Differences in Meaning: Data are simply facts or figures—bits of information, but not information itself. When data are processed, interpreted, organized, structured or presented so as to make them meaningful or useful, they are called information. Information provides context for data.

Information is created from data



14. Why is report generator used in database system?

(RWP-17)(MTN-19)

Ans. A report generator is a computer program whose purpose is to take data from a source such as a database, XML stream or a spreadsheet, and use it to produce a document in a format which satisfies a particular human readership. A database report is the formatted result of database queries and contains useful data for decision-making and analysis. Most good business applications contain a built-in reporting tool; this is simply a front-end interface that calls or runs back-end database queries that are formatted for easy application usage.

15. What is meant by reproduction?

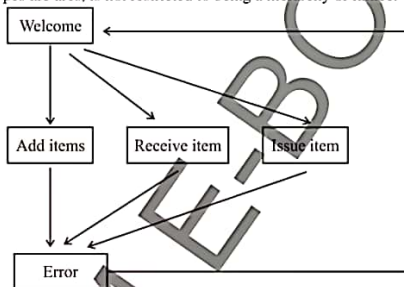
(SGD-14)

Ans. The process of making duplicate copies of output results (or data) is called reproduction. Usually, the output result is reproduced to distribute it to different persons or users.

16. Describe network model.

(RWP-18)

Ans. The network model is a database model conceived as a flexible way of representing objects and their relationships. Its distinguishing feature is that the schema, viewed as a graph in which object types are nodes and relationship types are arcs, is not restricted to being a hierarchy or lattice.

**17. Enlist different types of database models.**

(MTN/G1-15), (LHR-14)

Ans. There are four structural types of database management systems:

- Hierarchical databases.
- Network databases.
- Relational databases.
- Object-oriented databases.

18. Differentiate between data redundancy and data inconsistency.

(SGD-19) (RWP-19)(D.G.K – 19) (FSD-19)

Ans. The main difference between data redundancy and data inconsistency is that data redundancy is a condition that occurs when the same piece of data exists in multiple places in the database whereas data inconsistency is a condition that occurs when the same data exists in different formats in multiple tables.

19. What is meant by data inconsistency? (AJK/G1-16) (SGD-19) (RWP-19) (D.G.K – 19) (FSD-19)

Ans. Data inconsistency means that different files may contain different information of a particular object or person. Actually redundancy leads to inconsistency. When the same data is stored at multiple locations, the inconsistency may occur.

20. Define database.

(GRW-14) (SWL-14)

Ans. A collection of logically related data is called database. For example, the telephone directory, your personal address book, dictionary etc. are example of database. A Microsoft Excel spreadsheet or Microsoft Access are good examples of desktop database programs. These programs allow users to enter data, store it, protect it, and retrieve it when needed. They include databases like SQL Server, Oracle Database, Sybase, Informix, and MySQL.

21. Define data. Give example.

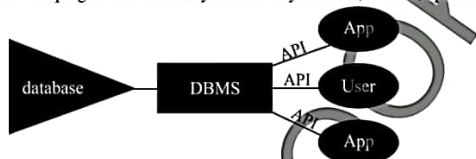
(D.G.K/G1-15)

Ans. Information in raw or unorganized form (such as alphabets, numbers, or symbols) that refer to, or represent, conditions, ideas, or objects. **Data** is limitless and present everywhere in the universe. ... Computers: Symbols or signals that are input, stored, and processed by a computer, for output as usable information.

22. Define database system.

(LHR/G1-15)

Ans. A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data



23. What is the main purpose of database system? (MTN-19)

Ans. The purpose of a database is to store and retrieve information in a way that is accurate and effective. The purpose for a database management system is to provide a system to manage the different databases it contains (performance, security, availability, etc.)

24. Describe data manipulation. (GRW-14), (D.G.K/G1-15), (BWP-14)

Ans. The process of applying different operation on the collected data to achieve the required result is called data manipulation. Data manipulation is the process of changing data to make it easier to read or be more organized. For example, a log of data could be organized in alphabetical order, making individual entries easier to locate.

25. Describe activities in data processing.

(D.G.K-14), (LHR/G1-15)

Ans. Data processing is simply the conversion of raw data to meaningful information through a process. The process includes activities like data entry, summary, calculation, storage, etc. Useful and informative output is presented in various appropriate forms such as diagrams, reports, graphics, etc

26. Define Sorting.

(D.G.K-14)

Ans. The fields of a database can be sorted or rearranged in a number of ways: Ascending order - Arrangement of data from lowest to highest in sequence. Multiple-field sorting - Sorting on two or more fields of information, one primary and the others secondary. (OR)

The process of arranging data or records of a table in a specific order i.e., in alphabetical or numeric order is called data sorting. In a table, data is sorted based on a data in a specific field (or group of field).

27. Define file organization & Name different types of file organization. (SGD-14), (FSD-14), (SWL/G1-15)

Ans. File organization is a way of organizing the data or records in a file. There are several types of file organization, the most common of them are sequential, relative (direct) and indexed.

28. Differentiate between parent table and child table.

(SGD-14)

Ans. A parent is the table that stores the primary key, A child is any table that references the parent with a foreign key. We name these tables parents and children because the child inherits values from the parent (just like children in real life).

29. Who is an end user?

(SGD-14) (GRW-14), (FSD-14)

Ans. An end-user is the person who actually uses a product or service. This often differs from the customer, defined as the entity that purchases a product or service from the perspective of the seller.

The following are illustrative examples of an end-user.

1. An architecture firm builds a hotel for a hotel company. The end-users are the guests who stay in the hotel.
2. Parents buy breakfast cereal and kids eat it.
3. A firm purchases business software and its employees use it. In this case, individual employees are end-users.

- 30. Differentiate between master file and transaction file. (LHR-18),(SGD-14) (GRW-19)**
- Ans.** A collection of transaction records. The data in transaction files is used to update the master files, which contain the data about the subjects of the organization (customers, employees, vendors, etc.). See master file for examples of typical master records master file is never empty, it is always updated. Before updating or processing the master file is called transaction file.
- 31. What is direct Access File and Sequential Access File? (BWP-15)**
- Ans.** "Sequential access must begin at the beginning and access each element in order, one after the other. Direct access allows the access of any element directly by locating it by its index number or address. Arrays allow direct access. Magnetic tape has only sequential access, but CDs had direct access.
- 32. List the file type from usage point of view. (FSD-14), (FSD/G1-15) (RWP-19)**
- Ans. File Types (Usage Point of View):**
- Master File**
- These file are latest updated files and never become empty.
 - When the information in records is changed it is updated in master files.
- Transaction File**
- In these files data is kept before processing.
 - These can be temporary files.
 - Data in these files retained till the master files is updated.
- Backup File**
- These are permanent like master files.
 - These files are created using some software utilities.
 - These files are used to protect data.
- 33. What is extension of database file in MS ACCESS? (MTN/G1-15)**
- Ans.** Microsoft Access versions from Access 1.0 (1992) through Access 2003, used the .mdb extension for all database files. However, the underlying MDB format changed significantly over the years. With Access 2007, Microsoft introduced a new default file format, using the .accdb extension.

SECTION III

LONG QUESTIONS

1. Briefly describe the four advantages and four disadvantages of database management system (DBMS).
(LHR/G1-16), (D.G.K/G1-16) (MTN-19) (D.G.K - 19) (SGD-19)
2. What is database? Explain three types of database models. (SGD/G1-16) (FSD-19)
3. Define file. Briefly describe the file types from usage point of view. (BWP/G1-16)(RWP-19)
4. What is database system? Explain any three major components of database systems.
(KSMR/G1-16), (MTN/G1-16), (AJK/G1-16) (SWL-19)



BASIC CONCEPT & TERMINOLOGY OF DATA BASE

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

1. Insert command is used to insert: (SWL/G1-15) (RWP/G1-15) (LHR/G1-15) (FSD-19) (MTN-19)
a) a new table
b) a new record
c) a view
d) dependencies
2. CREATE command is used to create a:
a) table
b) view
c) report
d) query
3. SQL is used for:
a) data definition
b) data manipulation
c) data definition and manipulation
d) searching records
4. The foreign key is found in:
a) parent table
b) dependent table
c) pivot table
d) index table
5. A table /Relation Must have : (SWL-14), (SGD-14) (MTN-19) (SWL-19) (LHR-19)
a) primary key
b) secondary key
c) composite key
d) sort key

From Punjab Board:

1. A virtual table that is constructed from other tables is called: (RWP-19)
(a) view (b) Table (c) Relation (d) Tuple
2. Which of the following is correct association? (SWL-19)
(a) file = column (b) record = row
(c) field = row (d) record = table
3. Which of the following can be a primary key: (LHR-19)
(a) Last name (b) Salary
(c) Customer ID (d) Region
4. A logical grouping of characters is a :- (MTN-18) (RWP-19)
(a) Field (b) Record
(c) File (d) Table
5. Create command is used to create. (D.G.K/G1-15)
(a) Table (b) View
(c) Report (d) Query

5. In a relational database, a single piece of information is called:- (MTN-18)
(a) Table (b) Record
(c) Entity (d) Attribute
6. An attribute is also known as a. (SGD-18) (AJK-19) (AJK/G1-15) (GRW-14)/(LHR-14)
(a) Table (b) Relation
(c) Row (d) Field
7. A primary key that consists of more than one attributes is called: (RWP-18)
(a) Foreign key (b) Composite key
(c) Multi valued key (d) Global key
8. Which of the following is used to associate entities with one another? (SWL-18)
(a) attribute (b) relationship
(c) identifier (d) object
9. The foreign key is found in (SWL-18)
(a) parent table (b) pivot table
(c) dependent table (d) index table
10. A row of a relation is called: (BWP-15)
(a) Attribute (b) Entity
(c) Tuple (d) Field
11. No. of primary keys can exist in a table: (MTN/G1-15)
(a) One (b) Two
(c) Three (d) Four
12. Which of the following keys does not hold uniqueness property? (SWL/G1-15)
(a) Candidate key (b) Foreign key
(c) Primary key (d) Secondary key
13. A row of a relation is called a(n): (RWP/G1-15)
(a) Attribute (b) Entity
(c) Tuple (d) Field
14. In MS Access, table doesn't contain: (SGD-18)
(a) Field (b) Record
(c) Character (d) File
16. Which of the following is also known as control key? (AJK/G1-14) (SGD-19)
(a) Primary key (b) Sort key
(c) Candidate key (d) Secondary key
17. Who serves as bridge between end user and data base administrator? (AJK/G1-14)
(a) Data administrator (b) Data base administrator
(c) Application (d) Analyst
18. The actual data of data base is stored in: (AJK/G1-14)
(a) Tables (b) Queries
(c) Forms (d) Reports
20. Color of car is an example of; (AJK/G1-16)
(a) Entity (b) Attribute
(c) Relation (d) Relationship
21. The row of a table is also called; (AJK/G1-16)
(a) Entity (b) Attribute
(c) Cell (d) Record

22. A table in which the foreign key is found is called: (BWP-14)/ (D.G.K-14)/ (FSD-14)/ (LHR-14)
(a) Parent Table (b) Dependent Table
(c) View (d) All these
23. A category of data or information that describes an entity is called a (n): (BWP-14)
(a) Attribute (b) Data Item
(c) Record (d) Tuple
24. How many primary keys can a relation have? (D.G.K-14)
(a) At least two (b) Only one
(c) No limit (d) Three
25. In relational database, a table is also called a: (SWL-18) (FSD-14)/ (GRW-14)/ (LHR-14)/ (SWL-14)
(a) Tuple (b) Relation
(c) File (d) Schema
26. In a relational database, a single piece of information is called. (FSD-14)/ (LHR-14)
(a) Field (b) Record
(c) Entity (d) Attribute
27. A primary key that consists of more than one attribute is called: (GRW-14)
(a) sort key (b) candidate key
(c) secondary key (d) composite key
28. A relation is analogous to: (LHR-14)/(RWP-14)
(a) Row (b) Field
(c) Record (d) File
29. A database containing all students in a class would store basic data of students in: (RWP-14)
(a) Record (b) Field
(c) Cell (d) File
30. A table / relation must have: (SWL-14), (SGD-14) (MTN-19) (SWL-19) (LHR-19)
(a) Primary key (b) Secondary key
(c) Composite key (d) Sort key

SECTION II

SHORT QUESTIONS ANSWERS

From PTB Exercise:

Q1. How the records and files are constructed in traditional file management system?

Ans. Traditional data recording technology is based on *file systems*. *Record* data follows certain storage format required by *file systems*, so it could be directly accessed in the operating *system*. However, while *file system* enables easy data access, it also limits *system* recording bandwidth.

Q2. How the data is stored and retrieved in fms(file management system)?

Ans. Data retrieval means obtaining data from a database management system such as ODBMS. In this case, it is considered that data is represented in a structured way, and there is no ambiguity in data. In order to retrieve the desired data the user present a set of criteria by a query.

Q3. How The Tables/Relations Are Formed Up In Dbms?

Ans. A *relation* is usually described as a *table*, which is organized into rows and columns. All the data referenced by an attribute are in the same domain and conform to the same constraints.

Q5. Write down the properties of relations in details?

Ans. Relational tables have six properties:

- Values are atomic.
- Column values are of the same kind.
- Each row is unique.
- The sequence of columns is insignificant.
- The sequence of rows is insignificant.
- Each column must have a unique name.

From Punjab Board:**1. What is key? (AJK/G1-16)**

Ans. A key is an attribute (field) or set of attribute of a relation. Keys are defined in the relation for the following purposes:

- For uniquely identifying the records of a relation.
- For establishing relationships between of a relation.
- For quickly accessing particular records from the relation/ table.
- For sorting records of a relation based on the data of one or more columns.

List different types of keys used in relations.

Different types used in relations are: Primary key, secondary key, candidate key, alternate key, composite key, sort key and foreign key.

2. Define candidate key? (AJK/G1-14)

Ans. A **candidate key** is a column, or set of columns, in a table that can uniquely identify any database record without referring to any other data. Each table may have one or more **candidate keys**, but one **candidate key** is unique, and it is called the **primary key**.

3. What are the properties of a relation? (AJK/G1-14), (RWP/G1-15), (AJK/G1-16), (BWP-15) (SWL-14)

Ans. A relational database model is based on the concept of a relation. By definition, a relation must have certain properties or characteristics. These properties are discussed below:

Relational tables properties:

1. Values are atomic.
2. Column values are of the same kind.
3. Each row is unique.
4. The sequence of columns is insignificant.
5. The sequence of rows is insignificant.
6. Each column must have a unique name.
7. Each Column Has a Unique Name

4. Define the term table or relation. (FSD-14) (SWL-14) (D.G.K/G1-15) (AJK/G1-15) (MTN-19)

Ans. A **table** is a collection of related data held in a structured format within a database. It consists of columns, and rows. ... "**Table**" is another term for "**relation**"; although there is the difference in that a **table** is usually a multiset (bag) of rows where a **relation** is a set and does not allow duplicates.

5. List different types of keys. (D.G.K-14) (SWL/G1-15) (LHR-19)

Ans.

1. **Super Key.** Super key is a set of one or more than one keys that can be used to identify a record uniquely in a table. Example: Primary key, Unique key, Alternate key are subset of Super Keys.
2. **Candidate Key.**
3. **Primary Key.**
4. **Alternate key.**
5. **Composite/Compound Key.**
6. **Unique Key.**
7. **Foreign Key.**

6. Define an attribute. Give an example. (SWL-14) (LHR/G1-15) (GRW-14) (BWP-15)

Ans. An **attribute** is an element that takes a value and is associated with an object, such as an item, a region, a page. In general, an **attribute** is a characteristic. In a database management system (DBMS), an **attribute** refers to a database component, such as a table. It also may refer to a database field. **Attributes** describe the instances in the row of a database.

7. What is the different between Record and Field? (BWP-15) (MTN/G1-15)

Ans. Data is stored in records. A **record** is composed of **fields** and contains all the data about one particular person, company, or item in a database. In this database, a **record** contains the data for one customer support incident report. **Records** appear as rows in the database table.

8. Write two field types used in MS-Access. (BWP-15)

Ans. The **Text data type** is used for short text **fields** of limited length, and can contain up to 255 characters. **Memo.** A **Memo field** is a virtually unlimited text **field**. It can store up to 1GB of text. Microsoft Access supports text in a variety of languages and alphabets.

Fields in a Microsoft Access database can be of many different data types. Access supports different kinds of text, numbers, dates and some more special data types. This document will describe each data type and note which versions of Microsoft Access support it.

Textual types (AJK-19)

Text

The Text data type is used for short text fields of limited length, and can contain up to 255 characters.

Memo

A Memo field is a virtually unlimited text field. It can store up to 1GB of text.

Microsoft Access supports text in a variety of languages and alphabets. Access 97 supported multiple code pages in the same document. This worked great in most cases, but caused problems when columns contained text in different alphabets. Therefore Microsoft switched to the Unicode character encoding in Access 2000. MDB Viewer automatically supports all encodings used in Access.

Numeric Types

Byte, Integer and Long Integer

Decimal

Currency

Float and Double

Other Types

Date/Time

Yes/No

OLE Object

Complex Types

Multi-Valued Field

Attachments

Memo version history

9. What is the use of text data type? (AJK-19)

Ans. Text or combinations of text and numbers stored as text and used as a hyperlink address. Up to 8,192 (each part of a Hyperlink data type can contain up to 2048 characters). The Lookup Wizard entry in the Data Type column in the Design view is not actually a data type



10. Distinguish between primary key and foreign key.

(LHR/G1-15), (LHR-14), (RWP-14) (GRW-14) (RWP/G1-15) (BWP-14) (SWL-19)

Ans.

Primary Key	Foreign Key
We can have only one Primary key in a table.	We can have more than one foreign key in a table
By default, Primary key is clustered index and data in the database table is physically organized in the sequence of clustered index.	Foreign key do not automatically create an index, clustered or non-clustered. You can manually create an index on foreign key.
Primary Key can't accept null values.	Foreign key can accept multiple null value.
Primary key uniquely identify a record in the table.	Foreign key is a field in the table that is primary key in another table.

11. Differentiate between candidate key and alternate key.

(FSD/G1-15) (BWP-14)

Ans. I'll take example of an Employee table:

Employee (Employee ID, FullName, SSN, Dept ID):

1. **Candidate Key:** are individual columns in a table that qualifies for uniqueness of all the rows. Here in Employee table **EmployeeID** & **SSN** are Candidate keys.

2. **Primary Key:** is the columns you choose to maintain uniqueness in a table. Here in Employee table you can choose either **EmployeeID** or **SSN** columns, **EmployeeID** is preferable choice, as **SSN** is a secure value.

3. **Alternate Key:** Candidate column other than the Primary column, like if **EmployeeID** is PK then **SSN** would be the Alternate key.

12. Differentiate between primary key and secondary key.

(SWL/G1-15) (MTN/G1-15) (D.G.K/G1-15) (D.G.K-14) (SGD-14)

Ans. The difference between Primary Key and Secondary Key are stated below:-**Primary Key:**

- 1) Is used for Unique Identification of Rows
- 2) We have only one Primary Key per table
- 3) A Primary Key is one of the Candidate Keys or you can say one of the irreducible super key, depends on database designer which one he needs.

Secondary Key:

- 1) Is used for Identification of Rows but not usually Unique
- 2) We can have multiple Secondary Key per table
- 3) Attribute used for Secondary key are not the ones used for Super Key i.e. Secondary Key is not even be one of the Super Key.

13. Identify name of entity and primary key in the following:

(LHR-18)

Ans. **STUDENT** (Student ID, St name, Group)**Entity name = STUDENT****primary key = Student ID****14. Define key / field.**

(RWP-14) (RWP-14) (BWP-15) (AJK/G1-15)

Ans. A **field** in a record that holds unique data which identifies that record from all the other records in the file or database. Account number, product code and customer name are typical **key fields**. As an identifier, each **key value** must be unique in each record.**15. Distinguish between data dependence and data independence.**

(RWP-14) (BWP-15)

(SGD-19)

Ans. Data dependence mean the data is dependence upon application programme. When we change in data so also change in application program which we use

Data independence mean data is independent on application programme or vice versa. If we change in data so there is no change in application programme

SECTION III**LONG QUESTIONS****No long questions in this Chapter by Punjab Text Book Board.**

DATABASE DESIGN PROCESS

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

- Which of the following keys does not hold uniqueness property
 a) Candidate key b) foreign Key
 c) Primary Key d) Secondary Key
- An entity related to itself in an ERD model refers to:
 (RWP/G1-15)(BWP-14)(FSD-14)(LHR-14)(FSD-19)
 a) Recursive relationship b) One-to- relationship
 c) Many-to-many relationship d) One-to-one relationship
- Database development process involves mapping of conceptual data model into:
 a) Object oriented data model b) network data model
 c) Implementation Model d) Hierarchical data model
- In ERD model, the relationship between two entities is represented by a: (AJK-17)(AJK-19)
 a) Diamond Symbol b) Rectangular box
 c) Oval Symbol d) Line
- In hybrid distribution which kind of fragments is stored at only one site?
 a) Critical fragments b) Non-critical fragments
 c) Critical and non-critical fragments d) only large fragments

From Punjab Board:

- Graphical representation of a program is called: (D.G.K – 19)
 (a) Logical chart (b) Binary chart
 (c) Flow - chart (d) E - R chart
- All the hardware costs are considered during. (SWL-19)
 (a) project planning (b) requirement analysis
 (c) feasibility study (d) data analysis
- In an ER diagram, a diamond represents: (RWP-19)
 (a) Entity (b) Attribute
 (c) File (d) Relationship
- E-R diagram represents: (MTN-19)
 (a) Attribute (b) Relationship (c) Cardinals (d) Entity
- Which of the following is used to describe the characteristics of an object: (SGD-19) (LHR-19) (GRW-19)
 (a) Modality (b) Attribute
 (c) Cardinality (d) Entity
- In an E.R diagram, a rectangle represented. (D.G.K – 19)
 (a) Entity (b) Attribute
 (c) Relationship (d) Decision
- Merging the relation is also known as: (RWP-18)
 (a) view integration (b) view entitles
 (c) view tables (d) view relations

8. A database consists of various components called: (MTN/G1-15) (FSD-19) (BWP-15)
(a) Tools (b) Properties
(c) Entities (d) Objects
9. Which of the following is used to associate entities with one another: (LHR/G1-15)
(a) Entity (b) Attribute
(c) Identifier (d) Relationship
10. How many types of relationship can be used? (BWP-15)
(a) 1 (b) 2
(c) 3 (d) 4
11. An entity related to itself in an ERD model refers to:
(RWP/G1-15)(BWP-14)(FSD-14)(LHR-14) (FSD-19)
(a) Recursive relationship (b) One to many relationship
(c) Many to many relationship (d) One to one relationship
12. _____ types of relationship can be used. (MTN/G1-15)
(a) 2 (b) 3
(c) 4 (d) 5
13. Which is not included in the definition of an entity. (D.G.K/G1-15)
(a) Person (b) Object
(c) Concept (d) Action
14. A relational database, table is also called. (D.G.K/G1-15)
(a) Table (b) Relation
(c) File (d) Schema
15. A database consists of various components called. (D.G.K/G1-15)
(a) Tools (b) Properties
(c) Entities (d) Objects
16. How many types of relationship can be used. (D.G.K/G1-15)
(a) 2 (b) 3
(c) 4 (d) 5
17. A person's name, birthday and social security number are examples of: (MTN-18)
(a) Attributes (b) Entities
(c) Relationships (d) Descriptors
18. Which of the following is not a basic data distribution strategy? (AJK/G1-14)
(a) Centralized (b) Partitioned
(c) Replicated (d) Duplicated
19. The relationship between countries and their capitals is an example of: (AJK/G1-14)
(a) One-to-one (b) One-to-many
(c) Many-to-many (d) None
20. In ERD model, the relationship between two entities are represented by; (AJK-17) (AJK-19)
(a) Rectangle (b) Oval
(c) Square (d) Diamond
21. Why are validation rules used in a database: (BWP-14)
(a) Ensure correct data is entered (b) To avoid typing errors
(c) Both A and B (d) Neither A nor B
22. Which of the following is not a basic data distribution strategy? (D.G.K-14)
(a) Centralized (b) Partitioned
(c) Replicated (d) Duplicated
23. Which one is not related to an entity? (GRW-14)(SWL-14)
(a) person (b) concept
(c) action (d) object
24. A relationship between countries and capitals is an example of relationship: (RWP-14)
(a) One-to-one (b) One-to-many
(c) Many-to-many (d) Many-to-one

SECTION II

SHORT QUESTIONS ANSWERS

From PTB Exercise:

Q.1 Describe different steps involved in analysis stage while designing a database.

Ans. Data analysis involves the activities of Data Flow Diagram (DFD), Decision Tables (A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated.) and Decision Trees (Working on a decision tree database design. ... Each Decision tree is composed of Decision node(s), Branch(es) and Leaves. Each Decision node has a name and position. Each Branch is a path for a possible decision or occurrence).

Q.2 Explain the following with the help of figures:

(a) **Entity /Object :**

Ans. An entity is any object in the system that we want to model and store information about. Entities are usually recognizable concepts, either concrete or abstract, such as person, places, things, or events which have relevance to the database. Some specific examples of entities are Employee, Student, Lecturer.

(b) **Attribute:**

In general, an attribute is a characteristic. In a database management system (DBMS), an attribute refers to a database component, such as a table. It also may refer to a database field. Attributes describe the instances in the row of a database.

(c) **Relationship:**

A relationship, in the context of databases, is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table. Relationships allow relational databases to split and store data in different tables, while linking disparate data items.

(d) **Cardinality: (FSD-19)**

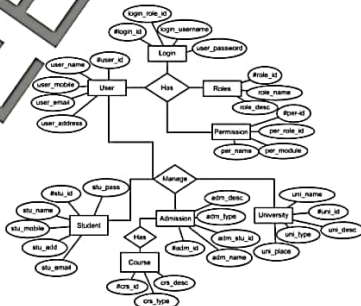
In SQL (Structured Query Language), the term cardinality refers to the uniqueness of data values contained in a particular column (attribute) of a database table. The lower the cardinality, the more duplicated elements in a column.

(f) **Modality: (FSD-19)**

Modality refers to the minimum number of times an instance in one entity can be associated with an instance in the related entity. Cardinality refers to the maximum number of times an instance in one entity can be associated with instances in the related entity.

Q.3 Draw and explain ER diagram for the system of getting admission in your college.

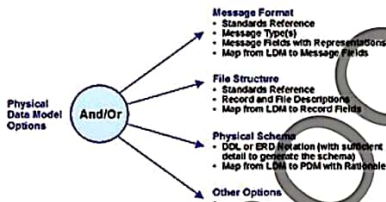
Ans.



ER Diagram for University Admission Management System

Q.4 Explain the following:**(a) Physical data model:**

Ans. A **physical data model** (or **database design**) is a representation of a **data design** as implemented, or intended to be implemented, in a **database management system**. In the lifecycle of a project it typically derives from a **logical data model**, though it may be reverse-engineered from a given **database implementation**.

**(b) Conceptual data model:**

The **conceptual model** is also known as the **data model** that can be used to describe the **conceptual schema** when a **database system** is implemented. It hides the internal details of physical storage and targets on describing entities, data type, relationships and constraints.

Q.5 What are the components of a logical data model?

Ans. A **logical data model** or **logical schema** is a **data model** of a specific problem domain expressed independently of a particular database management product or storage technology (**physical data model**) but in terms of data structures such as relational tables and columns, object-oriented classes, or XML tags.

- A data model theory has three main components: The structural part: a collection of data structures which are used to create databases representing the entities or **objects** modeled by the database.

Logical Data Model:

- Entity Types;
- Attributes;
- Relationships; and
- Domains.

Logical Data Model Components:

- An entity type is a representation of a person, place, thing, event or concept of interest to a retailer.
- An attribute identifies, names and defines a characteristic or property of an entity type.

Q.6 What elements combined, produce the physical database design? Explain.

Ans. **Physical database design** translates the logical data model into a set of SQL statements that define the **database**. For relational **database** systems, it is relatively easy to translate from a logical data model into a physical database. Rules for translation: Entities become tables in the **physical database**.

Relational Database Design Process:

- **Step 1:** Define the Purpose of the Database (Requirement Analysis)
- **Step 2:** Gather Data, Organize in tables and Specify the Primary Keys.
- **Step 3:** Create Relationships among Tables.
- **Step 4:** Refine & Normalize the Design.

Q.7 Define and explain the following terms:

Ans. (a) Data distribution strategy

Many organizations today have distributed computer networks. These organization face data distribution problem in physical database design. They have to decide nodes (or sites) in the network at which data will be located physically.

(b) File Organization:

Organizes data carefully to support fast access to desired subsets of records. **File organization** is a method of arranging the records in a **file** when the **file** is stored on disk. A relation is typically stored as a **file** of records

File organization is a way of **organizing** the data or records in a **file**. There are **several types of file organization**, the most common of them are sequential, relative (direct) and indexed.

Q.8 Define the term Analysis. (SWL-19)

Ans. Database analysis to Big Data analysis. Database analysis, or increasingly Big Data analysis, is the process of extracting this data, cleaning it, potentially merging it with other data and performing a statistical analysis to better understand customers and customer behavior.

Q.9 Briefly discuss the followings terms:

(a) Feasibility study

Ans. An analysis and evaluation of a proposed project to determine if it (1) is technically **feasible**, (2) is **feasible** within the estimated cost, and (3) will be **profitable**. **Feasibility studies** are almost always conducted where large sums are at stake. Also called **feasibility analysis**.

(b) Requirement Analysis

Requirements Analysis is the process of determining what the **database** is to be used for. It involves interviews with user groups and other stakeholders to identify what functionality they require from the **database**, what kinds of data they wish to process and the most frequently performed operations.

(c) Project Planning

The process **database** is a permanent repository of the process performance data from **projects**; it can be used for **project planning**, estimation, analysis of productivity and quality, and other purposes. The PDB consists of data from completed **projects**, with each **project** providing one data record.

(d) Data Analysis (SWL-19)

Database analysis, or increasingly **Big Data analysis**, is the process of extracting this **data**, cleaning it, potentially merging it with other **data** and performing a **statistical analysis** to better understand customers and customer behavior.

Q.10 Briefly explain the database design process with the help of a diagram.

Ans. Database design process

The database design is very important and creative activity. The major objective of database design is to map the conceptual data model to an implementation model that a particular DBMS can process. The database and database system must be acceptable to the organization and all its users. The users can easily perform different operations on the database and no problem can be created for users.

Phases in database development process:

The database development process includes a series of phases. The major phases are: planning analysis design and implementation. Each phase is divided into steps. The phases of database development process are shown graphically in figure.

1. Planning:

The database planning phase begins when a customer requests to develop a database system. It is a set of tasks or activities. It decides the resource required in the database development. It also decides the time limits for the completion of the system.

2. Analysis

Analysis is done in order to understand or study the current system. It is very important activity for the development of database system. In this phase the requirements and expectations of the users are collected and analyzed. The collected requirement help to understand the current system and for the improvement of that system (or for designing the new system).

3. Database design

The database design is very important step of database development process. In this phase, the database structure is designed.

Database design is divided into two steps:

(i) Logical database design: (GRW-19)

In logical database design the conceptual data model (or logical data model) is converted into database structure for a specific DBMS. If there is a relation DBMS, then the conceptual data model are mapped to the normalized relations.

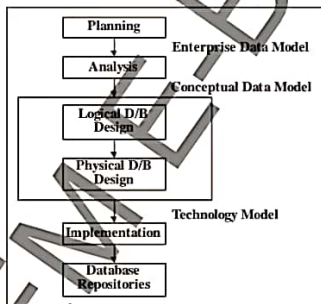
(ii) Physical database design:

In physical database design the logical database design is converted into physical storage structures such as files and tables. The indexes and access methods are also specified. Similarly physical design is also concerned with security backup and recovery etc.

4. Implementation:

After the design phase and selecting a suitable DBMS, the database system is implemented. The purpose of this phase is to install and run the database system.

In database implementation phase the database administrator (DBA) normally requires a server computer. The DBA may also need the service of network administrator to connect the user with the server. The users can share information through the server (database server).

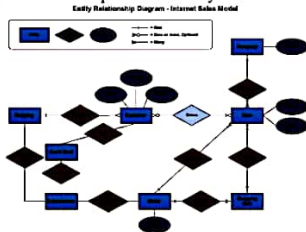


From Punjab Board:

1. Define E-R diagram.

(AJK/G1-16)

Ans. An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.



2. What is data modeling? (AJK/G1-16)

Ans. **Data modeling** is a representation of the **data** structures in a table for a company's database and is a very powerful expression of the company's business requirements. This **data model** is the guide used by functional and technical analysts in the design and implementation of a database.

3. State the purpose of database design process. (AJK/G1-16).

Ans. A **design process** suggestion for Microsoft Access. Determine the **purpose of the database** - This helps prepare for the remaining steps. Find and organize the information required. Gather all of the types of information to record in the **database**, such as product name and order number.

4. Write use of ER diagram. (AJK/G1-14)(BWP-15)

Ans. Entity Relationship Diagram is a high-level description of a conceptual data model. An **entity relationship diagram (ERD)** shows the relationships of entity sets stored in a database. By defining the entities, their attributes, and showing the relationships between them, an **ER diagram** illustrates the logical structure of databases. **ER diagrams** are used to sketch out the design of a database.

Example: There are three entities involved station railway and passenger's railway has station and attributes of railway are railway no. railway name railway type and railway seats and station attributes are waiting room, office, canteen, station id

5. Why is requirement analysis conducted? (D.G.K - 19)

Requirements analysis involves all the tasks that are conducted to identify the needs of different stakeholders. ... High-quality **requirements** are documented, actionable, measurable, testable, traceable, helps to identify business opportunities, and are defined to facilitate system design

6. What is the purpose of logical database design? (GRW-19)

Ans. The goal of **logical database design** is to create well structured tables that properly reflect the company's business environment. **Logical database design** helps you define and communicate your business' information requirements. When you create a **logical database design**, you describe each piece of information you need to track and the relationships among, or the business rules that govern, those pieces of information

7. Explain different components of logical model in detail. (AJK/G1-14), (AJK/G1-15)

Ans. **Logical Data Model Concepts:**

A **logical data model** or **logical schema** is a data **model** of a specific problem domain expressed independently of a particular database management product or storage technology (physical data **model**) but in terms of data structures such as relational tables and columns, object-oriented classes, or XML tags.

- A data model theory has three main components: The structural part: a collection of data structures which are used to create databases representing the entities or **objects** modeled by the database.

- **Logical Data Model.**

- Entity Types;
- Attributes;
- Relationships; and
- Domains.

- **Logical Data Model Components.**

- An entity type is a representation of a person, place, thing, event or concept of interest to a retailer.
- An attribute identifies, names and defines a characteristic or property of an entity type.

8. State the purpose of database design process. (FSD/G1-15)

Ans. The major objective of database design is to the conceptual data model to an implementation model that a particular DBMS can process. The database and database system must be acceptable to the organization and all its users. The users can easily perform different operations on the database and no problem can be created for users.

9. Which activities are involved in data analysis? (FSD-14)

Ans. Data analysis involves the activities of Data Flow Diagram (DFD), Decision Tables and Decision Trees.

10. Write the purpose of feasibility study. (RWP-17)(D.G.K-14)(D.G.K/G1-15)(SGD-19)

Ans. In a projects lifecycle, the project **feasibility study** is the second document that is created following the business case. The **purpose of this study** is to determine the factors that will make the business opportunity that was presented in the business case a success.

The word feasibility means possibility. Feasibility study is also called preliminary investigation. In this step, the financial, political, technical and time frame feasibility for the system are proposed. Actually feasibility study is conducted to find out whether the proposed system is possible to develop and acceptable for the organization. The objective of feasibility study to evaluate different solutions and to proposed the most feasible (possible) and least expensive solution.

11. Define entity, give examples also.

(D.G.K-14) (RWP/G1-15) (SWL-14) (D.G.K/G1-15) (LHR/G1-15) (AJK/G1-15) (LHR-19)

Ans. Anything in the real world that has a set of different attributes or properties is known as entity. An entity may be an object with a physical existence such as a person, a place, a An entity is given a unique name. An entity may have attributes (or properties) each with a unique name. The data about different entities is collected and stored in tables/relations of database. For example, the entity "Student" can be expressed in the form of database modeling as follows:

STUDENT (ROLL_NO, Name, Address, Date _ of _ Birth)

In the above example, "STUDENT" is the name of entity and its attributes are given in brackets after the entity name. From the above definition of "Student" entity we columns of the table as follows:

Roll_No	Name	Address	Date_of_Birth

car, a computer, a house, a student etc. similarly, an entity may be an object with a conceptual existence such as university course, account, a job, an event etc.

12. Define modality.

Ans. The modality define the nature of relationship as either it is optional or mandatory. If the minimum instance in one entity is zero for an instance of another relates entity, then the relationship is called optional. Similarly if the minimum instance in one entity is at least one for an instance of another related entity, then the relationship is referred to as mandatory.

13. How a relation is formed in Database?

(BWP-14)

Ans. The maximum number of instances of one entity that can be associates with each instance of another related entity is known as cardinality of relationship. The cardinality is expressed as one or many such a country can have only one capital. Similarly, a mother can have many children.

14. Define the term cardinality of a relation.

(RWP-14) (BWP-14) (SWL/G1-15) (LHR-14) (FSD-19)

Ans. The maximum number of instances of one entity that can be associates with each instance of another related entity is known as cardinality of relationship. The cardinality is expressed as one or many such a country can have only one capital. Similarly, a mother can have many children.

15. Differentiate between cardinality and molitday.

(FSD-19)

Ans. **Cardinality** refers to the maximum number of times an instance in one entity can be associated with instances in the related entity. **Modality** refers to the minimum number of times an instance in one entity can be associated with an instance in the related entity.

16. What is meant by data modeling?

(SWL/G1-15)

Ans. A representation of the real world processes, devices, objects (or concept) and their association is known as model. Data modeling is the process of identifying the data objects and the relationships between them. The data models are created during the database designing process.

Ingredient of data modeling

The word ingredient means element. A data model consists of different elements or objects. The objects that are used to represent data items in a data model are known as ingredient of data model. The important ingredient of data model are:

1. Entities
2. Attributes
3. Relationships

17. **Define database integrity.** (MTN/G1-15)(FSD/G1-15) (AJK-19) (MTN-19) (RWP-19) (D.G.K - 19)
- Ans. Data integrity e.g Database integrity means the correctness and consistency of data. It is another form of database protection. Security means that the data must be protected from unauthorized operations. Integrity is related to the quality of data. Integrity is maintained with the help of integrity constraints. These constraints are the rules that are designed to keep data consistent and correct. They act like a check on the incoming data. It is very important that a database maintains the quality of the data stored in it. DBMS provides several mechanisms to enforce integrity of the data.

TYPES OF INTEGRITY

1. Entity integrity
2. Referential Integrity

1. **Give one example of one – to – one relationship.** (MTN-17)

Ans. **One-to-One**

A row in table A can have only one matching row in table B, and vice versa.



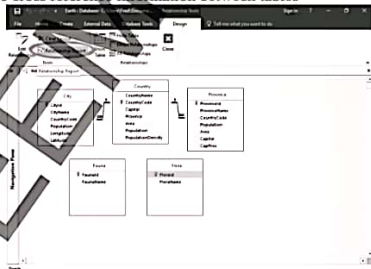
Example of a one-to-one relationship

This is not a common relationship type, as the data stored in table B could just have easily been stored in table A. However, there are some valid reasons for using this relationship type. A one-to-one relationship can be used for security purposes, to divide a large table, and various other specific purposes. In the above example, we could just as easily have put an Hourly Rate field straight into the Employee table and not bothered with the Pay table. However, hourly rate could be sensitive data that only certain database users should see. So, by putting the hourly rate into a separate table, we can provide extra security around the Pay table so that only certain users can access the data in that table.

2. **What is a relationship?**

(GRW-18) (MTN/G1-15) (LHR-14)

Ans. In a relational database (Access), the data in one table is related to the data in other tables. In general, tables can be related in one of three different ways: one-to-one, one-to-many or many-to-many. The relationship is used to cross reference information between tables



3. **List four facilities provided by good RDBMS.** (SWL-14)(D.G.K/G1-15),(GRW-14)(FSD-14)(GRW-19)

Ans. **The Benefits of using Database Management Systems**

- Data Sharing Is Improved In The Organization.
- Improvement In Data Security.
- Effective Data Integration.
- Database Management Systems Minimize Data Inconsistency.
- Better Access To Data.
- Increase In Productivity Of The End User.
- Quick Decision Making.

4. **What is DBMS? Please define also its Name two features of DBMS?** (BWP-15)(SWL-14)(MTN/G1-15)
Ans. A collection of program that are used to create, maintain and access databases is called database management system (DBMS). Today, the most popular and commonly used DBMS is Microsoft Access. The important objectives of DBMS are: data sharing, availability, evolve ability and database integrity. Some important advantages of DBMS are: data sharing, data independence, support complex data relationships, data security and data backup and recovery.
5. **What is an entity in an ERD?** (FSD-14)
Ans. Anything in the real world that has a set of different attributes or properties is known as entity. An entity may be an object with a physical existence such as a person, a place etc. Similarly an entity may be an object with a conceptual existence such as university course, account, a job an event etc. Some example of entities are teacher, class, student, car. Computer and Aeroplane.
 In e-r diagram, a rectangular box is used to represent an entity. The name of entity in capital letters is written inside the rectangular box. For example, student and computer entities are represented as follows:
- STUDENT

COMPUTER
6. **What is data integrity?** (MTN/G1-15). (FSD/G1-15) (AJK-19) (MTN-19) (RWP-19) (D.G.K - 19)
Ans. Data integrity refers to the correctness and consistency of data. Integrity is usually expressed in terms of certain constraints. These are the consistency rules that check the data before storing into database. These rules can be applied to database so that the correct data can be entered into database.
7. **How entity integration attained?** (AJK-17)
Ans. Entity integrity is attained (achieved) by specifying the primary key in the relation. No primary key value can have a null value. If no value is entered in primary key field then DBMS display an error message. If primary key consist of composite key then entity integrity is also applied to all attribute that are part of the primary key
8. **What is the concept of Data Distribution Strategy?** (LHR-14)
Ans. Many organizations today have distributed computer networks. These organization face data distribution problem in physical database design. They have to decide nodes (or sites) in the network at which data will be located physically.
9. **Distinguish between entity class and entity instance.** (AJK-19)
Ans. An entity is a real-world object that are represented in database. It can be any object, place, person or class. Data are stored about such entities. In dbms we store data in the form of table containing information about entity type like students, teachers, employees etc.
 An entity instance is a single occurrence of an entity. **Instance** means: In a database model, a row where the Primary Key identifies a single instance in Mathematics, a tuple.
10. **What is entity?** (FSD-19)
Ans. An Entity may be an object with a physical existence – a particular person, car, house, or employee – or it may be an object with a conceptual existence – a company, a job, or a university course. An Entity is an object of Entity Type and set of all entities is called as entity set.

SECTION III

LONG QUESTIONS

- Define an ER diagram. Explain its symbol with diagram. Draw an example of ER diagram.
 (AJK-19) (BWP-14) (SWL/G1-15) (LHR-19)
- What is conceptual Data Model? Explain logical database design.
 (SWL-14)
- What elements are combined to produce physical database design? Explain.
 (RWP-14)
- Define Data Modeling. What are the ingredients of Data Modeling?
 (GRW-14), (LHR-14) (FSD-14), (MTN/G1-15) (RWP/G1-16) (RWP-19)
- What is a relationship? Explain the three types of relationships among the entities.
 (FSD/G1-15) (D.G.K-14), (LHR/G1-15) (FSD/G1-16)
- Write down the properties relations in detail.
 (LHR-17) (SWL-17) (SWL-17)
- Explain different components of logical model in detail.
 (AJK/G1-15)(RWP/G1-15)(AJK/G1-14)
- Describe different steps involved in designing a data base with the help of diagram. (SGD-14) (GRW-19)
- Write in detail basic Data Distribution Strategies.
 (BWP-15) (AJK-17)



8. **Transitive dependency is removed in;** (AJK/G1-16)(BWP-14) MTN/G1-15) (MTN-18) (MTN-19) (RWP-19)
 (a) 1st Normal form (b) 2nd Normal form
 (c) 3rd Normal form (d) 4th Normal form
9. **Data inconsistency is caused by;** (AJK/G1-16)
 (a) Organized data (b) Integrated data
 (c) Independent data (d) **Redundant data**
10. **In 3NF, which form of dependency is removed:**
 (RWP/G1-15) (MTN-18) (RWP-19) (BWP-14) MTN/G1-15) (MTN-18) (MTN-19)
 (a) Functional (b) Non-functional
 (c) Associative (d) **Transitive**
11. **Referential integrity is applied on:** (D.G.K-14)
 (a) Primary Key (b) **Foreign Key**
 (c) Secondary Key (d) Candidate Key
12. **Which one refers to the correctness and consistency of data?** (GRW-14)
 (a) data independence (b) date integration
 (c) **data integrity** (d) data model
13. **The goal of normalization is to:** (LHR-14)
 (a) Increase number of relation
 (b) Get stable data structure
 (c) Increase redundancy
 (d) **None of these**
14. **In 2NF, which form of redundancy is removed:** (RWP-14)
 (a) Functional (b) **Partial**
 (c) Associative (d) **Transitive**

SECTION II

SHORT QUESTIONS ANSWERS

From PTB Exercise:

Q.2 What do we do to attain entity integrity?

Ans. Entity integrity is attained (achieved) by specifying the primary key in the relation. No primary key value can have a null value. If no value is entered in primary key field then DBMS display an error message. If primary key consist of composite key then entity integrity is also applied to all attribute that are part of the primary key

Q.3 Define referential integrity. How can it be achieved? (D.G.K/G1-15)(FSD/G1-15)(SGD-19) (D.G.K-19)

Ans. Referential integrity is achieved by establishing relationship between two tables. One table called parent table that consists the primary key. Other table contains the foreign key. The foreign key value must match the primary key value of some tuples in its home (parent) relation.

Q.4 Explain the following terms:

(a) **Synonym**

i) A synonym is created when two different names are used for the same information or attribute.

ii) The name of attribute must be same if it exists in two or more entities.

Example:

STUDENT

STD_ID

STD_NAME

STD_ADDRESS

STDSUBJECT

STD_NO

SUB_ID

SUB_NAME

(error)

In above example two entities are related and have common field STD_ID. But here it is used with different name. It is an error. It must be used with same name and STD_NO in STDSUBJECT table must be replaced with STD_ID.

(b) Homonym (FSD-19)

Homonyms are two words that are spelled the same and sound the same but have different meanings.

(c) Redundancy (RWP-14) (GRW-14) (SWL-14) (LHR/G1-15) (SGD-19) (RWP-19)

- i) **Data redundancy** is a condition created within a database or data storage technology in which the *same piece of data is held in two separate places*.
- ii) This can mean two different fields within a single database, or two different spots in **multiple** software environments or platforms. Whenever data is repeated, this basically constitutes data redundancy. This can occur by accident, but is also done deliberately for backup and recovery purpose.
- (d) **Mutual Exclusiveness of data :**
- ii) Mutually exclusive mean having one out of a few options.
- iii) In mutual exclusion only one option or set of specified options can be selected.

Example:

STUDENT

- i) STD_ID
- ii) STD_MARRIED
- iii) STD_SINGLE
- iv) In above example attributes STD_MARRIED and STD_SINGLE are mutually exclusive fields. Only one field can give the required information therefore only one field is required instead of two i.e. STD_MARRIED_STATUS.

Q.5 What is normalization? How it can be used to bring the database in a consistent state?

(RWP-19) (AJK-19)

Ans. Normalization is process of converting complex data structure into simple and stable data structures.

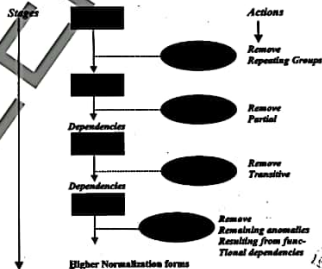
Normalization is a technique for reviewing the list of entities and their attributes to ensure that attributes are stored from where they belong.

In other words we can say that it is process of analyzing the dependencies of attributes within entities.

Before normalization a few terms must be discussed.

Normalization steps:

Normalization is accomplished in steps, each of which represents a normal form. The few of normalization through different steps can be explained as follows.



A normal form is a state of relation that can be determined by applying simple rules to that relation. Following is the brief discussion of different stages.

Database Normalization Examples

Assume a video library maintains a database of movies rented out. Without any normalization, all information is stored in one table as shown below.

Full Names	Physical Address	Movies rented	Salutation	Category
Janet Jones	First Street Plot No 4	Pirates of the Caribbean, Clash of the Titans	Ms.	Action, Action
Robert Phil	3 rd Street 34	Forgetting Sarah Marshal, Daddy's Little Girls	Mr.	Romance, Romance
Robert Phil	5 th Avenue	Clash of the Titans	Mr.	Action

Table 1

Here you see **Movies Rented** column has multiple values.

Database Normal Forms

Now let's move into 1st Normal Forms

1NF (First Normal Form) Rules

- Each table cell should contain a single value.
- Each record needs to be unique.

The above table in 1NF-

1NF Example

FULL NAMES	PHYSICAL ADDRESS	MOVIES RENTED	SALUTATION
Janet Jones	First Street Plot No 4	Pirates of the Caribbean	Ms.
Janet Jones	First Street Plot No 4	Clash of the Titans	Ms.
Robert Phil	3 rd Street 34	Forgetting Sarah Marshal	Mr.
Robert Phil	3 rd Street 34	Daddy's Little Girls	Mr.
Robert Phil	5 th Avenue	Clash of the Titans	Mr.

Q.6 When is a relation in first normal form? Explain with example.

Ans. A relation is said to be in first Normal form (1NF) if and only if each attribute in each row (or each cell) of the relation, contains only one value. It means that the relation does not contain any repeating group or multi-valued attribute.

1st Normal Form Example

How do we bring an unnormalized table into first normal form? Consider the following example:

TABLE_PRODUCT

Product ID	Color	Price
1	red, green	15.99
2	yellow	23.99
3	green	17.50
4	yellow, blue	9.99
5	red	29.99

This table is not in first normal form because the [Color] column can contain multiple values. For example, the first row includes values "red" and "green."

To bring this table to first normal form, we split the table into two tables and now we have the resulting tables:

TABLE_PRODUCT_PRICE

Product ID	Price
1	15.99
2	23.99
3	17.50
4	9.99
5	29.99

TABLE_PRODUCT_COLOR

Product ID	Color
1	red
1	green
2	yellow
3	green
4	yellow
4	blue
5	red

Now first normal form is satisfied, as the columns on each table all hold just one value.

Q.7 What are the conditions for a relation to be on second normal form? give example.

Ans. 2nd Normal Form Definition (FSD-19)

A database is in second normal form if it satisfies the following conditions:

- It is in first normal form
 - All non-key attributes are fully functional dependent on the primary key
- In a table, if attribute B is functionally dependent on A, but is not functionally dependent on a proper subset of A, then B is considered fully functional dependent on A. Hence, in a 2NF table, all non-key attributes cannot be dependent on a subset of the primary key. Note that if the primary key is not a composite key, all non-key attributes are always fully functional dependent on the primary key. A table that is in 1st normal form and contains only a single key as the primary key is automatically in 2nd normal form.

2nd Normal Form Example

Consider the following example:

TABLE_PURCHASE_DETAIL

Customer ID	Store ID	Purchase Location
1	1	Los Angeles
1	3	San Francisco
2	1	Los Angeles
3	2	New York
4	3	San Francisco

This table has a composite primary key [Customer ID, Store ID]. The non-key attribute is [Purchase Location]. In this case, [Purchase Location] only depends on [Store ID], which is only part of the primary key. Therefore, this table does not satisfy second normal form.

To bring this table to second normal form, we break the table into two tables, and now we have the following:

TABLE_PURCHASE

Customer ID	Store ID
1	1
1	3
2	1
3	2
4	3

TABLE_STORE

Store ID	Purchase Location
1	Los Angeles
2	New York
3	San Francisco

What we have done is to remove the partial functional dependency that we initially had. Now, in the table [TABLE_STORE], the column [Purchase Location] is fully dependent on the primary key of that table, which is [Store ID].

Q.8 Define transitive dependency. How it can be removed? Explain with the context of normalization.

Ans. 3rd Normal Form Definition

A database is in third normal form if it satisfies the following conditions:

- It is in second normal form
- There is no transitive functional dependency

By transitive functional dependency, we mean we have the following relationships in the table: A is functionally dependent on B, and B is functionally dependent on C. In this case, C is transitively dependent on A via B.

3rd Normal Form Example

Consider the following example:

TABLE_BOOK_DETAIL

Book ID	Genre ID	Genre Type	Price
1	1	Gardening	25.99
2	2	Sports	14.99
3	1	Gardening	10.00
4	3	Travel	12.99
5	2	Sports	17.99

In the table, [Book ID] determines [Genre ID], and [Genre ID] determines [Genre Type]. Therefore, [Book ID] determines [Genre Type] via [Genre ID] and we have transitive functional dependency, and this structure does not satisfy third normal form.

To bring this table to third normal form, we split the table into two as follows:

TABLE_BOOK

Book ID	Genre ID	Price
1	1	25.99
2	2	14.99
3	1	10.00
4	3	12.99
5	2	17.99

TABLE_GENRE

Genre ID	Genre Type
1	Gardening
2	Sports
3	Travel

Now all non-key attributes are fully functional dependent only on the primary key. In [TABLE_BOOK], both [Genre ID] and [Price] are only dependent on [Book ID]. In [TABLE_GENRE], [Genre Type] is only dependent on [Genre ID].

Q.9 What are the database anomalies? Briefly discuss insertion, deletion and modification anomalies.

(SGD-19)

Ans. Without normalization many problems can occur when trying to load an integrated conceptual model into the DBMS. These problems arise from relations that are generated directly from user views are called **anomalies**. There are three types of **anomalies**: update, deletion and insertion **anomalies**.

An **Insert Anomaly** occurs when certain attributes cannot be inserted into the database without the presence of other attributes. For example this is the converse of delete anomaly - we can't add a new course unless we have at least one student enrolled on the course. StudentNum. CourseNum.

A **Delete Anomaly** exists when certain attributes are lost because of the deletion of other attributes. For example, consider what happens if Student S30 is the last student to leave the course - All information about the course is lost. StudentNum.

Normalization generally entails splitting one database table into two simpler tables. **Modification anomalies** are so named because they are generated by the addition of, change to, or deletion of data from a database table. ... This situation is called a **deletion anomaly**.

Q.10 What anomalies arise due to transitive dependency? Discuss briefly.

Ans. Functional dependency between two or more non-key attribute in a relation is called transitive dependency. The transitive dependency is a type of functional dependency between two or more non-key attributes. Transitive dependency exists if a non-key attribute depends on any other non-key attribute. It means that a relation is not in 3NF if the value of a non-key attribute can be obtained by knowing the value of another non-key attribute.

Anomalies Due to Transitive Dependency

The following anomalies arise due to transitive dependency:

- Insertion anomaly
- Deletion Anomaly
- Modification anomaly

Suppose we have the following relation:

CUSTNO	NAME	SALESMAN	REGION
10	Ahsan	Ahmad	South
20	Babar	Bashir	West
30	Chauhan	Ahmad	South
40	Daood	Khalid	East
50	Ehtasham	Bashir	West
60	Farooq	Munir	North

Table 4.6: The SALES Relation

The above relation contains the following functional dependencies:

CUSTNO → NAME, SALESMAN

SALESMAN → REGION

The above relation is in 2NF because the primary key consists of single attribute. A transitive dependency exists in the relation. The REGION is functionally dependent on SALESMAN and SALESMAN is functionally dependent on CUSTNO. It means that REGION IS TRANSITIVELY DEPENDENT ON CUSTNO.

The transitive dependency create the following anomalies:

Insertion Anomaly, Deletion Anomaly, Modification anomaly

Q.11 Define functional dependency? How partial dependencies effect a relation?

Ans. **Functional dependency** is a relationship that exists when one attribute uniquely determines another attribute. Defining functional dependency is an important part of relational database design and contributes to aspect normalization.

Partial dependency means that a nonprime attribute is functionally dependent on part of a candidate key. (A nonprime attribute is an attribute that's not part of any candidate key.) For example, let's start with R(ABCD), and the functional dependencies AB → CD and A → C. ... That's a **partial dependency**.

Convert the ER diagram you have design in the previous exercise for the admission system of your college to relational database . also normalize the relations up to third-normal form.

From Punjab Board:

1. Why is Normalization used?

(RWP-19) (AJK-19)

The main purpose of normalization is to minimize the redundancy and remove Insert, Update and Delete Anomaly. It divides larger tables to smaller tables and links them using relationships. Data redundancy happens when the same piece of data is held in two separate place.

2. Define database anomalies.

(SGD-19)

Anomalies are problems that can occur in poorly planned, un-normalised databases where all the data is stored in one table (a flat-file database). **Insertion Anomaly** - The nature of a database may be such that it is not possible to add a required piece of data unless another piece of unavailable data is also added. E.g.

3. What is homonym?

(FSD-19)

Homonyms are two words that are spelled the same and sound the same but have different meanings. ...

A simple example of a **homonym** is the word "pen." This can mean both "a holding area for animals" and "a writing instrument."

4. Explain normalization.

(AJK/G1-14) (D.G.K-14)(SGD-14)

Ans. Normalization is a process of efficiently organizing data in database . OR

Normalization is process of converting complex data structure into simple and stable data structures. Normalization is a technique for reviewing the list of entities and their attributes to ensure that attributes are stored from where they belong.

In other words we can say that it is process of analyzing the dependencies of attributes within entities.

5. Define determinant.

(AJK-17)

Ans. The attribute (or group of attribute) that enables to obtain that value(s) of other related attribute is called determinant. It appears on the left-hand side of the arrow () in a functional dependency. For example:

Roll_No → Name, Marks, Phone

In the above example, then attribute 'Roll_No' represents the determinant.

6. What do you mean by Synonyms?

(BWP-14)

Ans. Synonyms

➤ A synonym is created when two different names are used for the same information or attribute.

➤ The name of attribute must be same if it exists in two or more entities.

Example:

<u>STUDENT</u>		<u>STDSUBJECT</u>
STD_ID	→	STD_NO (error)
STD_NAME		SUB_ID
STD_ADDRESS		SUB_NAME

In above example two entities are related and have common field STD_ID. But here it is used with different name. It is an error. It must be used with same name and STD_NO in STDSUBJECT table must be replaced with STD_ID.

7. When is a relation in second normal form 2NF? (FSD-19)

A **relation** is in **second normal form** if it is in 1NF and every non key attribute is fully functionally dependent on the primary key. The attributes IDSt and IDProf are the identification keys. The table in this example is in first normal form (1NF) since all attributes are single valued.

8. When is a relation in 3NF? (SWL-19) (LHR-19)

A **relation** will be in 3NF if it is in 2NF and not contain any transitive partial dependency. 3NF is used to reduce the data duplication. It is also used to achieve the data integrity. If there is no transitive dependency for non-prime attributes, then the **relation** must be in third normal form.

9. Define functional dependency with example. (RWP/G1-15) (LHR-19)

Ans. **Functional dependency** is a relationship that exists when one attribute uniquely determines another attribute. ... Here X is a determinant set and Y is a dependent attribute. Each value of X is associated with precisely one Y value. **Functional dependency** in a database serves as a constraint between two sets of attributes. (OR)

If we know the value of one attribute, then we obtain the value of another attribute of the relation. Suppose A and B are attributes of relation R. If we know the value of A, then we can find the value of B. We can say that B is functionally dependency upon A. the functional dependency of B on A represented by an arrow such as

A → B.

10. Define synonym.

(MTN-19)

Ans. A **synonym** is a word having the same or nearly the same meaning as another word in certain contexts. The adjective form is **synonymous**. but in database a **synonym** is an alternative name for objects such as tables, views, sequences, stored procedures, and other **database** objects. You generally use **synonyms** when you are granting access to an object from another schema and you don't want the users to have to worry about knowing which schema owns the object.

11. Differentiate between partial dependency and transitive dependency. (SGD-19) (GRW-19)(MTN-19)

Ans. A partial dependency is a dependency where A is functionally dependant on B ($A \rightarrow B$), but there is some attribute on A that can be removed from A and yet the dependency stills holds. For instance if the relation existed StaffNo, sName \rightarrow branchNo Then you could say that for every StaffNo, sName there is only one value of branchNo, but since there is no relation between branchNo and staffNo the relation is only partial.

12. What is a repeating group?

(AJK-19)

Ans. A multiple **repeating group** is where there are two or more **repeating groups** within the table to be normalised. The data within each **repeating group** is unconnected with any other data within a **repeating group**. ... Each group of **repeating** attributes will result in a separate table.

SECTION III**Long Questions**

No long questions in this Chapter by Punjab Text Book Board.



INTRODUCTION TO MICROSOFT ACCESS

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

1. A data base consists of various components called the:
 - a) tool
 - b) properties
 - c) Entities
 - d) object
2. Which of the following objects of database is used to retrieve data from database? (AJK-19)
 - a) Queries
 - b) forms
 - c) Report
 - d) All of these
3. The output of a query is in the form of a:
 - a) Table
 - b) Form
 - c) Report
 - d) Marco
4. Which of the following objects is used to retrieve data from database and present in a formatted way?
 - a) Report
 - b) Form
 - c) Table
 - d) Query
5. Micro soft Access saves the database file with the extension: (SWL-18) (MTN-19) (D.G.K - 19)
 - a) mdb
 - b) msd
 - c) mbd
 - d) None
6. A record is a complete set of:
 - a) Distinct fields
 - b) Related Fields
 - c) Designed fields
 - d) All of these
7. In Access, the structure of a table can be created in:
 - a) Design view
 - b) data sheet view
 - c) Both (a) & (b)
 - d) None

From Punjab Board:

1. It is simple to create database using. (GRW-19)
 - (a) query
 - (b) common standard
 - (c) easier programming
 - (d) wizard
2. _____ is not a database object. (MTN-18) (LHR-19)
 - (a) Table
 - (b) Query
 - (c) Report
 - (d) MS Word
3. A table is a two dimensional structure that consists of :- (MTN-18)
 - (a) X and Y coordinates
 - (b) Matrix elements
 - (c) rows and columns
 - (d) Intersection of data
4. A form is an object of: (D.G.K-14)
 - (a) Table
 - (b) Database
 - (c) Query
 - (d) Report
5. An IDE consists of: (LHR-14)
 - (a) Text Editor
 - (b) Compiler
 - (c) Debugger
 - (d) All of these

SECTION II

SHORT QUESTIONS ANSWERS**From PTB Exercise:**

Q.1 Define the database objects are used to store and retrieve data. (FSD/G1-15) (FSD-19)

Ans. A **database object** in a relational **database** is a **data** structure used to either **store** or **reference data**. The most common **object** that people interact with is the **table**. Other **objects** are **indexes**, **stored procedures**, **sequences**, **views** and many more.

Q.5 What is an IDE?

Ans. An integrated development environment (IDE) is a software application that provides comprehensive facilities to computer programmers for software development. An IDE normally consists of a source code editor, build automation tools, and a **debugger**.

Q.7 What are the advantages of using a Microsoft Access IDE?

Ans. Quick and easy to create database systems. ... Microsoft Access produces very user-friendly applications through its comprehensive programming language (VBA). MS Access produces **flexible** and adaptable database systems. Well-known so you can get help and support easily.

Using an **IDE** makes it easy to see a visual representation of the location of these files and makes it more understandable for the user. **Advantages to IDEs:** Increased Efficiency – faster coding with less effort. Collaboration – A group of programmers can easily work together within an **IDE**.

Q.8 Write a procedure to open an existing database file.

Ans. **Open** a database from within Access. If Access is already running, use the following procedure to **open** a database: Click the Microsoft Office Button, and then click **Open**. Click a shortcut in the **Open** dialog box, or in the **Look in** box, click the drive or folder that contains the **database** that you want.

Q.9 How is Microsoft Access Started or Loaded?

Ans. This section introduces how to **start Access** and open/create a database. Click on the **START** button on your computer and position your cursor over the **PROGRAMS** menu to view a list of installed programs. Once you see the list, position the cursor over **Microsoft Office** and then click on **Microsoft Access**.

Q.10 Differentiate between Form and Report.

(SWL-14) (FSD/G1-15) (D.G.K/G1-15) (LHR-14) (SGD-14) (RWP-19) (FSD-19)

Ans.

Form	Report
1. Form is primarily used for entering data.	1. Report is used for presenting the data.
2. Form is also used for displaying records but one record at a time.	2. Report is used for displaying whole records.
3. Data can be modified through the form.	3. Data can not be modified through report.
4. Form is designed to be used on screen.	4. Report are designed to be printed.

From Punjab Board:

1. What is MS access? (AJK/G1-14) (AJK/G1-15)

Ans. It allows us to create the framework (forms, tables and so on) for storing information in a database. Access provides a user-friendly forms interface that allows users to enter information in a graphical form and have that information transparently passed to the database.

2. State the use of Ms access. (AJK-17)

Ans. Some of the uses of MS Access:

It is used by small business, departments of large corporations, and by amateurs to create applications on their desktop for data applications. Access is very useful for small web based database applications hosted on IIS and using ASP.NET pages.

3. Define query.

(AJK/G1-15) (AJK/G1-16)

Ans. A **query** is a request for data or information from a database table or combination of tables. This data may be generated as results returned by Structured Query Language (SQL) or as pictorials, graphs or complex results, e.g., trend analyses from data-mining tools.

4. Define database object.

(FSD/G1-15) (FSD-19)

Ans. A **database object** in a relational database is a data structure used to either store or reference data. The most common object that people interact with is the table. Other objects are indexes, stored procedures, sequences, views and many more. Databases in Access are composed of four objects: tables, queries, forms, and reports.

5. Define any two database objects.

(RWP-14)

Ans. Databases in Access are composed of four objects: tables, queries, forms, and reports.

6. How Query is written?

(BWP-14)

Ans. A query is designed in design view by specifying table names, fields and criteria to select specific records of tables of database.

Review data from select fields

1. Open the database and on the Create tab, click Query Design.
2. In the Show Table box, on the Tables tab, double-click the Products table and then close the dialog box.
3. In the Products table, let's say that you have Product Name and List Price fields. ...
4. On the Design tab, click Run.

7. What is IDE in MS-Access AND List some advantages of IDE.

(D.G.K-14) (BWP-15) (SWL-14) (RWP-14) (BWP-14) (AJK/G1-14) (AJK-19) (D.G.K - 19) (LHR-19)

Ans. IDE stands for integrated development Environment. An IDE is very important component of DBMS used by users as an interface. The users interface with the database program through is IDE part to perform different operations on the database. The users can create database and database application very easily. Using an IDE makes it easy to see a visual representation of the location of these files and makes it more understandable for the user. Advantages to IDEs: Increased Efficiency – faster coding with less effort. Collaboration – A group of programmers can easily work together within an IDE.

SECTION III

LONG QUESTIONS

- 1) Enlist different database objects in MS-Access. (AJK/G1-16)
- 2) Briefly describe the advantages of using MS-ACCESS. (BWP-14)
- 3) Define the database objects that are used to store and retrieve data. (D.G.K/G1-15) (AJK/G1-16)
- 4) Briefly describe the advantages of using MS-ACCESS. (BWP-14)



6

TABLE AND QUERY

SECTION I

MULTIPLE CHOICE QUESTIONS

1. Which data type is the default type? (SWL-19)
(a) memo (b) number
(c) text (d) auto number
2. The maximum number of tables in a database are: (RWP-19)
(a) 01 (b) 02 (c) 03 (d) Many
3. The output of a database application is. (SGD-19)
(a) Form (b) Query (c) Report (d) Macros
4. The output of a query is in the form of a; (AJK/G1-15)/ (AJK-17)
(a) Table (b) Form
(c) Report (d) Query
5. A request for information from database terminology is called. (D.G.K/G1-15)
(a) Report (b) Letter
(c) Table (d) Query

6. The output of a query is in the form of a. (SWL/G1-15)
(a) Table (b) Form
(c) Report (d) Query
7. The default field size of a text data type field is. (SWL-18) (AJK-19)
(a) 2 (b) 5
(c) 20 (d) 50
8. The graphical query tool is known as: (RWP-18) (LHR-19)
(a) Query grid (b) Design grid
(c) Query form (d) Design form
9. Which form of property will put text into title bar of the form: (RWP-18)
(a) Title (b) Border text
(c) Text (d) MENU
10. Which data type can be used to define a field that consist of only numbers to be used in calculations. (SGD-18)
(a) Text (b) Number
(c) Memo (d) Date & time
11. How many query views are available in MS-Access: (AJK/G1-16)
(a) 2 (b) 3
(c) 4 (d) 5
12. Which of the following is degree of a table; (AJK/G1-16)
(a) Total number of rows (b) Total number of columns
(c) Total number of cells (d) Total number of foreign keys
13. The row of relation can be of ----- order; (AJK-17)
(a) Any (b) Same
(c) Sorted (d) Constant
14. Cars and parts are example of: (LHR-14)
(a) Concepts (b) Attributes
(c) Entities (d) None of these
15. 'Create command' is used to create: (RWP-14)(SGD-14)
(a) Table (b) View
(c) Report (d) Query

SECTION II

SHORT QUESTION ANSWERS

From PTB Exercise:

Q.1 Define the Different data types available in Microsoft Access?

Ans. **MS Access - Data Types.** ... MS Access supports different types of data, each with a specific purpose. The data type determines the kind of the values that users can store in any given field. Each field can store data consisting of only a single data type.

Every field in a table has properties and these properties define the field's characteristics and behavior. The most important property for a field is its data type. A field's data type determines what kind of data it can store. MS Access supports different types of data, each with a specific purpose.

- The data type determines the kind of the values that users can store in any given field.
- Each field can store data consisting of only a single data type.

Here are some of the most common data types you will find used in a typical Microsoft Access database.

Type of Data	Description	Size
Short Text	Text or combinations of text and numbers, including numbers that do not require calculating (e.g. phone numbers).	Up to 255 characters.
Long Text	Lengthy text or combinations of text and numbers.	Up to 63,999 characters.
Number	Numeric data used in mathematical calculations.	1, 2, 4, or 8 bytes (16 bytes if set to Replication ID).
Date/Time	Date and time values for the years 100 through 9999.	8 bytes
Currency	Currency values and numeric data used in mathematical calculations involving data with one to four decimal places.	8 bytes
AutoNumber	A unique sequential (incremented by 1) number or random number assigned by Microsoft Access whenever a new record is added to a table.	4 bytes (16 bytes if set to Replication ID).
Yes/No	Yes and No values and fields that contain only one of two values (Yes/No, True/False, or On/Off).	1 bit.

Q.3 What is Referential Integrity?

Ans. **Referential integrity (RI)** is a relational database concept, which states that table relationships must always be consistent. In other words, any **foreign key** field must agree with the primary key that is referenced by the **foreign key**.

Q.4 What are relationships?

Ans. **Relationships** the way in which two or more people or things are connected, or the state of being connected. "the study will assess the **relationship between** unemployment and political attitudes"

A **relationship**, in the context of databases, is a situation that exists between two relational **database** tables when one table has a foreign key that references the primary key of the other table. **Relationships** allow relational databases to split and store data in different tables, while linking disparate **data** items.

Q.6 Differentiate between Relationship and Join.

Ans. A table can be part of any number of **relationships**, but each **relationship** always has exactly two tables. In a query, a **relationship** is represented by a **join**. When you add tables to a query, Access creates **joins** that are based on relationships that have been defined between the tables.

Q.7 Define different types calculation in a query and also specifies the some Functions.

Ans. **Performing calculations in a Query:**

A query can be used to perform a calculation on a group of records. The user can perform calculation in a query by using predefined calculation in MS-Access. The user can also create a custom calculation according to the requirement.

Ms-Access provides following calculation types.

Group by: identified the group to calculate.

Sum: Add the values.

Avg: Average of the values.

Max: Finds the maximum value.

Min: Finds the minimum value.

Count: Counts the number of values.

StDev: Calculates the standard Deviation.

Var: Calculate the Variance.

First: Finds the first field value.

Last: Finds the last field value.

Expression: Create a calculate field through an expression.

Where: Indicates criteria for a field not included in the query.

From Punjab Board:

1. **What is the concept of degree of a relation?**(FSD/G1-15) (AJK/G1-16) (FSD-19) (D.G.K – 19) (SWL-19)
Ans. The number of fields (or attribute) in a relation is called degree of relation. For example, a relation with 10 fields has a degree of 10. Once a relation/ table is created, its degree usually does not change.
2. **What is list index property?** (MTN/G1-15)(MTN-17)
Ans. The indexes are created to query and sort records faster in MS-Access. The index property is used to set an indexed field. Indexing is a data structure technique to efficiently retrieve records from the database files based on some attributes on which the indexing has been done. Indexing in database systems is similar to what we see in books. Indexing is defined based on its indexing attributes.
3. **What is Linking in Ms-ACCESS?** (BWP-14)
Ans. Link to data in another Access database. We cannot **link** to queries, forms, reports, macros, or modules. When we **link** to a table in an Access database, Access creates a new table, called a linked table, which maintains a **link** to the source records and fields.
4. **What is OLE object in MS-Access?** (SGD-14)
Ans. Earlier versions of Access used a technology called **Object Linking and Embedding (OLE)** to store images and documents. By default, **OLE** created a bitmap equivalent of the image or document.
5. **Why is the important to specify data type?** (SGD-18)
Ans. **Data type** and length are the most fundamental integrity constraints applied to **data** in a **database**. Simply by specifying the **data type** for each column when a table is created, DB2 automatically ensures that only the correct **type of data** is stored in that column. ... If the data is character, use **CHAR** or **VARCHAR** data types.
6. **How a new record is added to a table using data sheet view?** (FSD-14)
Ans.
i. **To add records to a table in datasheet view**, open the desired table in datasheet view.
ii. Click the "New Record" button at the right end of the record navigation button group. It is located in the lower left corner of the datasheet view. It is the button with the arrow and asterisk [**►***] on its face.
iii. Then enter the information into the fields in the "New Record" row. It is the bottommost row in the datasheet view that displays the asterisk [*****] at the left end of the row.
iv. When you have finished entering the new record, you can move down to enter the next new record into the new row that has appeared.
v. Close the table when you are finished adding records.
7. **How does a database differ from a table?** (SWL/G1-15)
Ans. So in a layman's terminology, **database** is a collection of several components like **tables**, indexes, stored procedures and so on. A **table** is a two dimensional structure that contains several columns and rows. It contains all the data in form of several records. Note: this is an answer with reference to **rdms**.
8. **Write two characteristics of tables.** (MTN/G1-15)
Ans. A table of relational database has the following characteristics:
1. Each cell of a table contains only one value.
2. Each column has a unique name.
3. The order of columns has no importance. They may be in any order.
4. The order of rows also has no importance. They may be in an order.
5. Each row represents a record.
6. The row cannot be duplicated.
9. **How is query designed in Access?** (SGD-14)
Ans. A query is designed in design view by specifying table names, fields and criteria to select specific records of tables of database.
10. **List out two important types of query language.** (AJK-17)
Ans. **An Overview of Query Language Types:**
• **DDL (Data Definition Language)** A database schema is a visualization of information.
• **Why Should You Care That Much?** DDL queries define the structure on which you develop your application.
• **DML (Data Manipulation Language)**
• **DCL (Data Control Language)**
• **TCL (Transaction Control Language) Queries.**

- 11. List some roll of query / advantages of query.** (GRW-14)(SWL/G1-15)
Ans. There are several advantages of query. The main advantages of query are:
 ➤ Information from different tables of the database can be retrieved at once.
 ➤ The result of query can be used for Forms and for creating Reports.
 ➤ The Data of database can also be modified or deleted.
 ➤ The retrieved records from tables can be displayed in particular order based on the data of specific field(s).
- 12. State the use of query language.** (SWL/G1-15)(BWP-15) (RWP/G1-15) (LHR/G1-15) (LHR-19)
Ans. SQL is a language used to insert, update, retrieve and delete any data item from the database.
 ✓ This language is also used to grant access rights on database.
 ✓ This is a standard language used with all relational databases.
 ✓ The instruction of this language is English like and easily understandable.
- 13. How a relation is formed in data Base? (D.G.K – 19)**
A relationship between tables is an important aspect of a good relational database.
 1) It establishes a connection between a pair of tables that are logically related to each other. 2) It helps to refine table structures and minimize redundant data.
- In an Access database, we create a table relationship using one of the following methods:**
1. In the Relationships window, add the tables that we want to relate, and then drag the field to relate them from one table to the other table.
 2. Drag a field on to a table datasheet from the Field List pane.
- 14. Why is it important to specify data type and size of a field? (AJK-19) (GRW-19)**
 The field's data type determines which other properties that you can set. ... For Number and Currency fields, the Field Size property is especially important, because it determines the range of field values. For example, a one-bit Number field can store only integers ranging from 0 to 255. (OR)
 The properties of a field describe the characteristics and behavior of data added to that field. A field's data type is the most important property because it determines what kind of data the field can store.
- 15. How is criteria specified in a query? (LHR-19)**
Apply criteria to a query
 ➤ Open your query in Design view.
 ➤ In the query design grid, click the Criteria row of the field where you want to add the criterion.
 ➤ Add the criteria and press ENTER.
 ➤ Click Run to see the results in Datasheet view.

SECTION III

LONG QUESTIONS

- 1) What is MS Access? Write in detail data types used in MS Access.
 (RWP-14) (SWL/G1-15)(RWP/G1-15) (FSD/G1-15)(LHR/G1-15)(RWP/G1-16) (RWP-14) (FSD/G1-16) (RWP-19)
- 2) Define query, explain uses and advantages of query. (AJK-17) (AJK/G1-16) (AJK/G1-16)
- 3) Define Query. Discuss different types of Queries. (BWP-17)
- 4) What is field property? Discuss different field properties in detail.
 (D.G.K-14) (MTN-18) (SGD-19) (AJK-19)
- 5) What is relationship? Write procedure to define relationship between two tables in Access.
 (SGD-14) (SWL-14)
- 6) Discuss different characteristics of a Table. Also define degree and cardinality of a relation.
 (BWP-15) (LHR/G1-16)
- 7) Discuss different methods of Modifying a table.
 (FSD-14) (SGD/G1-16) (LHR-14) (MTN/G1-16) (MTN-19) (LHR-19)
- 8) What is table? Discuss different parts of a table. (MTN/G1-15)
- 9) What is query? Discuss its three uses and three advantages. (AJK/G1-16)



MICROSOFT ACCESS FORMS AND REPORTS

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

- Forms are Designed for:
 - Input Data
 - manipulate Data
 - According Change
 - All of them
- A _____ auto form displays one record at time:
 - tabular
 - columnar
 - datasheet
 - justified
- A report provides a column for each field of the records in rows under the column header is known as :
 - tabular
 - columnar
 - subform
 - none of them
- _____ can be previewed on the screen before printing.
 - report
 - form
 - subform
 - none of them
- How many are layout of report?
 - 2
 - 3
 - 4
 - 5

From Punjab Board:

- A report may be based on: (RWP-18)
 - Table
 - Field
 - Byte
 - Bits
- Forms are designed for. (SWL/G1-15)
 - input data
 - Manipulate data
 - Accepting change
 - All of these
- How many are layout of reports. (D.G.K/G1-15)
 - 2
 - 3
 - 4
 - 5
- A report that provides a column for each field of the records in rows under the column header is known as: (BWP-14)
 - Tabular Report
 - Columnar Report
 - Datasheet Report
 - Justified Report
- How many are the layout of report? (SWL-14)
 - 2
 - 3
 - 4
 - 5

SECTION II**SHORT QUESTION ANSWER****From PTB Exercise:**

Q.2 Differentiate between form or reports?

Ans. **Differences between Forms and Reports:** **Forms** are Input to the information system and **Reports** are output from the system. **Form** gathers information for essentially one record of the database. That is, information about one person or object.

Q.7 What is a report?

Ans. A **database report** is the formatted result of **database** queries and contains useful data for decision-making and analysis. Most good business applications contain a built-in **reporting** tool; this is simply a front-end interface that calls or runs back-end **database** queries that are formatted for easy application usage.

Q.8 How can you save and closed an auto report?

Ans. **Saving Reports**

1. To save a **report**, first select data points, statistics, and filters as usual ...
2. **Saved reports** can be loaded from the **report** pages ...
3. To delete a **saved report**, open the **report** and click **Delete** above the data table.
4. In the **Schedule** section of a **report** page, select "in CSV" from the "Get results:" dropdown.

Q.10 How can you apply a sort order to the report?

Ans. **Define the default sort order for a report:**

1. Open the **report**.
2. On the **Home** tab, in the **Sort & Filter** group, click **Advanced** and then click **Advanced Filter/Sort** on the shortcut menu. ...
3. In the window, double-click the field that you want to use to sort.
For queries and reports, you can also define a default sort order. The default sort order is applied to the data in the query or report when no other sort order is specified.

Q.11 How is a report created from two tables?

Ans. Open the **database Customer**. In the main **database** window, select the tab for **Reports**, then double click on the **Create Report By Using Wizard** menu selection. Select the query **SC_Purchases** from the pulldown menu at the top left of the window, and then click **OK**. Go through the **report** wizard and create a **report**.

Q.12 How can report created for a single table?

Ans. **Use the Report Button:**

1. Open the **Navigation** pane.
2. Click the **table** or query on which you want to base your **report**.
3. Activate the **Create** tab.
Click the **Report** button in the **Reports** group. Access creates your **report** and displays your **report** in **Layout** view. You can modify the **report**.

From Punjab Board:

1. Write down the procedure for creating a form?

(AJK/G1-14)

Ans. **To create a form:**

1. In the **Navigation** pane, select the **table** you want to use to create a **form**. We do not need to open the **table**.
2. Select the **Create** tab on the **Ribbon**, and locate the **Forms** group.
3. Our **form** will be created and opened in **Layout** view.
4. To save the **form**, click the **Save** command on the **Quick Access** toolbar.

2. Write the procedure to create a report using wizard.

(AJK/G1-15)

Ans. **How to Create a Report with the Report Wizard in Microsoft Access:**

1. On the **Create** tab in the **Reports** group, click **Report Wizard** ...
2. From the **Tables/Queries** drop-down list, select the **table** (or **query**) to base the **report** on. ...
3. Move the fields to include on the **report** from the **Available Fields** list box to the **Selected Fields** list box.
4. Click **Next** > .

3. Differentiate between form and report.

(SWL-14) (FSD/G1-15) (D.G.K/G1-15) (LHR-14) (SGD-14) (FSD-19)

Ans. Differences between Forms and Reports: Forms are input to the information system and Reports are output from the system. Form gathers information for essentially one record of the database. That is, information about one person or object.

4. What is Auto form?

(AJK/G1-16)

Ans. There are two types of automatic forms. The AutoForm generates a simple data entry form based on the selected table or query. The other type is the multi-step Form wizard, which allows you more control over the creation process.

5. List different types of forms in MS-Access.

(RWP-14) (MTN/G1-15)

Ans. In a database context, a form is a window or screen that contains numerous fields, or spaces to enter data. Each field holds a field label so that any user who views the form gets an idea of its contents. A form is more user friendly than generating queries to create tables and insert data into fields.

Types Forms

MS-Access provides the following types or layout of forms:

1. Columnar form
2. Datasheet Form
3. Tabular Form
5. Justified Form

6. Write any two advantages of a form. (FSD-14)**Ans. Advantages of Using Forms:**

These are many advantages of using Form. The main advantages of Form are:

1. Forms provide easy way to perform various operations on one or more tables of the database.
2. Forms provide facilities to access data from multiple tables at a time.
3. Forms provide user friendly interface.
4. Only the data of required fields of table can be accessed. In this way, confidential data of database tables can be hidden from the end users.
5. Some control objects provide features to select the required value from control objects (such as list boxes, combo boxes etc.), instead of typing through keyboard.
6. The pictures can also be displayed on the Form. Some records may contain the image or video type data. This type of data can only be viewed on Form.
7. Forms can provide proper message or guideline to users to perform different operations on the data of database. In this way, correct operations on database can be performed.

7. Define report and list two advantages of report. (D.G.K-14) (LHR-18) (MTN/G1-15)

Ans. Reports are the important part of a database program. Presentation of processed data retrieved from database tables is called report. A report can be displayed on the screen, printed on the paper or can be stored on the disk for later use. However reports are usually created for making hard copies of the required information. The main advantages of reports are as follows:

- Reports are used to display data retrieved from one or more tables of database.
- Report produce output in a formatted way according to the needs of the user.
- Using reports same data of database tables can be presented in different ways.
- Reports provided facility to process data and to generate graphs and charts.
- Reports help in making important decisions.

8. Describe the use of datasheet view in MS-Access.

(RWP-14) (RWP/G1-15)

Ans. Datasheet form is also used to display (or enter) multiple records at a time. It is like a datasheet view of table. It means that datasheet Form looks like a spreadsheet. The field names appear as labels at the top of each column. The values of records are displayed under field names. Each row displays one record of a table. This type of form is usually used as sub-form.

9. Which control is used to execute commands in MS-Access forms?

(MTN-19)

Ans. We use a command button on an Access form to start an action or a set of actions. ... On the Design tab, in the Controls group, ensure that Use Control Wizards Button ... To make the command button run an existing built-in function.

SECTION III**LONG QUESTIONS**

- 1) What are reports? Explain two types of reports. (FSD-18)
- 2) What is filter? Discuss different types of filters in MS-Access.
(GRW-14) (AJK/G1-15) (D.G.K/G1-16) (BWP/G1-16) (AJK/G1-15) (D.G.K - 19) (FSD-19) (SWL-19)
- 3) What is a form? Explain its uses and advantages. (GRW-19)

GETTING STARTED WITH C

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

1. C is a: (MTN-19)
 - (a) High level language
 - (b) Low-level language
 - (c) Assembly language
 - (d) Machine Language
2. Turbo C++ can compile for:
 - (a) C++ Programs only
 - (b) C and C++ Programs
 - (c) Turbo C programs only
 - (d) Turbo C++ programs only
3. Debug is the process of :
 - (a) creating bugs in program
 - (b) identifying and removing errors
 - (c) Turbo C programs only
 - (d) Turbo C++ programs only
5. Preprocessor directives are commands for:
 - (a) Microprocessor
 - (b) language processor
 - (c) C-Preprocessor
 - (d) Loader
6. The expression in define directive :
 - (a) can only be changed at the end of the program
 - (b) can not be changed
 - (c) can not be changed but can be redefine
 - (d) can not be assigned a value
7. Which of the following languages requires no translator to execute the program?
 - (a) C
 - (b) C++
 - (c) Assembly
 - (d) Machine
8. The .exe file is created / produced by: (MTN-18) (MTN-19) (FSD-19) (SWL-19)
 - (a) Loader
 - (b) Compiler
 - (c) Linker
 - (d) Interpreter
9. Which of the following key is used to save a file?
 - (a) F2
 - (b) F3
 - (c) F5
 - (d) F9
10. Void occupy how many bytes in memory?
 - (a) zero
 - (b) one
 - (c) two
 - (d) four

From Punjab Board:

1. The output of the compiler is called: (GRW-19) (LHR-19)
(a) program (b) source code
(c) linked code (d) object code
2. The ".exe" file is produced by: (FSD-19) (SWL-19)
(a) Linker (b) Loader
(c) Compiler (d) Interpreter
3. The process of identifying and removing errors in the program is called: (AJK-19)
(a) Debugging (b) Processing
(c) Executing (d) Linking
4. Division by zero (e.g., 5/0) is an example of: (RWP-19)
(a) Compiler error (b) Run time error
(c) Syntax error (d) Logical error
5. The extension of C Program file (GRW-19)
(a) .txt (b) .obj (c) .h (d) .c
6. The programmer usually enters source code into a computer using: (D.G.K - 19)
(a) Compiler (b) Text editor
(c) Debugger (d) Linker
7. In C language the linker creates a file with extension: (AJK/G1-14)
(a) .exe (b) .obj (c) .cpp (d) .e
8. Which type of error give un-intended results? (MTN/G1-15)
(a) Run time error (b) Logical error
(c) Syntax error (d) VB error
9. The first line of the function definition is known as: (SWL-18)
(a) function body (b) arguments
(c) function header (d) return type
10. The output of the compiler is (SWL-18)
(a) library code (b) linked code
(c) object code (d) source code
11. Which header file contains information about standard input/output functions; (AJK/G1-16)
(a) Stdio.h (b) Conio.h
(c) String.h (d) Math.h
12. Which of the following is used to donate preprocessor directive; (AJK-17)
(a) % (b) \$
(c) # (d) @
13. The statements written by programmer are called: (D.G.K-14)
(a) Source code (b) Object code
(c) Syntax (d) Debugging
14. In C Language, first line of the function definition is known as: (D.G.K-14)
(a) Function header (b) Function body
(c) Argument (d) Parameter
15. Void occupy how many bytes in memory: (FSD-14)/ (LHR-14)/ (RWP-14)
(a) Zero (b) One
(c) Two (d) Four

16. A function that does not return any thing has return type: (GRW-14)
(a) nothing (b) float
(c) void (d) null
17. C-language programs are divided into units called: (GRW-14)
(a) section (b) syntax
(c) function (d) debug
18. In text file, data is stored in: (GRW-14)
(a) ASCII-Code (b) Binary Code
(c) Octal Code (d) Text Code
19. _____ is the native language of computer.
(a) C-Language (b) Pascal
(c) Both A and B (d) Machine Language
20. Which of the following language requires no translator to execute the program: (SGD-14)
(a) C (b) C++
(c) Machine language (d) Assembly language

SECTION I

SHORT QUESTION ANSWERS

From PTB Exercise:

Q.2 List two reasons why it would be preferable to write a program in C rather than machine language.

Ans. Unlike machine because high level language (C) are programmers friendly, easy to code, debug and maintain. High level language provides higher level of abstraction from machine language. They do not interact directly with the hardware. We can also preferable C language because C is a very powerful and widely used language. It is used in many scientific programming situations. It forms (or is the basis for) the core of the modern languages Java and C++. It allows you access to the bare bones of your computer.

Q.3 What necessary steps taken to prepare a C program for execution? Explain with diagram.

Ans. Steps to Prepare a C program for Execution:

Following necessary steps are taken to prepare a C program for execution:-

- 1- Creating & Editing C program
- 2- Saving C program
- 3- Compiling C Program
- 4- Linking C Program
- 5- Loading C program
- 6- Executing C program

1- Creating & Editing C Program:

The first step is to create and edit the source program in C. the C program is usually written and edited in Turbo in Turbo C/C++ editor.

2- Saving C Program:

After writing or editing the source program, it is saved on the disk. The extension of the file must be ".c".

3- Compiling C Program :

In this step the source program is compiled. The source program is converted into machine code. The C-compiler is used to translate the program, source code into the machine code. After compiling source code an object file with extension object is created.

4- Linking C Program :

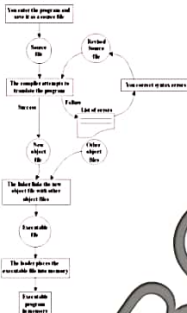
1. In this step the linker links object file to the required library files. After linking the object code to the libraries and executable file with extension EXE is created.

5- Loading C Program :

In this step the loader loads the executable file from disk into memory for execution. The program must be loaded into memory for execution.

6- Executing C Program :

In this step the program is executed on the computer. The CPU fetches instructions of program from memory one by one and takes action on them.



A diagram is shown below which shows the necessary steps that are taken to prepare a C program for execution.

Q.4 Define a **bug**. Discuss some debugging features of Turbo C++.

Ans. A **bug** can also be described as error, flaw, mistake, failure, or fault. Most **bugs** arise from programming mistakes, and a few are caused by externalities (compiler, hardware or other systems outside of the direct responsibility of the programmer).

Turbo C++ provides many useful debugging features. The debugging features of Turbo C++ are available in the Debug menu. Some important debugging features of Turbo C++ are as follows:

Single stepping Select Run > Trace Into OR press F7 key.

WatchesBreakpointsEvaluate/Modify Window

Q.5 While writing a program, how many types of errors can occur? Discuss briefly. Which one is the most difficult to locate and remove? Justify your answer

Ans. There are basically three types of errors that you must contend with when writing computer programs:

- Syntax errors.
- Runtime errors.
- Logic errors.

However, since the syntax of the incorrect code is acceptable, it will not produce a syntax error and the code will compile successfully. The logic error might only be noticed during runtime. Because logic errors are often hidden in the source code, they are typically harder to find and debug than syntax errors.

Q.6 What is programming language? Discuss the two main categories of programming language.

Ans. These high-level languages will be converted into a machine-readable form by a compiler or interpreter. **DIFFERENCE BETWEEN THE HIGH AND THE LOW LEVEL LANGUAGE:** **Low level language** is machine-readable form of program. Whereas the **high level language** will be in human-readable form. Assembly is made of abbreviated commands we can assemble into machine code. And high-level language is a lot like English and is computer-independent. Some famous high-level languages are FORTRAN, COBOL, C++, BASIC, and Java. There are also scripting languages for the web.

Q.7 Describe characteristics of high-level programming languages.

Ans. A list of some characteristics of high-level programming language is given below:

- 1- Easy
- 2- Easy to detect errors
- 3- Machine-independent
- 4- Available of library functions
- 5- Shorter programs
- 6- Well-defined syntax and standard

Q.8 Briefly describe the basic structure of a C program.

Ans. Basic Structure of C Program:

The format or way according to which a computer program is written is called the structure of the program. The program structure of different programming languages is different. C is structured programming language. It provides a well defined way of writing programs.

The basic structure of C program consists of the following main parts:

- (i) Preprocessor Directives
- (ii) The main() Function
- (iii) C Statements

To explain the structure of C program, a program example that prints a message "My first program in C." is given below.

```
#include<stdio.h>
void main (void)
{
    printf("My first program in C");
}
```

In the above program:

- 1- The header file "stdio.h" is used at the beginning of program. This file contains the declarations (or information) of standard input and output functions. These functions are used to get input and to print output. This header file is included because "printf()" function is used in the main body of program to print the message "My first program in C". The main lines beginning with # sign indicates that this is an instruction for the C-compiler. The commands that give instructions to the C-compiler are called preprocessor directive.
- 2- The "main()" function comes after the preprocessor directives. It indicates the beginning of the actual C program. It is the point at which execution of program is started.
- 3- The set statements (or instructions) of C program are written under the "main()" function between curly brackets i.e. { }. Each statement of C program ends with semicolon (;).

Q.9 How would you create, edit, compile, link and execute a C program ? Discuss briefly.

Ans. Answer Mention above previous question

Q.10 Differentiate the following:

Ans.

Preprocessor Directive and the Compiler.

(MUL-18)(RWP-18) (LHR-18) (SWL-17)

(BWP/G1-16) (SGD-18) (FSD/G1-16) (LHR/G1-15) (BWP-14) (SWL/G1-15) (AJK/G1-14)

(D.G.K/G1-15) (FSD-14) (D.G.K-17) (FSD-18) (GUJ-18) (MTN/G1-15) (AJK/G1-14) (FSD/G1-15) (SGD-14) (AJK/G1-15) (LHR-13) (AJK-19)

The **preprocessor** is considered to be the first phase in the C compilation process. In some cases, it is actually a separate program which runs before the compiler runs. ... The input to the preprocessor phase is your source code that contains preprocessor directives, comments, C declarations, definitions, and statements compiler convert source code into object code as a whole

(1) Structure and Unstructured programming languages. (FSD-19)

The main difference between structured and unstructured programming language is that a structured programming language allows a programmer to code a program by dividing the whole program into smaller units or modules.

Key Difference: the main difference between structured and unstructured programming language is that a structured programming language allows a programmer to code a program by dividing the whole program into smaller units or modules. in unstructured programming language, the program must be written as a single continuous, i.e. nonstop or unbroken block.

(2) Linker and Loader. (FSD-19)

- **Linker** : it is a system software which combines One or more object files and possible some library code into either some executable some library or a list of error. **Loader**: A program which loads the executable file to the primary memory of the machine

From Punjab Board:

1. Define the term bug and debug.

(FSD/G1-16) (D.G.K-17) (GRW/G1-15) (RWP/G1-16) (FSD/G1-15) (D.G.K-14)

Ans. Errors in the program are called bugs. The process of finding and removing bugs (or errors) in a program is known as debugging.

2. Differentiate between compiler and Interpreter.

(RWP-18) (MTN/G1-15) (GRW/G1-15) (SGD-19) (LHR-19)

Ans.

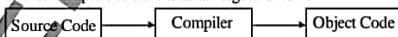
Compiler	Interpreter
1. It translates source code into object code as a whole.	1. It translates the source code into machine code statement by statement.
2. It creates an object file.	2. It does not create an object file.
3. Program execution is very fast.	3. Program execution is slow.
4. Translator program is not required to translate the program each time you want to run the program.	4. Translator program is required to translate the program each time you want to run the program.
5. It is not easier to correct the errors in the source code.	5. It is not easier to correct the errors in the source code.
6. Most of the high-level programming languages have compiler program.	6. A few high-level programming languages have interpreter program.

3. How does Compiler work, shortcut keys for compiler and execute a C program. (SWL-18)

(GRW-18) (BWP-15) (AJK/G1-15) (GRW-14) (SGD-17) (AJK/G1-15)

Ans. The language translator program that translates the complete source program into machine code as a whole is called compilation. The C and C++ compiler are best examples of compilers. The program translated into machine code is called the object program. The source code is translated into object code successfully if it is free of syntax errors. If there are any syntax errors in the source code the compiler specifies the errors at the end of compilation. The errors must be removed before the compiler can successfully compile the source code. The object program can be executed a number of times without translating it again. The short cut key of compilation is (Alt + F9).

The translating process of compiler is shown in the figure below.



4. What do you mean by delimiters/curly braces { }.

(AJK/G1-14) (MUL-18) (FSD/G1-16) (SGD-17) (GUJ-17) (GUJ-18) (FSD/G1-15) (SGD-19) (GRW-19)

Ans. The body of the function is surrounded by braces (or curly brackets { and }). These braces are called delimiters. The left brace indicates the start of the body function, whereas the matching right brace indicates the body of the function.

5. Define runtime errors.

(AJK/G1-16) (KSMR/G1-16) (MTN/G1-16) (LHR/G1-15)

Ans. The errors that are occurred during the execution of program are called the run time errors. These type of errors occurred due to the following reasons.

- When the program attempts to perform an illegal operation such as dividing a number by zero.
- If input data given to the program is not in a correct format or input data file is not found in the specified path.
- If hardware problem occurs such as hard disk error, or disk full or printer error etc.

6. What is source code? Why it cannot be executed directly? (AJK/G1-15) (BWP/G1-16)

Ans. The program instruction written by programmer according to the rules in any high level language are known as program source code. The source code is like English statements but the computer understands only the instructions in machine code (0 and 1) like object code, so computer does not understand source code directly.

7. Write at least two differences between source code and object code. (LHR-17) (BWP-15) (RWP-18) (SGD-18) (LHR/G1-16) (SGD/G1-16) (SWL-17) (SWL-14) (GRW/G1-16) (MTN/G1-16) (AJK-17) (D.G.K/G1-16) (RWP-17) (GRW/G1-16) (SGD/G1-16) (MTN/G1-15) (GRW/G1-15) (LHR-18) (SWL-19)

Ans. Source code is easy to understand and modify. Object code is difficult to understand and modify. Source code contains fewer statements than object code. Source code is user understandable and object code is machine understandable.

8. What is the difference between machine language and assembly language? (BWP-14) (RWP-17) (D.G.K-17), (LHR-17)

Ans. Machine Language:

It is a low level language understood by machines in this language combination of 0's and 1's are used to specify the instructions and data.

Simply we can say: The machine language is the internal language of the computer system. It is a difficult programming language to handle by any humans. It is usually made up of a binary string of 0s and 1s that is understood by the machine to follow any instructions. In fact, we can say that the machine can only recognize these 0s and 1s and nothing else. So, it is a language of the lowest degree made for machines only. Programmers therefore prefer to use either a high-level programming language or an assembly language to deliver various instructions by translating it to machine understandable codes known as machine codes.

Assembly Language :

It is difficult to code using Machine Language. So that Assembly Language is came. In this some symbolic notations are used to specify instructions called mnemonics. Memory location is used to specify the data. The assembly language instructions are converted into machine language using assembler for execution.

Simply we can say: Assembly language is a second generation programming language used in the computer systems. In assembly language, a programmer uses symbolic instructions instead of machine language instructions and descriptive names for data items and memory location. An assembly language program is written according to strict rules and then translated by an assembler into machine code. It is machine dependant language hence it is not portable. It has very less restrictions and also features high interaction between the operating system and the hardware thus enabling to write easy hardware dependant programs. The various symbolic notations used in the assembly language is called mnemonics.

Example is given below,

ADD R1, R2

The above instruction is more sensible than machine language instruction which is converted into machine language for execution.

9. Write two reason why it would be preferable to write a program in C rather than machine language. (LHR-14)

Ans. It is preferred to write computer program in C language rather than in machine language. The two main reasons are:

(i) Easy to Learn And Use:

C-language is high-level language. It is very easy to learn and to write program. The errors in the C program can easily be detected and removed. It is also easy to modify the C program.

On the other hand machine language is low-level language. The program is written as a combination of 0s and 1s. Machine language is not easy to learn and to write program. The errors in the program are difficult to detect and to remove. It is also difficult to modify the program.

(ii) Machine Independent:

Program written in C-language is machine independent. It means that a program written in C language can be executed (run) on any type of computer.

On the other hand a program written in machine language is machine dependent. Every type of computer has its own set of rules for written program in machine language. Program written on one type of computer system cannot run on different type of computer.

10. What is difference between machine language and high level language with examples (BWP-17)

(FSD/G1-15) (MTN/G1-15) (RWP/G1-16) (AJK/G1-16) (KSMR/G1-16) (BWP-17) (SWL-18) (BWP/G1-16) (AJK/G1-15) (AJK-17) (AJK/G1-16) (FSD-14) (KSMR/G1-16) (AJK/G1-15) (BWP-14) (SGD-14) (D.G.K-19) (SWL-19)

Ans. A **machine language** is the only language that a computer directly understands, it is usually written in zeros (0) and ones (1). A program instruction in machine language may look something like this 111010110010001 whereas, a **high-level language** is a programming language that uses English and mathematical symbols, like +, -, % and many others, in its instructions.

At the machine level you need to deal with many details of the machine that are not related to the computation. In HLLs these details are suppressed and you focus on the computation.

At the machine level many things might not be orthogonal (independent), that is you have a different way of dealing with many things (different add operators for integers and floating-point (the implementation of real) numbers is an example).

In an HLL you just put <float> + <float> or <integer> + <integer>. Same operator, very expressive, but the compiler might chose a different machine-level operator. Or some language systems leave this choice to run time. This is the basis of polymorphism in OO dispatch the correct operator based on the data type. (Static analysis of types and choice of operator at compile time results in more efficient programs.)

11. What is object oriented language?

(AJK/G1-14)

Ans. Object-oriented language (OOL) is a high-level computer programming language that implements objects and their associated procedures within the programming context to create software programs.

Object-oriented language uses an object-oriented programming technique that binds related data and functions into an object and encourages reuse of these objects within the same and other programs.

Object-oriented language typically supports the following features, at minimum:

- The ability to create classes and their associated objects
- Encapsulation
- Inheritance

Java, C++ and Smalltalk are popular examples of object-oriented languages.

12. Why is C known as strongly typed language?

(GUJ-18) (BWP/G1-16)

Ans. A **strongly-typed programming language** is one in which each type of data (such as integer, character, hexadecimal, packed decimal, and so forth) is predefined as part of the programming language and all constants or variables defined for a given program must be described with one of the data types.

As we know, there are many data types in C programming. C is an example of a **strongly typed programming language**. **Strongly typed** means that every piece of data has a specific type. ... The opposite of a **strongly typed language** is called a **loosely typed language**.

13. Write use of Turbo C++.

(BWP-17)

Ans. The compiler used for C language is Turbo C++. It is the implementation of Borland International for C language. It is used to create, edit and save programs. It also provides a powerful debugger. The debugger helps users in detecting and removing errors in programs.

14. What is statement terminator / semicolon in C-language?

(LHR/G1-15)(GUJ-17)

Ans. Every statement of C program must be terminated with semicolon (;). It called statement terminator. If any of the statement is missing the statement terminator (;), the following error message will be reported by compiler during compiling process.

15. Define constant macro.

(MUL-17) (RWP-19)

Ans. The define directive is an important preprocessor directive. It is used to define a constant known as constant macro. A constant macro is an identifier, which is assigned a particular constant value. The value of identifier remains constant during the execution of program.

16. What do you know about C-statements? (AJK-19)

Ans. A **statement** is a command given to the computer that instructs the computer to take a specific action, such as display to the screen, or collect input. A computer program is made up of a series of **statements**.

17. How is a header file included in C-program? (MTN-19)

Ans. **Header file** is a file that contains function declaration and macro definition for C in-built library functions. ... When we include **header files** in our C program using “`#include <filename.h>`” command, all C code of the **header files** are included in C program. Then, this C program is compiled by compiler and executed.

SIMPLY WE CAN SAY, **Header files** contain definitions of functions and variables, which is imported or used into any C program by using the pre-processor `#include` statement. ... Like `stdio.h` **header file** contains declarations of standard input and output functions available in C which is used for get the input and print the output just like for input and out put the header file is `#include <stdio.h>`

18. Why do you include “stdio.h” header file in c-program? (RWP-19)

Ans. `stdio.h` stands for standard input output dot **header**. `#include <stdio.h>` is a **header file** in C library which contains many standard library functions for file input and output, such as `printf()` & `scanf()`. You cannot use these functions without including `stdio.h` in your program.

19. What is the purpose of include directive? (RWP-19)

Ans. It is used to **include** an library function or to define a macro or to import codings from other programs or sources. `#include` is used import the libraries into the c programming. for getting input we need to use `scanf()` **function** to print data we need `printf()`.

20. Write the purpose of define directive. (RWP-19)

Ans. In the C Programming Language, the `#define` **directive** allows the **definition** of macros within your source code. These macro definitions allow constant values to be declared for use throughout your code. Macro definitions are not variables and cannot be changed by your program code like variables.

21. What is a computer program?(MUL-17)

Ans. A well-defined set of instructions given to the computer is called a computer program. A computer program is written in a programming language. A set of words and symbols used to write programs is called programming language. The programming languages are used to write computer programs. A programming language is a means of communication between a user and computer.

22. List four advantages of C-language. (D.G.K/G1-16) (LHR-17) (SWL/G1-15)

Ans. C-language has several advantages. The important advantages of C-language are described below

- Easy to Understand
- Middle-Level Language
- Machine Independent
- Built-in Functions
- Hardware Control
- Easy to Learn and Use
- Basis for C++
- Modularity

23. What is the role of linker in C-language? (D.G.K/G1-16) (MUL-18) (LHR/G1-16)

Ans. A program that combines the object program with additional library files is known as linker. It is a part of compiler. It combines object program and library files and saves the fir machine language program as executable file. The extension of executable files is `.exe`.

24. What is the process of linking ? (KSMR/G1-16)

Ans. The process of linking library files with object code is known as linking. The programmer may refer to many files in a C program. The library files must be linked with the object file before execution the program.

25. Define assembler. (GRW/G1-15) (RWP/G1-15) (GRW-14)

Ans. An assembler is translating program that translates the instruction of a assembly language into machine language

26. State the purpose of body of function. (RWP/G1-16)

Ans. A **function** is a group of statements that together perform a task. ... A **function** declaration tells the compiler about a **function's** name, return type, and parameters. A **function** definition provides the **actual body** of the **function**. The C standard library provides numerous built-in **functions** that your **program** can call.

27. Define the term loading. (D.G.K – 19)

Ans. Loader is the **program** of the operating system which loads the executable from the disk into the primary memory(RAM) for execution. It allocates the memory space to the executable module in main memory and then transfers control to the beginning instruction of the **program**.

28. What is the process of linking and loading differentiate between them in C Programs?

(SWL/G1-15) (SGD-14) (AJK/G1-16) (GRW-14) (BWP-14)

Ans. **Linking** is the process of taking some smaller executables and joining them together as a single larger executable. **Loading** is **loading** the executable into memory prior to execution.

Loading is **loading** the executable into memory prior to execution. There are two types of **linking**: static **linking** and dynamic **linking**. Static **linking** occurs at compilation time, hence it occurs prior to **loading** a **program**. ... Dynamic **linking** occurs at run time, so it occurs after or at the time of the **loading** of a **program**.

Dynamic **linking** is performed at run time by the operating system. In dynamic **linking** only one copy of shared library is kept in memory. **Linking** is the process of taking some smaller executables and joining them together as a single larger executable. **Loading** is **loading** the executable into memory prior to execution.

29. Who is a programmer?

(RWP-14) (RWP/G1-15) (SWL-14) (MTN/G1-16)

Ans. A person who can write, test and debug a computer program is called programmer. A programmer is also called computer programmer or coder. A person who develops a computer program is called programmer. The programmer develops programs to instruct the computer how to process data to convert into information. Programmer uses programming languages or tools to write programs

30. Define problem solving.

(MTN/G1-15)

Ans. A set of sequential steps usually written in Ordinary Language to solve a given problem is called Algorithm. It may be possible to solve to problem in more than one ways, resulting in more than one algorithm. The choice of various algorithms depends on the factors like reliability, accuracy and easy to modify.

31. What is integrated development environment IDE?

(MTN/G1-15)

Ans. IDE stands for integrated development Environment. The environment in which a computer program is written, compiled, executed, tested and debugged is called Integrated Development Environment

32. Differentiate between preprocessor directives and header file with example.

(MUL-18) (RWP-18) (LHR-18) (SWL-17) (BWP/G1-16) (SGD-18) (FSD/G1-16) (LHR/G1-15) (BWP-14) (SWL/G1-15) (AJK/G1-14) (D.G.K/G1-15) (FSD-14) (D.G.K-17) (FSD-18) (GUJ-18) (MTN/G1-15) (AJK/G1-14) (FSD/G1-15) (SGD-14) (AJK/G1-15) (LHR-14) (AJK-19)

Ans. **Preprocessor directives** appear in source code. There are many different **directives**. One of them is **#include**, which is used to include a **header file**. **Header files** contain a collection of declarations, often for functions and types (and sometimes variables) found in a library.

A header file is generally used to define all of the **functions**, variables and constants contained in any function library that you might want to use. The header file **stdio.h** should be used if you want to use the two standard I/O functions **printf** and **scanf**.

33. What is the error in following statement: `int x = 0 ; y = 0 ;` (LHR/G1-15)

Ans. Here semicolon is not used, only comma is used. If we use semicolon then we declare the data type also with variable y

28. Differentiate between logical error and syntax error and also give reason why logical errors is the most difficult error to find, Give example.

(BWP-15) (FSD-18) (SGD-18)(D.G.K/G1-15) (FSD-14)(D.G.K-14) (MTN/G1-15)(SGD/G1-16)

(BWP-15) (FSD-18) (SGD-18) (RWP-14) (GRW/G1-15) (SWL-18)

- Ans. There are generally two types of errors: **syntax errors** and **logic errors**. **Syntax errors** occur when a program does not conform to the grammar of a programming language, and the compiler cannot compile the source file. **Logic errors** occur when a program does not do what the programmer expects it to do.

Logical errors are the **most difficult** to fix. They occur when the program runs without crashing, but produces an incorrect result. The **error** is caused by a mistake in the program's **logic**. You won't get an error message, because no **syntax** or runtime error has occurred.

29. What is structured programming language?

(RWP/G1-15) (MUL-17) (RWP-17) (GRW/G1-16) (AJK-17) (GUJ-17) (FSD-19)

- Ans. In structured programming languages, the entire logic of the program is divided into number of smaller modules or function. Each module is a piece of code that implements a different functionality. There is also a less possibility of errors in the programs.

30. Name two main categories of programming languages. (LHR-18) (LHR/G1-16) (RWP-14)

(D.G.K/G1-15) (SWL-17) (FSD-14)

- Ans. (a) **Low-Level Programming Languages:**

The programming languages that are very close to machine language are called low-level programming languages. The program instructions written in these languages are in binary form (0 and 1) and symbols. The examples of low-level programming languages are:

(i) **Machine Language** (D.G.K - 19)

(ii) **Assembly Language**

(b) **High-Level Programming Languages:**

The programming languages that are close to human languages are called high-level programming languages. The program instructions written in these programming languages are similar to English language. Unlike low-level languages, these languages are easy to learn. Every high-level language has its own set of rules for writing program. These rules are called the syntax of the language. The program instructions are written according to the syntax of the programming language.

Some commonly used high-level languages are as follows:

- C/C++
- FORTRAN
- COBOL
- Java
- Pascal
- Basic

31. Why main function is used in C-Program? (MTN-19)

- Ans. The only difference is that the **main function** is "called" by the operating system when the user runs the program. The execution of a C program starts from the **main() function**. The **function main()** invokes other functions within it. ... The **main function** is simply the entry point or the start point of our program.

It is from the **main function** that the execution of a program starts. So it can be said that without a **main()** function a program can't execute.

32. Define a program. (GRW-19)

Ans. A **program** (noun) is executable software that runs on a computer. ... Instead, a **program** consists of compiled code that can run directly from the computer's operating system. **Examples of programs** include Web browsers, word processors, e-mail clients, video games, and system utilities

33. Define the term Linking. (LHR-19)

Ans. **Linking** as the name suggests, refers to creation of a single executable file from multiple object files. The file created after **linking** is ready to be loaded into memory and executed by the system. There is difference in **linking** and compilation when it comes to understanding errors.

In short **Linking** as the name suggests, refers to creation of a single executable file from multiple object files. The file created after **linking** is ready to be loaded into memory and executed by the system. There is difference in **linking** and compilation when it comes to understanding errors.

SECTION III

LONG QUESTIONS

- How would you create, edit, compile, link and execute a program?
(SGD/G1-16) (FSD/G1-16) (LHR/G1-16) (BWP-14) (FSD/G1-15)
(D.G.K/G1-16) (LHR-17) (LHR/G1-13) (RWP-17) (SGD-18)
- Describe Language processors / translator and their use. Discuss different types of language processors.
(SGD-14) (FSD-18) (AJK/G1-14) (LHR/G1-15) (FSD/G1-16) (GRW-17)
(D.G.K-17) (SWL-17) (AJK-17) (RWP/G1-15)
- What is computer language? Discuss one type of Computer Language.
(D.G.K-14) (BWP-17) (GRW-14) (MTN/G1-15)
- Describe characteristics of High Level Language / C-Language. (LHR-19) (FSD-19) (MTN-19)
- Write a detailed note on basic structure of a C Program.
(BWP-15) (LHR-14) (MTN/G1-16) (SGD-17)
- What is bug and debugging? How many types of errors can occur in writing a C-program? Which one is the most difficult to locate and remove? Justify your answer.
(RWP-14) (RWP-18) (BWP/G1-16) (RWP-14) (SWL/G1-15)
(MTN-17) (RWP-18) (SWL/G1-15) (BWP/G1-16) (RWP-14)
- What is program? Write features / characteristics / advantages commonly used
(D.G.K/G1-16) (LHR-17) (SWL/G1-15) (BWP-17)



ELEMENTS OF C

SECTION I

MULTIPLE CHOICE QUESTIONS

From PTB Exercise:

1. Variable are create in:
(a) RAM (b) ROM
(c) Disk (d) Cache
2. Which of the following is valid character constant?
(a) a (b) "a"
(c) 'a' (d) None
3. Which of the following data type offers the highest precision?
(a) float (b) long int
(c) long double (d) unsigned long int
4. In C language , the symbol '=' represents:
(a) Comparison operator (b) assignment operator
(c) Equal to operator (d) None
5. The symbol "=", represent :
(a) comparison operator (b) assignment operator
(c) Equal-to operator (d) none of these
6. Which of the following operator has precedence?
(a) ! (b) +
(c) = (d) = =
7. Relational operators are used to:
(a) Establish a relationship among variables
(b) compare two values
(c) construct compound condition
(d) perform arithmetic operation
8. C is strongly typed language, this means that:
(a) Every program must be compiled before execution.
(b) Every variable must be declared before it is being used.
(c) The variable declaration also defines the variable.
(d) Sufficient data types are available to manipulate each type of data.
9. The logical not operator, denoted by !, is a :
(a) ternary operator (b) Uniary operator
(c) binary operator (d) bitwise operator
10. $a + = b$ is equivalent to:
(a) $b += a$ (b) $a += b$
(c) $a = a + b$ (d) $b = b + a$

From Punjab Board:

1. Which data type is most appropriate for storing a name? (FSD-19)
(a) float (b) int
(c) char (d) long
2. The left side of an assignment statement holds: (MTN-19)
(a) Variable (b) Constant
(c) Expression (d) Digit
3. What does a compound condition use to join two conditions? (SGD-19)
(a) Relational Operator (b) Relational Result
(c) Logical Result (d) Logical operator
4. The number of digits after a decimal point is called: (SGD-19)
(a) Significance (b) Range
(c) Precision (d) Scope
5. The naming prefix for a label control is: (GRW-19)
(a) lab (b) lbl
(c) lbe (d) lbl
6. Which operator in C is called a ternary operator? (MTN-19)
(a) If (b) b++
(c) ? (d) ()
7. In C-Language, variable name(s) cannot begin with a(n) (D.G.K - 19)
(a) Number (b) Lower - case letter
(c) Upper-case letter (d) Underscore
8. The number of bytes used by float data type in C is: (AJK-19)
(a) 2 (b) 4
(c) 6 (d) 8
9. The functions used for input and output is stored in: (MTN-19)
(a) Stdio.h (b) Conio.h
(c) Math.h (d) Inout.h
10. Which character signifies the beginning of an escape sequence? (SWL-19) (SGD-19)
(a) . (b) =
(c) * (d) \
11. Variable name cannot contain: (AJK/G1-14)
(a) Number (b) Underscore
(c) Letter (d) Period
12. Variable and constant names cannot contain a(n):- (MTN-18) (MTN-18) (LHR-19)
(a) Number (b) Underscore
(c) Letter (d) Period
13. Which of the following is a logical operator? (SWL-18)
(a) AND (b) >
(c) + (d) =
14. Find the output of following expression $10 * (24 / (5 - 2)) + 13$. (D.G.K/G1-15)
(a) 80 (b) 90
(c) 93 (d) 96
15. $a + = b$ is equivalent to. (D.G.K/G1-15) (SWL-19)
(a) $b + = a$ (b) $a + = b$
(c) $a = a + b$ (d) $b = b + a$
16. Variable that holds a large group of similar type of data is called: (BWP-15)
(a) Array (b) Constant
(c) Scalar Value (d) Multiple Variables

17. Which Keyword is used to trigger the execution of another procedure: (BWP-15)
(a) Go (b) Call
(c) Start (d) Run
18. Which is not an arithmetic operator: (BWP-15)
(a) + (b) Mod
(c) & (d) \$
19. Comments are used to increase the of the program: (BWP-15)
(a) Visibility (b) Beauty
(c) Readability (d) Complexity
20. Memory is allocated to a local variable at the time of its: (BWP-15)
(a) Declaration (b) Destruction
(c) Definition (d) First Reference
21. Relational operators allow you to ----- numbers. (D.G.K/G1-15)
(a) Compare (b) Add
(c) Multiply (d) Divide
22. The Logical NOT Operator denoted by ! is a: (BWP-15)
(a) Ternary Operator (b) Unary Operator
(c) Binary Operator (d) Bitwise Operator
23. Which of the following is not ternary operator? (MTN/G1-15)
(a) ++ (b) +
(c) -- (d) ?
24. Relational operator allows us to --- numbers: (LHR/G1-15)
(a) Compare (b) Add
(c) Multiply (d) Divide
25. The number of bytes used by int data type in C is: (SWL-18)
(a) 4 (b) 8
(c) 16 (d) 32
26. Logical operator: (RWP-18)
(a) And (b) =
(c) >= (d) <>
27. Int is a _____ in C: (SGD-18)
(a) Special word (b) Key word
(c) Cut word (d) First word
28. Relational operator allow you to: (MTN-18)
(a) Compare (b) Add
(c) Multiply (d) Divide
29. When the relational expression is false, it has the value: (AJK/G1-14)/(LHR-14)(SWL-19) (RWP-19)
(a) Zero (b) Less than zero
(c) One (d) Greater than zero
30. Variables are created in; (AJK/G1-15)
(a) RAM (b) ROM
(c) Hard disk (d) Cache

31. The number of bytes used by character data type is; (AJK/G1-15)
(a) Zero (b) Two
(c) Three (d) One
32. A memory location with some data that cannot change is called; (AJK/G1-16)
(a) Constant (b) Variable
(c) Named constant (d) Symbolic constant
33. Global variable is created in; (AJK/G1-16)
(a) Random Access Memory (b) Read only Memory
(c) Cache memory (d) Hard disk
34. Variable that holds a large group of similar type of data is called: (GRW-14)
(a) constant (b) scalar value
(c) array (d) multiple variable
35. Variables are created in; (AJK-17)
(a) ROM (b) RAM
(c) Hard disk (d) USB
36. An identifier whose value can be changed during the execution of the program is called:
(a) Token (b) String
(c) Constant (d) Variable
37. Which of the data type offers the highest precision? (FSD-14)/ (LHR-14)
(a) Float (b) Long int
(c) Long double (d) Unsigned long int
38. The symbol "=" represents: (FSD-14)/ (LHR-14)
(a) comparison operator (b) Assignment Operator
(c) Equal to operator (d) None of these
39. Which operator has lowest precedence? (FSD-14) (GRW-19) (RWP-19) (LHR-14)
(a) ! (b) +
(c) = (d) ==
40. An expression that uses a relational operator is known as: (FSD-14)/ (LHR-14) (GRW-19)
(a) Operational (b) Sequential
(c) Serial (d) Relational
41. This means to increase a value by one: (FSD-14) (LHR-14)
(a) Modulus (b) Increment
(c) Decrement (d) None of these
42. The symbols that perform operations on data are called: (GRW-14)
(a) syntax (b) operands
(c) operators (d) operation code
43. An ampersand before the name of a variable denotes: (GRW-14) (FSD-19)
(a) actual value (b) variable name
(c) data type (d) address
44. Which is a numeric data type? (LHR-14)
(a) Floating point (b) Integer
(c) Both A and B (d) None of these
45. The number of bytes used by int data type in C is: (RWP-14)
(a) 8 (b) 6
(c) 4 (d) 2

46. Which of the following is a valid character constant? (SWL-14)
(a) a (b) "b"
(c) '6' (d) 4
47. C statements end with a: (SWL-14)
(a) Period (b) Comma
(c) Semi-colon (d) Question mark
48. Relational operators allows you to _____ numbers: (SGD-14)
(a) Add (b) Compare
(c) Divide (d) Multiply

SECTION II

SHORT QUESTION ANSWERS

From PTB Exercise:

1. What is an identifier? Discuss the two types of identifier. (MTN-19)
- Ans. An identifier, in C#, is the user-defined name of a program element. It can be a namespace, class, method, **variable** or interface. Identifiers are symbols used to uniquely identify a program element in the code. They are also used to refer to types, constants, macros and parameters.
User define and standard identifiers are main types of identifiers.
2. What is variable? Discuss the difference between declaring and defining a variable.
- Ans. Every **variable** has a name, called the **variable name**, and a data type. A **variable's** data type indicates what sort of value the **variable** represents, such as whether it is an integer, a floating-point number, or a character. Because of their inflexibility, constants are used less often than **variables in programming**.
Declaration of a variable in C means that the **variable** is only **declared** and allocated a block of memory but still has no value. **Definition of a variable** means to assign or initialize it with some specific value. For example: `int a; //declaration`.
3. Write down the rules of naming variable in C.
- Ans. Rules for naming variables:
All **variable names** must begin with a letter of the alphabet or an underscore(_).
After the first initial letter, **variable names** can also contain letters and numbers. ...
Uppercase characters are distinct from lowercase characters. ...
You cannot use a C keyword (reserved word) as a **variable name**.
4. Differentiate the following:
- Ans. I. **Constant and variable**.
A **constant** once defined cannot be changed. You use **constant** to have a value, which is guaranteed to remain the same by C language. ... A **variable** can change value, a **constant** has a fixed value.
5. **Character constant and Numeric constant**
- Ans. The string **constants** are always enclosed in double quotes, while **character constants** are enclosed in apostrophes (single quotation marks). A **character constant** has an equivalent integer value, whereas a single-character string **constant** does not have an equivalent integer value. (OR)
The value of a **character constant** containing a single **character** is the **numeric value of the character** in the current character set. **Character constants** containing multiple **characters** within the single quotation marks have a value determined by the compiler.
- Standard data type and User defined data type**
Data type defines maximum or minimum set of values and set of operations on those values.
Standard data type is one which is predefined in C language. In C language there are four basic standard data types (int, float, char and double). C-language also allows the user to define his/her own data types. The data types defined by the user are known as user defined data types.

6. Keyword and Identifier

Ans. Every language has keywords and identifiers, which are only understood by its compiler. Keywords are predefined reserved words, which possess special meaning. An identifier is a unique name given to a particular variable, function or label of class in the program.

7. What is data type? Discuss various C data types to manipulate integers, following point numbers and characters.

In the C programming language, **data types** are declarations for memory locations or variables that determine the characteristics of the **data** that may be stored and the methods (operations) of processing that are permitted involving them.

There are five basic data types associated with variables:

- **int - integer:** a whole number.
- **float - floating point value:** ie a number with a fractional part.
- **double -** a double-precision floating point value.
- **char -** a single character.
- **void -** valueless special purpose **type** which we will examine closely in later sections.

8. How many types of operators are available in C? Describe briefly. Also mentioned their precedence.

Ans. TYPES OF OPERATORS ARE AVAILABLE

- Arithmetic operators.
- Relational operators.
- Logical operators.
- Bitwise operators.
- Assignment operators.
- Conditional operators.
- Special operators.

Operator Precedence

It determines the order of evaluation in an expression. Each operator has a fixed order. The operator with higher priority evaluated before those with lower priority. When an expression is a combination of arithmetic, relational and logical expression then the expression is evaluated in the following sequence.

- Arithmetic
- Relational
- Logical

In arithmetic expression the computer performs one operation at a time. All operations are evaluated from left to right. Multiplication and division has highest priority. After the completion of these operations addition and subtractions are evaluated from left to right. First parentheses are evaluated in any expression and the inner most parentheses have highest priority. All the relational operators have equal importance and they are evaluated from left to right. In logical expression operators are evaluated in a following sequence

- Not
- And
- Or

The following table represents the operator precedence.

Operator	Precedence (highest to lowest in downward direction)
! Logical not	Highest
*, /, %	
+, -	
<, <=, >, >=	
==, !=	To
&& (and)	
(or)	
=	Lowest

9. What data type would you use to represent the following items: number of children at your school, a letter grade on the exam and the average marks of your class

Ans. Number of children at your school = int,
letter grade on the exam = char
average marks of class = float

10. Write a note on the following:

Ans. **Arithmetic Expression (BWP-14) (BWP-15)(RWP-14) (SWL-19)**

An expression that is a combination of numeric variables, numeric constants and arithmetic operators is called arithmetic expression. In a C program, the arithmetic expression is used to calculate the value of an arithmetic formula. The arithmetic expression is evaluated and single value is returned.

11. **Comments of C**

Ans. A comment starts with a slash asterisk /* and ends with a asterisk slash */ and can be anywhere in your program. Comments can span several lines within your C program. Comments are typically added directly above the related C source code.

12. Let w,x,y and z be the name of your float variables, and let a,b and c be the name of three types of variables. Which of the following statements contains one or more violations of the rules for forming arithmetic expression. Rewrite these statements so that it is consistent with these rules.

Ans.

Invalid Expression	valid expression
z = 4.0 w * y;	z = 4.0*w * y;
y= yz ;	y= y*z ;
a = 6b4;	a = 6*b*4;
c = 3(a+b);	c = 3*(a+b);
z = 7w + xy;	z = 7*w + x*y;

13. Assume that each of the following variables declarations:

Ans.

int a, b, c, d,p; float v, w, x, y, z; Evaluate each of the following statements assuming a is 2, z is 1.3, c is 3, y is 0.3E+1	
Expression	Evaluation
v = a* 2.5 /y;	0.666667
w = a/y;	0.666667
p = a/d;	0
x = (a+c)/(z+0.3);	-3.00000
b = d/a+d%a;	2
y = c/d*a;	0.000000

From Punjab Board:

1. Define an identifier with example. (MTN-19)

Ans. An identifier is a string of alphanumeric characters that begins with an alphabetic character or an underscore character that are used to represent various programming elements such as variables, functions, arrays, structures, unions and so on. Actually, an **identifier** is a user-defined word. Just like age, marks, result etc.

2. Find output of the expression{ 10* (24/(5-2))+13}

(RWP-14)

Ans. = 93

3. Write the mathematical expression in C expression.

(BWP-14)(BWP-15)(RWP-14)(SWL-19)

Ans. **Arithmetic Expressions**

- C has a wide range of operators. ...We know that the **arithmetic** operators in C language include unary operators (+ - ++ --), multiplicative operators

Example Simple arithmetic expressions.

4. What is assignment sequence? (RWP-18)

Ans. The goal of assignment sequencing is a stepwise development of students' skills (i.e., movement from preliminary to higher level critical thinking or disciplinary skills) through a progression of assignments that all fit together to produce a larger end product.

5. What is the value of Y after the following code executes?

(MTN/G1-16)

Ans. float y = 3.4 + sqrt (25.0);

15. What is an Arithmetic Expression? (BWP-15)(RWP-14) (SWL-19)

Ans. A type of expression in which only arithmetic operators are used is called arithmetic expression. An arithmetic expression may contain integers and floating point numbers.

16. Name data types used to store real data.**(BWP-17) (SWL-17) (FSD-18) (GRW-14)****Ans. Data Types for Floating Point Numbers or real data**

These are the numbers fractional part. e.g. 2.13, 0.54 etc.

ANSI C specifies three floating point data types.

1. float
2. double
3. long double

All are different in memory requirement and range.

Data Type	Bytes	Decimal Places	Range of values
Float	4	6	10^{-38} to 10^{+38}
Double	8	15	10^{-308} to 10^{+308}
long double	10	19	10^{-4932} to 10^{+4932}

14. Describe single line comments in C. (BWP-14) (SGD-44)

Ans. Single line comments are inserted by typing two forward slashes before the beginning of a line.

Example // this program Calculates temperature

15. Write a statement to declare an integer variable i initialized to-1(SWL-19)

Ans. int i=1;

16. How are comments added on multiple lines?**(D.G.K-14) (LHR-14)**

Ans. Multi-line comments are used to provide such informative notes which extend to multi lines. Multi line comments are inserted by typing slash asterisk (/*) in the beginning and asterisk slash (*/) at the end of paragraph. Then the compiler ignores all lines within this block. By omitting ending letters */ will cause the whole program code beneath the opening letters /* for comments to be commented.

Example /* this program calculates factorial of a number.

Number is positive integer. And result is long integer. */

17. Why are comments used in C program Also give an example. (RWP-17) (AJK-17) (MTN/G1-15)**(RWP/G1-15) (FSD-19)**

Ans. Comments are used to increase the readability of the program. Comments are notes about different lines of code that explain the purpose of the code. The user can insert information notes in the code. It helps in debugging and modifying the program later.

Why comments used

1. It increases the readability of the program.
2. It is used to add informative notes about the statements in a program that helps in debugging and modifying programs.
3. If is a non-executable statement (needed).
4. Comments play no role in the execution of the program.
5. The compiler does not translate these statements.
6. It also explains the logic of the program.
There are two types of comments in C.
 1. Single line comments
 2. Multi line comments

18. Define keywords / reserved words with examples. (RWP/G1-16) (GUJ-17) (BWP-17) (AJK/G1-15) (GRW/G1-15) (SWL-18) (FSD-18) (MUL-18) (LHR-19)

Ans. These are the words, which have predefined meaning in C language. These are 32 keywords in C language. These words cannot be used or redefined for any other purpose in the C program. All keywords are written in lower case.

List of C keywords:

1.auto	2.double	3.void	4.unsigned
5.long	6.switch	7.signed	8.for
9.int	10.struct	11.extern	12.typedef
13.do	14.if	15.goto	16.default
17.break	18.else	19.case	20.continue
21.static	22.while	23.union	24.return
25.volatile	26.sizeof	27.short	28.float
29.char	30.const	31.enum	32.register

19. What is the use of assignment statement / What does the symbol = do in C. (SWL-14) (LHR-18) (FSD-14) (D.G.K/G1-15)

Ans. Assignment statement is used to assign a value or a computational result to a variable. The symbol equal (=) represents assignment operator.

There are two types of assignment statement.

1. Simple Assignment Statement.
2. Compound Assignment Statement.

Simple Assignment Statement

- The value is written on the right side of the operator and the variable is left side of the operator in the assignment statement.
- Writing variable to right and value to left causes a syntax error.
- General form of assignment statement
- Variable = expression

Expression can be variable, constant, arithmetic, relational or logical expression e.g.

A = 10;

D=B*B-4*A*C;

Compound Assignment Statement

- It is used to assign one value to more than one variables.
- General form of compound Statement
- Var-1=var-2=var-3=-----=var-n=var

Example

int x,y,z;

x=y=z=10; //The value 10 is stored in variables x,y and z

Compound Assignment Expression And Compound Assignment Operators

- It is used to add, subtract, multiply or divide the value to or from a variable.
- There are four compound assignment operators (+=, -=, *=, /=) that can increment or decrement the value of the variables by other than one.
- General form of writing compound assignment expressions is
- Variable operator = value

Example

int J = 10, I = 20;

J+=5; // VALUE OF J IS INCREASE BY 5 AND J=15 OR J=J+5

I*=5; //VALUE OF I IS MULTIPLIED BY 5 AND STORED IN I (I=100)

//OR EQUIVALENT TO I=I*5

20. **How does cancellation error occur?** (SWL-17), (GUJ-18)

Ans. The cancellation error occurs when a very small value is added to or subtracted from a very large value. The result of arithmetic operation may be unexpected. The large value may cancel out the small value and result may be the same as large value. For example, when a large value 158.01 is added to a small value 0.0000001245. The result of addition may be 1589.01 on some computers.

Or The cancellation error occurs due to the manipulation of very large and very small floating numbers are manipulated. The manipulation may show unexpected result. The larger number may cancel out the smaller number when both numbers are added.

21. **What is garbage value?** (MUL-18)

Ans. When a variable is declared, the compiler reserves the space for it. If we do not initialize it then it may contain a meaningless data is called garbage value, and with the involvement of such variable may cause unexpected results.

To avoid this situation, all variables must be declared and initialized according to program requirement.

22. **List any four types of integer data in C-language.** (D.G.K/G1-16)

Ans. The data types used for storing integer data are: int, short int, long int, unsigned int and unsigned long int.

23. **What happens when arithmetic underflow occurs?** (LHR/G1-16)

Ans. The arithmetic underflow occurs due to the manipulation of two very small numbers. The result may be too small to be represented when two very small numbers are manipulated. The result is represented as zero in this situation

Simply we can also say the Arithmetic underflow occurs when a very small value is added to or subtracted from another very small value. The result may be too small to be represented accurately, so it will be represented as zero. For example, when a value 0.00000018901 is added to a value 0.00000001245. The result of addition may be 0 on some computers.

24. **How are characters stored?** (MTN/G1-16)

Ans. The characters are stored in ASCII code form. ASCII stands for **American Standard Code for Information Interchange**. The ASCII code values of characters are used when they are added, subtracted or compared. The 'char' data type is used to store a single character such as 'a', '\$', '9' etc. It takes one byte in memory. When a character is stored in memory, the ASCII value of the character is assigned to the variable. The ASCII value of 'A' is 65 and 'a' is 97. Similarly, ASCII value of 'B' is 66 and 'b' is 98 and so on.

25. **What is the difference between implicit and explicit type casting?** (RWP-17) (D.G.K/G1-16)

(RWP-14) (SGD/G1-16) (MTN/G1-15)

Ans. The process of converting the data type of a value during execution is known as **type casting**. The types casting are Implicit type casting and Explicit type casting.

Implicit type casting is performed automatically by the C compiler. The operands in arithmetic operation must be of similar types. If the data types of operands are different, the value with lower data type is converted into higher data type

Explicit casting is performed by programmer. It is performed by using **cast operator**. The cast operator tells the computer to convert the data type of a value.

26. **What do you mean by Case Sensitive in C-Language?** (LHR-14) (AJK/G1-14) (SGD-17) (D.G.K-14)

Ans. C-language is a case sensitive language. Each statement of C program is normally written in lowercase letters. However variable names or user defined function names with uppercase letters can be used in C program. user define identifier (A, a) both are different variables name.

27. **Define the concept of operator's precedence.**

(KSMR/G1-16) (SGD/G1-16) (AJK/G1-16) (FSD/G1-15) (FSD/G1-15) (FSD/G1-16)

Ans. It determines the order of evaluation in an expression. Each operator has a fixed order. The operator with higher priority evaluated before those with lower priority. When an expression is a combination of arithmetic, relational and logical expression then the expression is evaluated in the following sequence.

- Arithmetic
- Relational
- Logical

- The following table represents the operator precedence.

Operator	Precedence (highest to lowest in downward direction)
! Logical not	Highest
*, /, %	
+, -	
<, <=, >, >=	
==, !=	To
&& (and)	
(or)	
=	Lowest

28. Which operator are used to evaluate compound condition? (MUL-17) (GRW-14) (SGD-17) (RWP/G1-16) (SWL/G1-15) (RWP-19)

Ans. A type of comparison in which more than one conditions are evaluated is called **compound condition**. It executes a statement or set of statements by testing many conditions. Logical operators are used to evaluate compound conditions. The logical operators in C language are AND operator (&&), OR operator (||) and NOT operator (!).

29. What is the use of OR operator? (FSD/G1-16) (FSD/G1-15) (FSD/G1-15) (SGD-14) (BWP-14)

Ans. The symbol used for OR operator is (||). It is used to evaluate two conditions. It produces **true** if either condition is **true**. It produces **false** result if both conditions are **false**.

30. Differentiate between variable and constant with types. (MTN/G1-15) (GRW/G1-15) (AJK/G1-15) (GRW-14) (AJK/G1-15) (LHR-17) (FSD/G1-16) (LHR/G1-15) (D.G.K/G1-15) (LHR-14) (GRW/G1-16)

Ans. A **variable** is a named memory location or memory cell. It is used to store program's input data and its computational results during execution. The value of variable may change during the execution of the program. However, the name of variable cannot be changed.

A **constant** is a quantity that cannot be changed during program execution.

C language provides two types of constants. These are numeric constants and character constants.

Example:

```
#define pi 3.142857
```

Pi whose value remains same during program execution.

There are two types of constant:

1. Numeric constants.
2. Character constants.

Numeric Constant

- These constants consist of numbers.
- There are two types of numeric constants.
- i. Integers
 - ii. Float

Integer constants represent values that are counted and without decimal or fractional part. e.g. +56, -678 etc.

Floating constants represent values that are measured. e.g. 4.786, 0.45 etc.

Character Constant

It is a single alphabet, a single digit or a single symbol enclosed within single quotation (apostrophes).

The maximum length of a character constant is 1.

Examples are: '5', '+', 'B' etc.

The value stored in a variable can be changed during the execution of the program. The values stored in constants cannot change.

31. Define assignment statement with example. (LHR-14)

Ans. In most expression-oriented programming languages (for example, C), the assignment statement returns the assigned value, allowing such idioms as $x = y = a$, in which the assignment statement $y = a$ returns the value of a , which is then assigned to x .

Once we declared a variable we can use it, but not until it has been declared - attempts to use a variable that has not been defined will cause a compiler error. Using a variable means storing something in it. You can store a value in a variable using:

name = value;

For example:

a=10;

Stores the value 10 in the int variable a. What could be simpler? Not much, but it isn't actually very useful! Who wants to store a known value like 10 in a variable so you can use it later? It is 10, always was 10 and always will be 10. What makes variables useful is that you can use them to store the result of some arithmetic.

32. Give some examples of valid variable name. (AJK/G1-14) (D.G.K-14)

Ans. Valid variable name are roll_no, XY, ABC, S2, totalmarks and so on

33. What is variable Initialization? (BWP-15) (GRW-19) (FSD-14)

Ans. Assigning a value to a variable at the time of its declaration is called variable initialization. The equal sign = is used to initialize a variable. Variable name is written on left side and the value is written on the right side of equal sign. For example float a = 5;

float b = 15;

float c = 10;

34. Differentiate between declaring and defining a variable. (D.G.K-14) (GRW/G1-16) (AJK/G1-14)

(AJK/G1-15) (LHR-14) (MTN/G1-15) (D.G.K/G1-15) (SGD-18) (AJK/G1-15)

(D.G.K-17) (SWL-14) (AJK-17)

Ans. Variable declaration only informs the compiler the name of variable and its data type. It does not reserve memory space for variable in the memory. Variable definition informs the compiler the name of variable and its data type. A memory location is also reserved for the variable. The size of memory location reserved for variable depends upon the data type of variable, just like int a, float b;

35. Write down two rules for declaring naming variable in C. (BWP/G1-16) (LHR/G1-15)

(LHR/G1-16)

Ans. The first character of Variable must be a letter or underscore. Blank spaces are not allowed in variable names. Variable may include letters, numbers and underscore (J. Reserved word cannot be used as variable name).

36. List any four types of operators in C. (SGD-17)

Ans. Different types of operators available in C are. Arithmetic operators, Relational operators, Logical operators, Assignment operator, Increment and decrement operators, and Compound assignment operator.

37. Define increment decrement operators. (SWL/G1-15)

Ans. It is used to increase the value of the variable by one. ++ is used as increment operator. It can be used before or after the variable name e.g. x++ or ++x. It cannot be used with constants and expressions. Only the variables can be incremented. It is unary operator.

Decrement Operator (--)

It is used to decrease the value of the variable by one. -- is used as decrement operator. It can be used before or after the variable name e.g. x-- or --x. It cannot be used with constants and expressions. Only the variables can be decremented. It is unary operator.

Both are two types of decrement / increment operator.

1. Postfix decrement / increment operator.

2. Prefix decrement / increment operator.

38. What are the uses of Operators? (BWP-15) (SWL-18)

Ans. Operators are the symbols that are used to perform certain operations on data. C provides a variety of operators. These are given below:-

➤ Unary-operators required single data item.

➤ Binary operators required two data items.

Following types of operators are available in C.

1. Arithmetic Operators.
2. Relational Operators.
3. Logical Operators.
4. Increment and decrement Operators.
5. Assignment Operators.
6. Bitwise Operators.

39. Define relational operators. Enlist different types of relational operators. (RWP-14)

Relational operators with constants and variables form relational expressions. These are six operators (<, >, <=, >=, ==, !=) used to compare two values. These always evaluations to true or false. A true state is represented by a non zero value (1). A false state is represent by a zero value (0). All are binary operators. All the relational expressions are evaluated from left to right. Relational operators are at the same level of hierarchy i.e. no relational operators has the preference over the other.

Operator	Meaning	Expression
<	Less than	a<b
<=	Less than or equal to	a<=b
>	Greater than	a>b
>=	greater than or equal to	a>=b
==	equal to	a==b
!=	not equal to	a!=b

Example

	A<B	A<=B	A>B	A>=B	A==B	A!=B
A=10, B=20	1	1	0	0	0	1
A=10, B=10	0	1	0	1	1	0
A=20, B=10	0	0	1	1	0	1

40. What is the use of AND logical operator?

(FSD/G1-16) (SWL/G1-15)

Ans. The symbol used for AND operator is (&&). It is used to evaluate two conditions. It produces **true** if both conditions are **true**. It produces **false** result if any one condition is **false**. It is used to combine two or more relational expressions. It produces **true** if all the relational expressions (conditions) are true. It produces false result if any one of the conditions or all the conditions are false.

A(Exp-1)	B(Exp-2)	A && B
1 (True)	1	1 (True)
1	0 (False)	0 (False)
0	1	0
0	0	0

Example: X=10, Y=20, Z=30

A=X>Y	B=X>Z	A&&B
0	0	0
A=X<Y	B=X<Z	A&&B
1	0	0
A=X>Y	B=X<Z	A&&B
0	1	0
A=X<Y	B=X<Z	A&&B
1	1	1

41. Differentiate between unary and binary operator with example. (AJK/G1-14). (LHR-18)
- Ans. **Unary:-** The operator that operates only on operands is called unary operator. The examples of unary operators are ++ (increment operators), -- (decrement operator) and logical negation operator (!).
Binary:- The operator that operates on two operands (data items) is called binary operator. For example, in an expression "a + b", the operator '+' operates on two operands 'a' and 'b'. All arithmetic operators, relational operators and two logical operators (&& and ||) are example of binary operators.
42. Define logical operator. Give example. (FSD/G1-15) (SGD/G1-16) (AJK/G1-15) (RWP-18) (FSD/G1-15)
- Ans. Logical operators are used to evaluate compound conditions. The logical operators in C language are AND operator (&&), OR operator (||) and NOT operator (!).
43. Differentiate between standard and user defined identifiers Give example. (LHR-17) (KSMR/G1-16) (GUJ-17) (MTN/G1-16) (AJK/G1-16) (FSD-14) (AJK-19)
- Ans. There are two types of identifiers in C language. These are Standard Identifiers and user-defined Identifiers
Standard identifiers:- A type of identifier that has special meaning in C is known as **standard identifier**. C cannot use a standard identifier for its original purpose if it is redefined, printf and scanf are samples of standard identifiers. It is also called reserve words just like for, printf, main, getch, goto, if and switch.
User-defined identifiers:- The type of identifier that is defined by the programmer to access memory location is known as **user-defined identifier**. The user-defined identifiers are used to store data and program results. Some examples of user-defined identifiers are num, age, marks etc.

Programmes

- 1) What is the output of the following C-program? (GRW/G1-16)
- ```
main ()
{
 int a, b;
 a = 77
 b = 40
 a += b;
 printf ("a = %d", a);
}
```
- Output:**  
a = 117
- 2) Trace the errors of the following code. (GRW/G1-16)
- ```
int numb = 6,
++ number;
Printf ( %d\n , number);
```
- Correction:**
number = 6;
- 3) Trace the errors in the following program. (GRW/G1-16)
- ```
void main ();
{
 float a = 3.14;
 printf ("%d", a);
 getch ();
}
```
- Correction:**  
Semi column is not allowed here  
%f used because data type is float

- 4) Find the errors in the following code:-

(MTN/G1-16)

```
void main (void)
{
 Int x, y, z
 Z = x + y + z
}
```

**Correction:**

C is a case sensitive language so "int" is in small  
Every statement is terminated with the semi column symbol ;

- 5) Find out errors in the following code:

(SGD/G1-16)

```
include (stdio.h)(1)
Main (2)()
```

```
{
 charch (3)= 'a' ;
```

```
float 3digit (4) = 33.3;
```

**Correction:**

(1) <stdio.h> less than and greater than sign are used in header files

(2) C is a case sensitive language so "Main" is used like this "main"

(3) Space is used in declaration of data type like this "char ch"

(4) 3 digits is wrong declaration because variable is not declared with digit

- 6) Write the output of the following code: (SGD/G1-16)

```
int x = 3
printf ("% d ", x);
printf ("% d ", x + +);
printf ("% d ", + + x);
```

**Output:**

345

- 7) Find the error in the following code.

(RWP/G1-16)

```
include (conio.h)
include (stdio.h)
void main ()
```

```
{
 Printf ("OK");
}
```

**Correction:**

<conio.h>, <stdio.h> less than and greater than sign are used in header files

- 8) Find the output of the following code.

(RWP/G1-16)

```
Printf ("Pakistan \n is an \n Islamic \n country");
```

**Output:**

Pakistan  
is an  
Islamic  
country

- 9) Find the error in the following code(RWP/G1-16)

```
Main (1)();
{
 Printf (OK)(2)
}
```

**Correction:**

(1) C is a case sensitive language so "Main" is used like this "main"

Main function is not terminated with statement terminator

(2) Double code sign is not used in printf statement in the programme correct statement is this printf ("OK");

- 10) Trace the error.

```
int a = 6 (1)
++;
Printf (" %f" (2), a) (3)
```

**Correction:**

- (1) Every statement is terminated with the semi column symbol ;
- (2) %d is used because data type is int declare
- (3) correct statement is printf ("%d", a)

- 11) Write the given statement in C language.

 $Z = (3*y) (a^2+b^2)$ **Ans.** $Z = (3 \times y) \times (a \times a + b \times b) ;$ 

- 12) Find the error in program code:

```
int x = 20, y = 35 (1)
x = y + + (2)
y = ++y (3)
printf ("%d f %f \n", x, y);
```

**Correction:**

- (1), (2), (3) terminator is not used at the end of the statement
- (4) correct statement is (" %d f %d \n", x, y);

- 13) Trace the output of the following code.

```
include < Stdio.h>
Void main () {
int a, b, c ;
a = 10 ;
b = 20 ;
c = 50 ;
Printf (" %d %d %d", a + b, a * b, a - b) ;
getch () ; }
```

**Ans.**

30200 - 10

- 14) Trace the errors (GRW-14)

```
int b = 8 (1)
int c = 0 (2)
x = 0 (3)
printf ("%f" (4), x)
```

**Correction:**

- (1), (2), (3) terminator is not used at the end of the statement
- (4) correct statement is ("%d" <sup>(4)</sup>, x);

- 15) What is the value of Y after the following code executes:

```
float y = 3.4 + SQRT(25.0)
```

**Ans.**

8.4

- 16) Trace the error of the following code: (SWL-14)

```
void man ()
{
int A (1) = 10
Printf ("%d", a) (2)
```

**Correction:**

- (1) Correct statement is int a
- (2) Statement terminator is not used at the end of printf statement (;)

(MTN/G1-15)

(RWP/G1-15)

(AJK/G1-14)

(D.G.K/G1-15)

(LHR-14)



**17) Find error in following code segment (AJK-17)**

```
#include <stdio.h>
void main ()
{ int n = 6;
 ++ n (1)
 Printf (2)(" %d", n); }
```

**Correction:**

- (1) Statement terminator is not used at the end of printf statement (;)
- (2) p is small used in printf key word

**SECTION III****LONG QUESTIONS**

1. How many types of operators are available in C? Describe briefly. (RWP/G1-16)
2. What do you know about identifiers? Explain two types of identifiers. (LHR-18) (FSD-18)
3. What are logical operators? Discuss different logical operators with truth table. (SWL-19)
4. Describe different data types to store integer data in C-Language (SGD-19)
5. What is data type? Write in detail data types used in C-language. (D.G.K/G1-15) (RWP-19)
6. Write down the rules for naming variables in C-Language.  
(SWL-18)(MTN-18)(SWL-14)(GRW/G1-16) (GRW-18)



## INPUT/OUTPUT

## SECTION I

## MULTIPLE CHOICE QUESTIONS

**From PTB Exercise:**

1. The function `getche()` is defined in:  
(a) `stdio.h` (b) `string.h`  
(c) `math.h` (d) `conio.h`
2. The escape sequence for backslash is: (RWP-19)  
(a) `\` (b) `'b'`  
(c) `\\` (d) `'t'`
3. The format specifier `%u` is used for:  
(a) integer (b) unsigned short  
(c) unsigned float (d) unsigned long int
4. In c program, the number of columns that are printed are specified in  
(a) format specifier (b) field width specifier  
(c) formatting integers (d) both a and b
5. Escape sequence `\\` is used to print :  
(a) new line (b) backslash  
(c) space (d) tab
6. `scanf` function is used to input  
(a) numeric value (b) string value  
(c) both a and b (d) non of above
7. `scanf` function stores input value into  
(a) signature of a variable (b) constant  
(c) position of a variable (d) address of a variable
8. In statement `scanf("%f", &kilometer);` kilometer is a(n)  
(a) integer variable (b) float variable  
(c) string variable (d) double variable
9. `getch()` is a part of  
(a) `stdio` (b) `conio`  
(c) `math` (d) all of above
10. ASCII code for a character is displayed using format specifier  
(a) `%d` (b) `%f`  
(c) `%c` (d) `%x`

**From Punjab Board:**

1. The function that is used to display output on the screen is called: (LHR-19)  
(a) Printf (b) Scanf  
(c) Pow (d) Display
2. Format specifier used for string (GRW-19)  
(a) %f (b) %s  
(c) %c (d) %d
3. The general form of format specified for floating point value is: (D.G.K - 19)  
(a) %m.nf (b) m.n %  
(c) m % .nf (d) m.n %f
4. Format specifier is started with symbol: (AJK-19)  
(a) ? (b) &  
(c) % (d) \*
1. Which of the following format specifier is used for character data type. (SGD-18)  
(a) %d (b) %s  
(c) %c (d) %e
2. The escape sequence for back slash is. (SWL/G1-15)  
(a) \ (b) \b  
(c) \\ (d) \t
3. Which of the following escape sequence can be used to begin a new line in C? (SWL-18)  
(a) \a (b) \b  
(c) \m (d) \n
4. Getch ( ) stands for (D.G.K/G1-15)  
(a) Go character (b) Get character  
(c) Give character (d) All
5. Which escape sequence can be used to insert a tab in C? (D.G.K/G1-15)  
(a) \a (b) \b  
(c) \t (d) \n
6. An ampersand (&) before the name of a variable denotes: (LHR/G1-15)  
(a) Actual value (b) Variable name  
(c) Address (d) Data type
7. Which of the following function is used to input data in C-programs: (LHR/G1-15)  
(a) Scanf (b) Printf  
(c) Get (d) Sqrt
8. Format specifier is started with symbol:- (MTN-18)  
(a) ? (b) %  
(c) & (d) \*
9. The format specifier for float type data will be: (AJK/G1-14)  
(a) %d (b) %s  
(c) %f (d) %c
10. The escape sequence for new line is; (AJK/G1-15)  
(a) \r (b) \n  
(c) \b (d) \a
11. The function getch ( ) is defined in; (AJK/G1-15)  
(a) Stdio.h (b) String.h  
(c) Conio.h (d) Math.h

12. Which of the following is format specifier symbol; (AJK/G1-16)/ (AJK-17)  
(a) @ (b) &  
(c) \ (d) %
13. Which of the following format specifier is used for float data type; (AJK/G1-16)  
(a) %c (b) %d  
(c) %f (d) %s
14. Which of the following functions is used to read data from the file; (AJK/G1-16)  
(a) F puts ( ) (b) F scan f ( )  
(c) Put c ( ) (d) F printf ( )
15.  $A = A + B$  is equivalent to; (AJK/G1-16)  
(a)  $A = + B$  (b)  $A + = B$   
(c)  $B = + A$  (d)  $B + = A$
16.  $A + b =$  is equivalent to; (SGD-14)  
(a)  $b + = a$  (b)  $a = + b$   
(c)  $b = b + a$  (d)  $a = a + b$
17. Which of the following is equivalent to  $(P >= q)$ ? (SWL-14)  
(a)  $p < q$  (b)  $!(p < q)$   
(c)  $p > q$  (d)  $!(p > q)$
18. Which of the following function is used to write string to a file; (AJK-17)/(FSD-14)/ (LHR-14)  
(a) Puts ( ) (b) Put c ( )  
(c) f puts ( ) (d) f gets ( )
19. Which operator is called ternary operator; (AJK-17)  
(a) If (b)  $++$   
(c) ? (d) ( )
20. Relational operator allow you to; (AJK-17)  
(a) Compare (b) Add  
(c) Multiply (d) divide
21. The functions used for input and output is stored in:  
(a) Stdio.h (b) Conio.h  
(c) Math.h (d) Tan.h
22. How many variables can be used in one printf ( ) function? (D.G.K-14)  
(a) One (b) Two  
(c) Three (d) Many
23. Which function is used to input data in C-program? (GRW-14)  
(a) scanf ( ) (b) printf ( )  
(c) sqrt ( ) (d) pow ( )
24. Another term used for a conditional operator is: (GRW-14)  
(a) unary (b) binary  
(c) ternary (d) iteration
25. \_\_\_\_\_ is not a valid escape code: (LHR-14)  
(a) \f (b) \r  
(c) \y (d) \f

26. The escape sequence for back slash is: (LHR-14) (SWL-14)  
 (a) \ (b) \b  
 (c) \y (d) \t
27. Which of the following functions is used to write character to a file: (RWP-14)  
 (a) putc ( ) (b) fputo ( )  
 (c) fget ( ) (d) getc ( )
28. Which format specifier is used for character data type? (RWP-14) (SGD-14)  
 (a) %S (b) %c  
 (c) %d (d) %f
29. Which escape sequence can be used to begin a new line in C? (SWL-14)  
 (a) \a (b) \b  
 (c) \n (d) /m
30. Which of the following character is used to make the end of the string? (SWL-14) (FSD-19) (LHR-19)  
 (a) \o (b) /o  
 (c) \a (d) \n
31. The ASCII code for escape key is: (SGD-14)  
 (a) 27 (b) 28  
 (c) 29 (d) 30

## SECTION II

### SHORT QUESTION ANSWERS

#### From PTB Exercise:

1. What do we mean by standard input and output? Illustrate the use of printf() and scanf() functions. (LHR-19)

Ans. The int scanf(const char \*format, ...) function reads the **input** from the **standard input** stream stdin and scans that **input** according to the format provided. The int printf(const char \*format, ...) function writes the **output** to the **standard output** stream stdout and produces the **output** according to the format provided. (or) Data Input and Output functions in C. For data input and output, C provides a collection of **library functions** such as getchar, putchar, scanf, printf, gets and puts. These functions enable the transfer of data between the C program and standard input/output devices.

2. Illustrate the difference between format specifiers and field-width specifiers with example.

Ans. **Format specifier** is used during input and output. ... Some **examples** are %c, %d, %f, etc. The **format specifier** in printf() and scanf() are mostly same but there are some **difference** which we will see. ... **field width** to be printed if the characters are less than the size of width the remaining space ... Character **format specifier** : %c.

3. Define the term 'escape sequence' list name and uses of any five escape sequence.

Ans. In C, all escape sequences consist of two or more characters, the first of which is the **backslash**, \ (called the "Escape character"); the remaining characters determine the interpretation of the escape sequence. For example, \n is an escape sequence that denotes a newline character.

- \a 'a' stands for alarm. It causes a beep sound in the computer.
- \b 'b' stands for backspace. It moves the cursor one space back. For example a statement is given below using '\b' escape sequence.  

```
printf("Welcome\b");
```
- \r 'a' stands for return. It moves the cursor to the beginning of the current line.
- \f 'f' stands for feed. It causes the output to feed one paper on the printed attached to the computer.
- \\ It is used to display a backslash character in the output. For example, a statement is given below using '\\' escape sequence.

4. Show the output displayed by the following program when the data entered are 10 and 15.

Ans. #include <stdio.h>

```
void main ()
{
 int m,n;
 printf("enter two numbers (separated by coma):");
 scanf("%d%d",&m,&n);
 m = m + 10;
 n = 5*m;
 printf("m = %d\t n = %d\n",&m,&n);
}
```

**OUTPUT**

m=20

n=100

5. Show how the value -17.246 would be printed using the formats %8.4f,%8.3f,%8.2f,%8.1f,%8.0f, and %0.2f.

Ans. **Output as the value is give i.e. -17.246**

Format output

|       |         |
|-------|---------|
| %8.4f | -17.246 |
| %8.3f | -17.246 |
| %8.2f | -17.246 |
| %8.1f | -17.246 |
| %8.0f | -17.246 |
| %2f   | -17.24  |

Note; denote a single blank space □

6. Assuming x (type double) is 21.335 and y (type int) is 200, show the output of the following statements (on paper). For clarity, use the symbol □ to denote a blank space.

Ans. Printf("x is %6.1f\t y is %4d\n",x,y);

Printf("y is %d\n", y);

Printf("x is %1f\n",x);

**OUTPUT**

X is □21.34

Y is 200

X is 21.3

7. If the variables a, b and c are 360, 408.56 and -12.31, respectively, write a statement that will display the following line: (for clarity, the symbol □ shows a blank space)

Ans. □□307 □□□□408.56 □□□□-12.31

Printf("%5d%11.2f%8.1f",a,b,c);

8. write the program that asks the user to enter the radius of the circle that computes and display the circle's area. Use the formula

Ans. area =  $\pi \times \text{radius} \times \text{radius}$

# define PI 3.14159

void main()

{

float r, area;

clrscr();

printf("Please Enter the Radius of the Area");

scanf("%f",&r);

area=PI\*r\*r;

printf("Area= %f", area);

getch();

}



**OUTPUT**

Please Enter the Radius of the Area 8

Area= 201.06

Where PI is the constant value of 3.14159

(Note : Define a constant macro PI with # define directive)

Write a program that stores the values 'A', 'U', 3.456E10 and 50 in separate memory cells. Your program should get the first three values as input data, but use an assignment statement to store the last value.

```
#include <stdio.h>
#include <conio.h>
void main (void)
{
 char ch1,ch2;
 float x;
 int n;
 clrscr();
 printf("Enter first character ? ");
 scanf("%c &ch1);
 printf("Enter second character ? ");
 scanf("%c &ch2);
 printf("Enter real value ? ");
 scanf("%f &x);
 n = 50;
 printf("First character is: %c \n", ch1);
 printf("Second character is: %c \n", ch2);
 printf("Real value is: %3E \n", x);
 printf("Integer value is: %d \n", n);
```

```
Output of the Program
Enter first character ? A
Enter second character ? U
Enter real value ? 3.456E10150
First character is: A
Second character is: U
Real value is: 3.456E10
Integer value is: 50
```

9. Write a program that converts a temperature in degrees fahrenheit to degree celsius. For conversation statement to store the last value.

Ans. Celsius = 5/9 (fahrenheit-32)

```
Void main()
{
 float c,f;
 clrscr();
 printf("Please Enter Fahrenheit:");
 scanf("%f", &f);
 c = 5/9*(f-32);
 printf("Temperature In Celcius Is = %f",c);
 getch();
}
```

**OUTPUT**

Please Enter Fahrenheit 77

Temperature In Celcius Is = 25

**From Punjab Board:**

1. List some important functions for output.

(GRW-14) (SWL-14)

Ans. C-language also provides various standard output library functions. These functions are display the output of program on the display screen. These are important and commonly used standard output functions in C language are as follows:

- > printf()
- > puts()

2. Write C statement to input value of integer variable x. (MUL-17)

Ans. Learn about how to print a integer entered by user in C programming with source ... printf("Enter an integer: "); // scanf() reads the formatted input and stores them ... Finally, the value stored in the variable number is displayed on the screen ...

```
int x;
scanf("%d",&x);
```

3. Discuss the purpose of % C format Specifier.

(SGD-14) (AJK-17) (BWP-14)

Ans. The %c format specifier is implemented for representing characters. This is used with printf() function for printing the character stored in a variable. When you want to print a character data, you should incorporate the %c format specifier.

4. What is an escape sequence? give example.

(D.G.K/G1-16) (RWP/G1-15) (RWP/G1-15) (FSD-14) (AJK/G1-15)  
(GRW-14) (D.G.K.-17) (MUL-18) (RWP-18) (SGD-17) (BWP/G1-16) (AJK-19)

Ans. In C, all escape sequences consist of two or more characters, the first of which is the backslash, \ (called the "Escape character"); the remaining characters determine the interpretation of the escape sequence. For example, \n is an escape sequence that denotes a newline character. \t for tab(8-spaces)

5. What is standard input? (GRW-14) (FSD-14) (LHR-17) (RWP-14) (LHR/G1-16) (D.G.K - 19)

Ans. Standard input" refers to a specific input stream, which is tied to file descriptor 0. It's the stream from which scanf, get char, gets (which you should never use), etc., all read. Basically, any studio input function that doesn't take a FILE \* as an argument is reading from standard input.

6. What is the use of "scanf" function? Also write its syntax. (KSMR/G1-16) (RWP/G1-15) (RWP/G1-15) (AJK/G1-16) (GRW-14) (LHR/G1-15)

Ans. The program will read in an integer value that the user enters on the keyboard (%d is for integers, as is printf, so b must be declared as an int) and place that value into b. The scanf function uses the same placeholders as printf: int uses %d. float uses %f.

Syntax:- scanf("%d",&a);

7. Why is the ampersand (&) used in scanf function? (AJK/G1-14) (SGD/G1-16) (GUJ-17) (SWL-18)

Ans. The ampersand is the address-of operator. The scanf function is going to fill a buffer, and it needs to know its address so that it can write to it. ... The scanf() function reads formatted input and has to put this input into something like a variable. OR

In C, & is actually the address of operator in scanf the address of operator (&) tells the scanf functions the address of variable where the input value is to store.

8. Differentiate between printf() and scanf() functions.

(FSD-14) (D.G.K/G1-15) (D.G.K-19) (MTN-19) (GRW-19)

Ans. Two commonly used functions for I/O (Input/Output) are printf() and scanf(). The scanf() function reads formatted input from standard input (keyboard) whereas the printf() function sends formatted output to the standard output (screen).

9. Give one example of control string in printf function. (LHR/G1-16) (MTN/G1-16) (D.G.K - 19)

Ans. This function is used to print any text as well as value of the variables on the standard output device/Console (monitor screen), printf is very basic library function in c language that is declared in stdio.h header file.

The printf() function prints all types of data values to the console

It requires format conversion symbol or format string and variable names to print the data. The format string symbol

**Syntax:**

- printf("message text");
- printf("message text+ format-string",variable-list);
- First printf() style print the simple text on the monitor, while second
- printf() prints the message with values of the variable list.
- String (char[]) %s
- Use 'u' for unsigned type modifier, 'l' for long.

**EXAMPLE: USE OF PRINTF()**

```
#include <stdio.h>
int main()
{ printf("C language");
 printf(" Apps");
 printf("Bits of computer");
 return 0;
}
```

**Output**

language Apps Bits of computer

**10. What is the use of printf() function? Write the syntax of printf() function.**

(D.G.K/G1-16) (AJK/G1-15)

**Ans.** This function is used to print any text as well as value of the variables on the standard output device/Console (monitor screen), printf is very basic library function in C language that is declared in stdio.h header file.

The printf() function prints all types of data values to the console.

It requires format conversion symbol or format string and variable names to print the data. The format string symbol

**Syntax:**

- printf("message text");
- printf("message text+ format-string",variable-list);
- First printf() style print the simple text on the monitor, while second
- printf() prints the message with values of the variable list.
- String (char[]) %s
- Use 'u' for unsigned type modifier, 'l' for long.

**EXAMPLE: USE OF PRINTF()**

```
#include <stdio.h>
int main()
{ printf("C language");
 printf(" Apps");
 printf("Bits of computer");
 return 0;
}
```

**Output**

language Apps Bits of computer

**11. Compare getch () and getche () functions.**(FSD/G1-16)(SWL/G1-15)(RWP/G1-16)(RWP-14)(FSD/G1-15)(BWP-14)  
(AJK/G1-16) (SGD-14) (LHR-18) (KSMR/G1-16) (AJK/G1-14)**Ans.**

1. The last e in getche() fn stands for echo, which means output to the console/stdout.
2. getch () doesn't displays the character and terminates the program but getche () displays the character and then terminates the program. For eg if we press g then in getch () g will be displayed and program is terminated.
3. getch() and getche() are not compiling.
4. getche() give output without any buffer but the getch() give output with buffer. Like getch(), this is also a non-standard function present in conio.h. It reads a single character from the keyboard and displays immediately on output screen without waiting for enter key.

12. List any two important functions for input in C. (BWP/G1-16) (RWP/G1-16)

**Ans.** Input means to provide the program with some data to be used in the program. C programming language provides many built-in functions to read any given input and to display data on screen when there is a need to output the result. All these built-in functions are present in C header files, we will also specify the name of header files in which a particular function is defined while discussing about it.

Examples `scanf()`, `getchar()`, `putchar()`, `gets()` & `puts()` functions

13. List out the names of functions used for character input? (FSD/G1-16)

**Ans.** `putchar()` function is a file handling function in C programming language which is used to write a character on standard output/screen. `getchar()` function is used to get/read a character from keyboard input.

14. Give single print statement to print the following message 'I Love Pakistan' (D.G.K-14)

**Ans.** `printf("I Love Pakistan");`

15. Describe `clrscr()` function. (D.G.K-14) (LHR-14) (BWP-17) (BWP-15)

**Ans.** `clrscr()` It is a predefined function in "conio.h" (console input/output header file) used to clear the console screen. It is a predefined function, by using this function we can clear the data from console (Monitor).

16. Differentiate between `gets()` and `puts()` Functions.

**Ans.** The main difference between `gets` and `puts` in C Language is that `gets` is a function that reads a string from standard input while `puts` is a function that prints a string to the standard output.

Actually these are two different things. As `gets()` is used to read strings which may include white spaces also. And `puts()` is used to print this string. You might ask for difference between `gets()` and `getchar()`.

17. Define standard output. (FSD/G1-16) (SWL-14) (FSD/G1-15) (GRW/G1-16) (LHR-19)

**Ans.** When we say Output, it means to display some data on screen, printer, or in any file. C programming provides a set of built-in functions to output the data on the computer screen as well as to save it in text or binary files. examples of output functions are `puts()` and `printf()`

18. List some important function for input. (MTN/G1-15)

**Ans.** Data Input and Output functions in C. For data input and output, C provides a collection of library functions such as `getchar`, `putchar`, `scanf`, `printf`, `gets` and `puts`. These functions enable the transfer of data between the C program and standard input/output devices.

19. Give single print statement to print the following message. (LHR-14)

**Ans.** `printf("No. of employees.\n No. of officers.");`

20. Define the format specifier used in `printf()` and `scanf()` functions. (GRW/G1-16) (FSD/G1-16) (GUJ-18) (SGD/G1-16) (MTN/G1-16)

**Ans.** In C, format specifiers defines the data type to be displayed at the output. It tells the compiler that which type of value you want to print or read `scanf("%d",&a);`. Take below example: `printf("%d",a);` and `scanf("%d",&a);`. Here %d is format specifier and it tells the compiler that we want to print an integer value that is present in variable a.

21. Write the use of format specifiers in C-language with four examples (Types). (GRW/G1-16) (FSD/G1-16) (GUJ-18) (SGD/G1-16) (MTN/G1-16) (RWP/G1-15) (RWP-14) (MTN/G1-15) (RWP-17) (SGD-18)

**Ans.** The % specifiers that you can use in ANSI C are:

- `%d` Usual variable type Display.
- `%c` char single character.
- `%d` (%i) int signed integer.
- `%e` (%E) float or double exponential format.
- `%f` float or double signed decimal.
- `%g` (%G) float or double use %f or %e as required.
- `%o` int unsigned octal value.
- `%p` pointer address stored in pointer.

22. Which symbol is used to start format specifier? (SWL-14)

Ans. The % specifiers that you can use in ANSI C are:

- Usual variable type Display.
- %c char single character.
- %d (%i) int signed integer.
- %e (%E) float or double exponential format.
- %f float or double signed decimal.
- %g (%G) float or double use %f or %e as required.
- %o int unsigned octal value.
- %p pointer address stored in pointer.

23. What is the use of field width specifier, getch() and getch() ?

(BWP-14) (SGD-14) (LHR/G1-15) (SGD-19)

Ans. getch():

getch() is a nonstandard function and is present in conio.h header file which is mostly used by MS-DOS compilers like Turbo C. It is not part of the C standard library. Like above functions, it reads also a single character from keyboard. But it does not use any buffer, so the entered character is immediately returned without waiting for the enter key.

**In Field width specifiers are format specifiers** used in programming languages for printing formatted output on the screen. **Field width** as the name suggests is the **width** or space which needs to be given for that particular display line and it is specified using numbers after % this symbol.

getche()

Like getch(), this is also a non-standard function present in conio.h. It reads a single character from the keyboard and displays immediately on output screen without waiting for enter key.

### Programmes

1. Find the errors in the following code:

(BWP/G1-16)

```
{
int n = 6 ;
n ++ , (1)
printf (" % d (2) , n) ;
}
```

**Correction**

(1) ; is not used at the end of line.

(2) " " is not used in printf statement.

2. Trace the output.

(SGD-14)

```
int i = 10;
printf ("%3d", i);
```

Ans.

□10

3. Trace the output.

(BWP/G1-16)

```
int number = 6;
int x = 0;
x = -- number;
Printf (" % d " , x);
```

Ans.

5

4. Trace out errors in the following code:

```
float r (1)
clrscr () (1)
printf (" enter radius (2));
scanf (" % f " , (3) r);
```

**ERRORS**

- (1) Statement terminator; is missing;
- (2) " is not used in the printf
- (3) , is used not ; is used
- (4) & is not used just like &r in scanf

5. Determine the output of the following code:

```
int i = 515 ;
printf (" i = % 5 d " , i);
printf (" i = % d " , i);
```

**Output**

```
i = □ □ 515
i = 515
```

6. What is the output of the following?

```
include < stdio.h >
include < conio.h>
void main)
{
 int i = 5 ;
 printf(" % d % d % d " , i , i + + , + i);
 getch ();
}
```

Output

```
766
```

7. Trace out errors in the following code.

```
include < stdio.h >
include < conio.h>
void main < > (1)
{
 float area, r = 2.5;
 area = 3.14 x (2) r x (2) r;
 printf(" Area = % f " , area);
 getch ();
}
```

**Correction:**

- (1) ( ) Small brackets are used not < >.
- (2) \* operator is used instead of x

8. Trace the error in the following code.

```
include < stdio.h >
void main ()
{
 Int A (1) = 10 ;
 Printf (2) (" % d " , a);
}
```

**Correction:**

- (1) A capital use correct is int a
- (2) printf is used not Printf

(BWP/G1-16)

(BWP/G1-16)

(KSMR/G1-16)

(KSMR/G1-16)

(KSMR/G1-16)



9. Predict the output of the following code.

(KSMR/G1-16)

```
void main ()
{
 char ch1 = 'A', ch2 = 'B', ch3 = 'C';
 printf(" ch2 \n % c \n ch1 ", ch3);
 getch ();
}
```

**Output**

```
ch2
C
ch1
```

10. What is output of : printf ("%C" ,67); (MUL-18)

**Output**

B

11. Trace the output of the following code.

(D.G.K/G1-15)

```
include < Stdio.h >
void main () {
 char word 1, word 2, word 3;
 word 1 = 'R' ;
 word 2 = 'M' ;
 word 3 = 'P' ;
 printf (" % c % c word3" , word 1, word 2);
 getch () ; }

```

**Output**

RMword 3

12. Trace the error in the code. (D.G.K/G1-15)

```
include < STDIO.H" (1)
void Main (2) () {
(4)X = 20 ;
(4)Y = 30 (3)
(4)Z = x + y(5)
Printf (" % f % f" , x, z);
getch () ; }

```

**Correction:**

- (1) Correct is <stdio.h>
- (2) void main ()
- (3) Statement terminator; is missing;
- (4) Not declare the data types
- (5) Statement terminator; is missing;

13. Trace the error. (AJK/G1-14)

```
void main () (1)
{ int R (2) ; r = 25 ;
if (R >= 0)
{ R = R * 3.14 * 3.14;
}; (3)

```

**Correction:**

- (1) , is not used here
- (2) R variable declare in small letters not in upper case
- (3) ; is not used here

**14. Trace the error.**

```
(1) integer x = 15;
printf ("%d", x);
Correction:
```

- (1) int keyword is used not use integer.  
(2) %d is missing just like printf ("%d", x);

**15. What is the output of the following code?**

```
printf ("Hello \n world \n Pakistan");
```

**Output**

```
Hello
world
Pakistan
```

**16. Find the output of the following code.**

```
include < stdio.h >
include < conio.h>
void main ()
{
 int x, y, z ;
 printf(" Enter three number = ");
 scanf (" %d %d %d ", &x , &y , &z);
 y = x + z ;
 x = z + 1;
 printf(" %d %d %d ", x , y , z);
 getch () ;
}
```

**Output**

```
Enter three number = 5 7 9
127325321272
```

**18. Determine the output of the following code segments.**

```
include < stdio. h>
include < conio. h>
void main ()
```

```
{
 float f = 3.14159;
 printf ("f= % 3.2", f);
 getch () ;
}
```

**Output**

```
f = % 3.2
```

**19. Find the output of the following code.**

```
include < stdio. h>
include < conio. h>
void main ()
{
 int a, b, c;
 a = 5 ; b = 10;
 c = a + b;
 printf ("the sum of a+b = %d", c);
}
```

**Output**

```
the sum of a + b = 25
```

(RWP/G1-15)

(AJK/G1-14)

(KSMR/G1-16)

(GRW/G1-16)

(GRW/G1-16)

20. Predict the output of the following code:

```
int number = 2;
number ++;
printf (" % d ", number);
```

Output

3

(FSD/G1-16)

21. Trace the output of the following code:

```
int a,
print (" % d ", a);
```

Output:

-28765

(FSD/G1-16)

22. Find the error in the following code:

```
main () ;(1)
{
 printf ("Hellow(2)");
}
```

Correction:

(1) ; is not used here

(2) " is missing in printf

(FSD/G1-16)

23. Trace the output of the following code:

```
void main ()
{
 printf (" 22\t");
 printf (" 666");
}
```

Output

22 666

(FSD/G1-16)

24. Trace the output:

```
{
 printf (" * \n ** \n *** \n ");
}
```

Output

\*
\*\*
\*\*\*

(FSD/G1-16)(SGD-17)

25. Trace out the errors from the following code segment:

```
main ()
{
 int a, b;
 a = 77;
 b = 40;
 a += b(2)
 print(1) ("a = %d", a);
}
```

Correction:

(1) <sup>(1)</sup>printf keyword is used instead of print

(2) Statement terminator; is missing;

(FSD/G1-16)

26. Trace out the errors from the following code segment

(FSD/G1-16)

```
#include (stdio.h)(1)
void main ()
{
 float n;
 scanf("%d" (2), &n);
 printf("The number is % f", n);
}
```

Correction:

(1) Using <> sign

(2) Use %f because data type is float in scanf ( ) %d escape sequence is only for int data type

27. Trace the output of the following code:

(FSD/G1-16)

```
int a = 9;
a = a % 4;
printf ("\\n %d is result" , a);
```

Output

1 is result

28. Trace the output of the following code:

```
void main ()
{
 printf ("55\\t");
 printf ("666");
 printf ("\\n 777");
}
```

Output

55 666  
777

29. Identify the errors in the following lines.

(D.G.K/G1-16)

<sup>(1)</sup>integer      A <sup>(2)</sup> = 2 + 3 ;

<sup>(3)</sup>Float      B = 5 ;

int    C = A + B ;

Correction:

(1) int keyword is always used not integer.

(2) Use small variable names just like a, b;

(3) Use small (lower case) float

30. Trace the errors in the following cod.

(D.G.K/G1-16)

```
#include <std.n(1)>
void main (void)
{
 printf ((3) " Pakistan " (3) (2))
}
```

Correction:

(1) not correct Header file correct is <stdio.h>

(2) Statement terminator; is missing;

(3) This " is used instead of \* in printf()

31. Predict the output of the following(D.G.K/G1-16)

```
main ()
{
 int x = 1 ;
 x ++ ;
 printf (" % d " , x ++) ;
}
```

Output

2

32. Predict the output of the following code.

```
void main (void)
{
 printf (" Hello \ " World " ");
}
```

**Output**

Error occur unterminated string function called missing

33. Trace output: (LHR/G1-16)

```
int n = 6;
n ++ ;
printf (" % d", n);
```

**Output**

7

34. Trace error:

```
int x(1) = 5;
y = x + 3(2)
int y(3)
printf (" % d", y);
```

Correction:

- (1) Space is used between int x;
- (2) Statement terminator; is missing
- (3) Statement terminator; is missing

35. Trace error: (LHR/G1-16)

```
Include < STDIO. (1)H >
VOID (2) Main (); (3)
{
 printf ("Pakistan");
}
```

Correction:

- (1) # include <stdio.h> correct header file is this
- (2) must be written in small letters
- (3) ; is not used here, nothing any symbol is used at the end of main function

36. Trace output: (LHR/G1-16)

```
int a = 512;
printf (" a = % 5d", a);
printf (" a = % 1 d", a);
```

**Output**

□ □ 512512

37. Trace output: (LHR/G1-16)

```
main ()
{
 printf (" 55");
 printf (" 5855");
}
```

**Output**

55 5855

38. Trace the error in the following code.

```
include < stdio.h >
void main (void)
Print (1)("Hellow world") (2)
```

(SWL/G1-15)

**Correction:**

- (1) printf() is used instead of Print
- (2) Statement terminator; is missing;

**39. Trace the output.**

```
include < stdio.h >
```

```
Void main (void) {
```

```
int x = 10;
```

```
printf (" % ", x % 2);
```

**Output:**

|   |
|---|
| % |
|---|

**40. Trace the output.**

(SWL/G1-15)

```
include < stdio.h >
```

```
Void main (void)
```

```
{
```

```
int x = 1;
```

```
int y = 2 ;
```

```
x = x + 1;
```

```
y = y + x;
```

```
Printf (" % d \ n % d ", x , y);
```

```
}
```

**Output**

|   |
|---|
| 2 |
| 4 |

**41. Trace the errors in the following code.**

(SWL/G1-15)

```
include < stdio.h >
```

```
(1)Void main (void)
```

```
{
```

```
int x = 4 (2)
```

```
(3)y = x + 10 (2)
```

```
printf (" % d ", x + y);
```

```
}
```

**Correction:**

- (1) void is "V" is in capital
- (2) ; is not used at the end of statement
- (3) y data type is not mentioned

**42. Trace out errors in the following code:**

(SGD/G1-16)

```
float area, r (2)
```

```
printf ("Enter Radius(1));
```

```
scanf (" % f" (4); &r) (2)
```

**Correction:**

- (1) " is not used in printf
- (2) Statement terminator; is missing;
- (3) & is not used with r in scanf just like scanf ("%f", &r);
- (4) In scanf() , is used instead of ;

**43. Write the output of the following code:**

(SGD/G1-16)

```
printf (" Programming \t is \n very \t interesting");
```

**Output**

Programming

Very interesting



44. Write the output of the following code:

```
int x, y, z, r;
printf ("Enter three numbers.");
scanf ("%d %d %d", &x, &y, &z);
r = x + y * z;
printf ("%d", r);
```

Enter three number 3 4 5

23

SGD/G1-16

45. Trace the errors in the following code:-

```
#include (stdio.h)
(2)Void main (void); (3)
```

```
{
 (4)x = 5;
 (4)y = 6;
 (4)z = x + y;
 printf ("%d", z); }
```

(MTN/G1-16)

**Correction:**

- (1) Header file is not correct with use ( ) , the correct is <stdio.h>
- (2) V is capital
- (3) main always write without; statement terminator
- (4) not declare data type of x, y, z

46. What is the output of the following code?

```
printf ("I\n Love\n programming");
```

**Output**

I

Love

Programming

(MTN/G1-16)

47. Find the output of following code segment:-

```
int x = 10;
int y = 5;
int z = x + y;
printf ("%d%d%d", x, y, z);
```

**Output**

10515

(MTN/G1-16)

48. Trace the output of the following code.

```
#include <stdio.h>
void main ()
{
 char word 1, word 2, word 3;
 word 1 = 'R';
 word 2 = 'M';
 word 3 = 'P';
 printf ("%c %c %c word3", word 1, word 2);
 getch (); }
```

(D.G.K/G1-15)

**Output**

RMword3

## 49. Trace the error in the code. (D.G.K/G1-15)

```
include < STDIO.H(1)
(2)Void Main () {
(3)X = 20 ;
(3)Y = 30 (4)
(3)Z = x + y (4)
(2)Printf (" % f % f", x, z);
```

```
getch () ; }
```

**Correction:**

- (1) Header file write is wrong correct is <stdio.h>
- (2) V & P is capital
- (3) Data type is not mention with x, y, z
- (4) ; is not used at the end of statement

## 50. Trace the error (AJK/G1-14)

```
void main(),(1)
{ int (2)R ; r = 25 ;
if (R(3) > = 0)
{ R = R * 3.14 * 3.14;
};(4)
```

**Correction:**

- (1) after main not write any sign
- (2) R is capital
- (3) R is capital
- (4) After delimiters not use any sign.

## 51. Trace the error. (RWP/G1-15)

```
(1)Integer x = 15;
printf (" %(2) ", x);
```

**Correction:**

- (1) integer is not any data type. Here is int is used
- (2) Format specifies % d is not used in printing statement

## 52. Trace the output. (RWP/G1-15)

```
printf ("Book\t reading \n");
printf ("is a\t very good");
printf ("Habbit");
```

**Output**

Book      reading  
is a      very good Habbit

## 53. Predict the output of the following code.

(AJK/G1-14)

```
printf (" * \n * * \n * * * \n");
}
```

**Output**

```
*
**

```

## 54. Find the error in the code:

(FSD-14)

```
#include <STDIO.H>(1)
void main () {
 int a;
 a = 20;
 printf("%f" (2), a);
 getch () ;}
```

**Correction:**

- (1) stdio is write in upper case that is wrong.  
(2) Correct format specifier is %d is used, because data type is int .

## 55. Trace the output.

(FSD-14)

```
void main () {
 int a, b, n;
 n = 400;
 a = n % 100;
 b = n / 10;
 n = n % 10;
 printf("%d\n", a);
 printf("%d\n", b);
 printf("%d\n", n);
 getch () ;}
```

**Output**

```
0
40
0
```

## 56. Trace the output:

(FSD-14)

```
void main () {
 int x, y, z;
 x = 10;
 y = 20;
 z = 30;
 x = x + y;
 y = y + z;
 z = x - y;
 printf("Result = % d % d\n", x, y, z);
}
```

**Output**

```
Result = 3050
```

## 57. Write the output of the following code.

(D.G.K-14)

```
main ()
{
 printf ("444 \n");
 printf (" 44 ");
}
```

**Output**

```
444
 44
```

## 58. Trace the errors in the following programs.

<sup>(1)</sup>include < stdio.h>

```
main ()
{
 int a = 10;
(2)Printf (" % d ", a); }
```

**Correction:**

- (1) Missing the # sign  
(2) p used here upper case C-language is case sensitive, so all source code is write in small letter.

59. Determine the output of the following code:

(BWP-15)

```
int a, b, c,
a = 3;
b = 3;
c = a % b;
printf(“ % d”, c);
```

**Output**

0

60. What is the output of the following code?

(BWP-15)

```
int b = 9;
```

```
b = $\frac{b}{2}$;
```

```
Printf (“ % d” , b);
```

**Output**

4

61. Predict the errors in the following code.

(BWP-15)

```
include < stdio . h > ;(1)
```

```
(2)Void Main ()
```

```
{ int a = 10 ;
```

```
printf (“% d”(3) ,) ;
```

```
}
```

**Correction:**

(1) After header file ; is not used

(2) V & M is capita letter

(3) Output variable is not mention like this printf (“%d”, a);

62. Predict the errors in the following code.

(BWP-15)

```
include < stdio . h >
```

```
(1)Main () ; (2)
```

```
{
```

```
print(3)(“% d” , a, b); }
```

**Correction:**

(1) in main write in M is in small letters

(2) ; is not use here and main always write without;

(3) f is missing in printf()

63. Trace out the errors from the following code segment:

(FSD/G1-15)

```
main ()
```

```
{
```

```
int a, b;
```

```
a = 77;
```

```
(1)c = 40;
```

```
a += b;
```

```
print(2)(“a = %d” , a);
```

```
}
```

**Correction:**

(1) c is not declare the data type

(2) print is not any function in C-Lang correct5 function is printf ();

64. Trace out the errors form the following code segment.

(FSD/G1-15)

```
include (stdio.h)(1)
void main ()
{
float n;
scanf("%d", &n);
printf("The number is % f", n);
}
```

**Correction:**

- (1) In header file always <> symbol are used.  
(2) In printf (%d) format specifier is used that is wrong because float data type is declare with "n" so here %f format specifier is used in printf ( ).

65. Trace the output of the following code:

(FSD/G1-15)

```
int a = 9;
a = a % 4;
printf("\n% d is result", a);
```

**Output**

1 is result

67. Correct the following statement. (LHR/G1-15)

<sup>(1)</sup>Printf (Hello World);

**Correction:**

printf ("Hello world");

68. Write down output of the following code:

(LHR/G1-15)

```
printf (" C programming is \t easy");
```

**Output**

C programme is      easy

69. Write down output of the following code:

(LHR/G1-15)

```
int χ = 103;
printf (" %f", χ);
```

**Output**

In printf floating point format not link because data type is int, so we use format specifies is %d used and secondly floating point formats not linked so abnormal programm termination

70. What will be the output of the following?

(RWP-14)

```
int a = 9;
a = a % 4;
printf("%d", a);
```

**Output**

1

71. What will be the output of the following?

(RWP-14)

```
float f = 3.24;
printf("value = %f", f);
```

**Output**

value = 3.24

72. Detect the errors from the following code;

```
int number = 6(2)
number ++;
printf(" %f(1)", number);
```

**Correction:**

- (1) Here %d is used because data type is declare int
- (2) Statement terminator; is missing;

73. Find error in the following code; (AJK/G1-15)

```
(1)Main ();(2)
{
 Print (" I love Pakistan(3)");
}
```

**Correction:**

- (1) No any sign is used after main ( )
- (2) M is capital
- (3) " is missing in printf()

74. What is output in the following program:

```
#include < stdio.h >
main ()
{
 int a = 10 ;
 printf (" %d ",a)
}
```

**Output**

10

75. Trace the output:

```
#include < stdio.h >
void main ()
{
 printf ("%c ", " pak ");
}
```

**Output**

If we use %c show that "result "pak" = !. If we want to print pak, then we use format specified is (%s) in printf( )

76. Trace output of the following:

```
float l = 4.17894;
printf (" %2.2f", l);
```

**Output**

4.18

77. Trace the output

```
void main ()
{
 printf ("55 \t ");
 printf ("666 ");
 printf (" \n777 ");
}
```

**Output**

55    666  
777

(AJK/G1-15)

(AJK/G1-15)

(AJK/G1-15)

(SWL-14)

(SWL-14)



78. Trace the error in the following C code.

(BWP-14)

```
#include <stdio.h>
void main ()
{
 (2)print(" C Language is interesting ");
}
```

Correction:

- (1) `stdio.h` is correct  
(2) `print` is function is not used in C language correct is `printf ( )`

79. Trace the output:

(BWP-14)

```
int i = 20 ;
printf ("% 6d" , i) ;
```

Output

□ □ □ □ 20

80. What will be the output of the following:

(BWP-14)

```
float t = 3. 8961;
printf ("value of t = % 6.2 f" , t) ;
```

Output

value of t = □ □ 3.90

81. Find errors in following code

(AJK-17)

```
(1)include < stio.h >(2)
void main ()
{
 (3)inti = 10 ;
 (4)Printf (" %d" , i) ;
}
```

Correction:

- (1) `#` is missing  
(2) Correct is `stdio`  
(3) Space is missed between `inti`  
(4) `p` is capital letter, it is use in lower case

82. Predict the output of the following code.

(AJK-17)

```
in i = 212 ;
Printf (" i = %5d" , i) ;
```

Output

i = □ □ 212

83. What is the output of the following?(AJK/G1-16)

```
include <stdio.h>
include <conio.h>
void main ()
{
 int i = 5;
 printf ("% d % d % d " , i , i + + , + + i)
 getch ();
}
```

Output

766

84. Trace out errors in the following code.

(AJK/G1-16)

```
include <stdio.h>
include <conio.h>
(1)void main <>(2)
{
 float area, r = 2.5;
 area = 3.14 x(4) r x(4) r;
 printf ("Area = % i(3)" area);
 getch ();
}
```

**Correction:**

- (1) V is capital that is wrong
- (2) ( ) Brackets are used not <> Brackets are used here.
- (3) %f is used in printf ( ) because data type of "area" is float.
- (4) The \* operator is used for multiplication not use X in equation

85. Trace the error in the following code.

(AJK/G1-16)

```
include <stdio.h>
void main ()
{
 int A(1) = 10;
 (2)Printf (" % d" , a);
}
```

**Correction:**

- (1) a declare in capital letters
- (2) Correct function is printf ( )

86. Predict the output of the following code.

(AJK/G1-16)

```
void main ()
{
 char ch1 = 'A' , ch2 = 'B' , ch3 = 'C';
 printf("ch2 \n %c \n ch1", ch3);
 getch ();
}
```

**Output**

|                 |
|-----------------|
| Ch <sub>2</sub> |
| C               |
| Ch <sub>1</sub> |

87. Find the output of the following code.

(AJK/G1-16)

```
include <stdio.h>
include <conio.h>
void main ()
{
 int x , y , z;
 printf ("%d %d %d" , &x , &y , &z);
 y = x + z;
 x = z + 1;
 printf ("%d %d %d" , x , y , z);
 getch ();
}
```

**Output:**

-12 - 14 - 16 - 292217555 - 29222

Because not use proper scanf it shows the garbage values

88. **Trace the Error** (SGD-14)  
#include<stdio.h><sup>(1)</sup>  
main ( )  
{  
print<sup>(2)</sup>("Hello world");  
}  
**Correction:**  
(1) In header file <> is always used.  
(2) C have not function print correct like this printf ("Hello world");
89. **Trace the output** (SGD-14)  
float f = 3.14159;  
printf("f=%4.2f",f);  
**Output**  
f = 3.14
90. **Write the output of the following code:** (LHR-14)  
Int x = 9;  
x = x%4;  
printf ("n%dis result", x); (LHR-14)  
**Output**  
n1is result
91. **Write the output of the following statements.** (GRW-14)  
main ( )  
{  
printf ("555/n")  
printf ("55");  
}  
**Output**  
555 /n 55
92. **Trace the output of the following code:** (LHR-14)  
{ int x, y, z;  
x = 3;  
y = 2;  
z = 4;  
printf ("%d%d%d", x + y, y + 5, x + z);  
getche ( );  
**Output**  
577
93. **Trace Errors in the following code:** (BWP-17)  
#include <stdio.h>  
{  
main ( )  
{  
printf("OK");  
}  
}  
**Correction:**  
(1) After include mention the header file that is include just like # include <stdio.h>  
(2) main is always used after header file mean in next line.
94. **Trace output of the following code:** (BWP-17)  
int i = 786;  
Printf ("%6d", i);  
**Output**  
□ □ □ 786

## 95. Trace errors in the following code.

(BWP-17)

```
#includes <stdio.h>
(1)Void main ()
{
 inti (2);
 scanf ("%d" ,i(3));
 Printf ("%d , i);
}
```

Correction:

(1) space is missing between int and i. Correct is "int i"

(2) V is capital used that is wrong.

(3) &amp; sign is missed with i in scanf function.

## 96. Find out the output of following code.

(BWP-17)

```
int a =10, b= 12;
sum = a+b;
printf ("%d" ,Sum);
Output:
```

In this programme sum is not declare data type  
if we declare "int sum" then output is 22 sum is  
undefined symbol

## 97. Find the error:

(D.G.K-17)

```
(1)Void main ()
{
 int b = 5 (2)
 printf ("%d" , b) (2)
}
```

Correction:

(1) Letters are used in upper cases

(2) Statement terminator; is missing;

## 98. Write the output

(D.G.K-17)

```
float f = 3.14159;
Printf ("%7.3 f" , f);
Output
```

□ □ 3.141

## 99. What will be the output from the following code:

(GUJ-17)

```
void main ();(1)
{
 inta (2) b, temp;
 (3)t =10;
 b =20;
 tem = a;
 a =b;
 b = temp(4);
 printf ("a=%d\tb=%d",a,b);
}
```

Correction:

Here is no any output because so many errors for example:

(1) main ( ) is not terminated with ;, no any symbol used at the end of main() function

(2) Not give space between "inta" like this int a and , is used after a

(3) t is undefined variable.

(4) Statement terminator; is missing;

100. Find our errors from the following code.

```
{
 (1)chr ch, ch2;
 (2)ch1 =2 (3)
 ch2 ='6(4)';
}
```

**Correction:**

- (1) Chr not any data type correct data type is char.
- (2) Ch1 is undefined variable
- (3) (' ') is not used
- (4) Statement terminator; is missing;

101. Trace errors from the following:

```
void main ();(1)
{
 float l(2) = 15.6;
 printf ("%d(3)",b);
}
```

**Correction:**

- (1) statement terminator is not used after main ().
- (2) l is write in upper case.
- (3) Format specifier is wrong used because data type is float.

102. Find any two errors of following code:

```
int number = 6 (1)
number + +;
printf ("% d \ n" ,f (2));
```

**Correction:**

- (1) Statement terminator; is missing;
- (2) here f is used that is undefined. here use number.

103. Write down output of the following:

```
float f = 3.14159;
printf ("f= % 4.2f" ,f);
```

**Output**

|       |   |    |     |
|-------|---|----|-----|
| f ≈ 3 | . | 14 | 159 |
|-------|---|----|-----|

104. Find error

```
{
 (1)Float area, r(2)
 printf("Enter radius");
}
```

**Correction:**

- (1) F used upper case
- (2) Statement terminator; is missing;

105. Trace error in the following code.

```
void main (); (1)
{
 (2)Float x;
 printf (3)(Input value);
 Scanf ("%d(4) ,x);
}
```

**Correction:**

- (1) Not statement terminator ; is used here
- (2) F used upper case
- (3) (" ") is not used in printf ( )
- (4) "&" is not used in scanf ( ) and format specifier %d is used in scanf() that is wrong, correct %f because data type is float

**106. Trace the output of the following:- (MUL-17)**

```
x = 5;
y = x ++;
printf ("%d %d %d", x, y);
```

**Output**

65

**107. Write output of the following code(MUL-17)**

```
int n = - 25.41;
Printf ("%d6.2 f": , n);
```

**Output**

-256.2f

**108. Find the error void main ( ) (D.G.K-17)**

```
{
 (1)int a = 10;
 printf ("%d(2)s", b(3));
}
```

**Correction:**

- (1) i is used in upper case
- (2) % d is used because %s is always used for string
- (3) b is undefined symbol

**109. Find the output of the following code. (RWP-17)**

```
#include <stdio.h>
void main ()
{
 int x = 10, y = 20, z = 30;
 x = x + y;
 y = y + z;
 z = x - y;
 printf ("result = %d %d %d", x, y, z);
}
```

**Output**

Result = 3050 -20

**110. Find the errors. (RWP-17)**

```
#include <stdio.h>
void main ()
{
 int x, y;
 x = 10(1); y = 20;
 printf ("%d %d %d", x, y(2));
}
```

**Correction:**

- (1) Here is always, is used not ;
- (2) ( ) is use here not any symbol is used before closing brackets)

**111. Write output of the following code: (GUJ-18)**

```
int x = 5;
printf ("%d %d, 2 * x, 3 *x);
```

**Output**

1015



**112. Write output of the following code: (GUJ-18)**

```
int a, b, c;
a = 15;
b = 10;
c = a + b;
printf ("c = %d", --c);
```

**Output**

c = 24

**113. Find errors from the following code: (GUJ-18)**

```
#include <stdio.h> (1)
void main ()
{
int x = 10, y = 13;
x = ++ y;
y = x ++;
printf ("%d %d", x, y);
}
```

**Correction:**

- (1) Header file is not correct, correct is stdio.h

**114. Show output:**

(SWL-17)

```
int i = 515 ;
Printf (" i=%5d; ,i)
Printf ("i=%2d" ,i);
```

**Output**

i = 515  
i = 515

**115. Find errors from the following codes: (SWL-17)**

```
include < studio.h> (1)
include < cono.h> (1)
{
void main () ; (2)
printf ("Assemble Language") (3)
}
```

**Correction:**

- (1) Header files are not correct.  
(2) Not ; is used after main ( )  
(3) Statement terminator is missing.

**116. Find errors:**

(SWL-17)

```
main [] ; (1)
{
float n ;
Printf (" %d" ; n);
}
```

**Correction:**

- (1) ; is not used after main ( ) and ( ) are used  
(2) %f format specifier is used for float data type.

**117. Find Errors in the following code: (SGD-17)**

```
void main ()
{
(1) intnum = 10 (2)
Num += 5 (2)
printf ("\\n%/d,(3) num);
}
```

**Correction:**

- (1) space is missed correct s int num;
- (2) Statement terminator; is missing;
- (3) ( " " ) is missed in printf ( )

**118. Find the errors in the following code:(SGD-17)**

```
Void main (); (1)
{
int A (2) = 10 (4)
printf ("%d(3) ,a); (4)
}
```

**Correction:**

- (1) Statement terminator ; not used after main ( )
- (2) Variable is declare in upper case
- (3) ( " " ) is not used in printf ( )
- (4) Statement terminator; is missing;

**119. Trace output of the following code: (LHR-18)**

```
int x = 0, y = 5, z = 4;
x = y + z * 5;
printf ("Result = %d: ,x);
```

**Output**

Result = 25

**120. Find errors in following code: (LHR-18)**

```
void main (); (1)
{
print (2) ("Pakistan");
}
```

**Correction:**

- (1) Statement terminator is not use here
- (2) printf is used in C-lang instead of print

**121. Trace output of following code: (LHR-18)**

```
int a = 5, b = 10;
int c = a + b * 2;
printf ("The output is %5d: ,c);
```

**Output**

The output is 30 25

**122. Find errors of C code: (MUL-18)**

```
int a = 10, b = 40 (1)
print (2) ("sum(3) ", a + b)
```

**Correction:**

- (1) Missing ;statement terminator
- (2) Correct function of output is printf ( )
- (3) Format specifier is not used in printf ( )

123. Write output of the code: (MUL-18)

```
float f = 3.1415 ;
printf ("%7.2f", f)
```

**Output**

□ □ □ 3.14

124. Find the error. (RWP-18)

```
{
 int c = 7(1)
 printf ("%d", C(2))(3)
}
```

**Correction:**

- (1) Statement terminator; is missing;
- (2) C is in upper case
- (3) Statement terminator; is missing;

125. Find the error. (RWP-18)

```
void main ()
{
 int a = 10(1)
 printf ("%d",a);
}
```

**Correction:**

- (1) Statement terminator; is missing;

126. Write the output. (RWP-18)

```
float f = 3.14159;
printf ("%8.4f",f);
```

**Output**

□ □ □ 3.1416

127. Write output of following code. (SWL-18)

```
int x = 4 * 5 \ 2 + 9;
Printf ("% d" , x);
```

**Output**

19

128. Trace errors in following code: (SWL-18)

```
{
 x = 20 ;
 y = 40 ;
 x + y = z(1)
 printf ("% d",z)(2))(3)
```

**Correction:**

- (1) Wrong Expression correct is  $z = x + y$
- (2) ; is not used , is used here
- (3) Statement terminator; is missing;

129. Write the output of following code: (SWL-18)

```
int a, b, c;
a = 5 ;
b = 10 ;
c = a + b
Printf ("The sum of a + b = % d",c) ;
```

**Output**

The sum of a + b = 15

130. What is the output of following code segment?

```
int n = 10;
n % = 2;
n + = 5;
printf ("%d", n);
```

**Output**

5

(SGD-18)

131. What is the output of following code segment.

```
int x, y = 5;
x = y + +;
printf ("%d / n %d", x, y);
```

**Output**

5/n 6

(SGD-18)

132. Find errors in the following code segment.

```
#include <stdio.h>;(1)
(2)VOID main ()
{
 printf ("Hello c");
}
```

**Correction:**

(1) C – lang is case sensitive lang. Void write in small letters.

(2) ; is not use after header file

(SGD-18)

133. Trace the errors.

(RWP-17)

```
#include <stdio.h>
void main ()
{
 (1)Printf ("High level language")(2)
}
```

**Correction:**

(1) C-language is case sensitive we write the source code in small letters, so printf write here print ( )

(2) Statement terminator is missing.

## SECTION III

### LONG QUESTIONS

1. Write the use of printf ( ) function with an example. (RWP-19) (GRW-19)
2. Write a detail note on scanf ( ) function. (AJK-19)

# DECISION CONSTRUCTIONS

## SECTION I

### MULTIPLE CHOICE QUESTIONS

*From PTB Exercise:*

No MCQs in this Chapter by Punjab Text Book Board.

*From Punjab Board:*

1. An if statement inside another if statement is called: (RWP-19)  
(a) if statement (b) if-else statement  
(c) Nested if statement (d) switch statement
2. Conditional operator is an alternative of (AJK-19) (SGD-19) (SGD-14)  
(a) if (b) if-else  
(c) nested-if (d) if-else-if
3. In if statement, false is represented by: (MTN-18)  
(a) 0 (b) 1  
(c) 2 (d) 3
4. Which keyword is not used in switch statement. (SGD-18)  
(a) If (b) Switch  
(c) Case (d) Default
5. How do you find loop statement in following options? (MTN-18) (RWP-18) (RWP-19)  
(a) if else (b) if  
(c) switch (d) for
6. Another term for a conditional operator. (RWP-18) (GRW-19)  
(a) Ternary (b) Binary  
(c) Byte (d) iteration
7. Conditional operator takes: (RWP-18)  
(a) one operand (b) Two operands  
(c) three operands (d) four operands

8. Another term for a computer making a decision is. (SWL/G1-15) (FSD-19) (RWP/G1-15)  
(a) Sequential (b) Selection  
(c) Repetition (d) Iteration
9. Which programming structure makes a comparison? (SWL/G1-15) (BWP-15) (D.G.K. - 19)  
(a) Relation (b) Decision  
(c) Sequence (d) Repetition
10. Another term used for making a decision is: (BWP-15)  
(a) Relation (b) Sequence  
(c) Repetition (d) Selection
11. The programming structure used for a computer making a decision is: (SWL-18)  
(a) sequential (b) selection  
(c) repetition (d) iteration
12. Another term used for decision is: (AJK/G1-14)  
(a) Repetition (b) Iteration  
(c) Selection (d) Sequential
13. Which of the following operators is not used as decision making structure: (AJK/G1-14)  
(a) If (b) Break  
(c) If-else (d) Switch
14. Which programming structure executes the statement in order; (AJK/G1-16) (FSD-19) (SGD-19)  
(a) Nested (b) Repetition  
(c) Selection (d) Sequence
15. Another name for conditional operator is: (AJK/G1-16)  
(a) Binary operator (b) Ternary operator  
(c) Unary operator (d) Quaternary operator
16. In if statement, true is represented by: (FSD-14) (LHR-14)  
(a) 0 (b) 1  
(c) 2 (d) 3
17. In If statement false is represented by: (LHR-14)  
(a) 0 (b) 1  
(c) 2 (d) 3
18. Another term used for making a decision is: (GRW-14)  
(a) selection (b) sequence  
(c) repetition (d) iteration
19. What is the final value of X after executing the following code: (SGD-14)  
For(int x=0; x<10 ; x++)  
(a) 10 (b) 9  
(c) 0 (d) 1

## SECTION II

## SHORT QUESTIONS ANSWERS

*From PTB Exercise:*

- 1. What is control structure? Briefly describe the basic control structures for writing program.**

(BWP-17) (SGD-14) (FSD/G1-15) (SWL-18) (FSD/G1-16) (MTN/G1-16) (SWL-19)

**Ans.** In programming, “control structures” are the structures used to control the flow of a program. The two main classes are conditionals and loops. In C, blocks (also called compounds statements) group multiple statements together so they can be treated as one. They also coping mechanism.

**Flow of control through any given function is implemented with three basic types of control structures:**

**Sequential:** default mode. ...

- **Selection:** used for decisions, branching -- choosing between 2 or more alternative paths.

**Repetition:** used for looping, i.e. repeating a piece of code multiple times in a row.

2. How many selection statements are available in C? Discuss the difference between them.

**Ans.** The selection structure is used for selecting a statement or group of statement for execution on the basis of given a condition. The selection structure is also known as decision structure. In C the most important selection structure: if statement, if else statement, and switch statement.

**Switch** is generally faster than a long list of ifs because the compiler can generate a jump table. The longer the list, the **better** a **switch statement** is over a series of **if** statements. Far more important than the performance benefits of **switch** (which are relatively slight, but worth noting) are the readability issues.

- 3. Write the general form of statements:**

- (i) if statement with one alternative

**Ans. Conditional operator**

- (ii) if statement with two alternatives

**Ans. if-else-if**

- (iii) if statements with multiple alternatives

**Ans. Switch**

- (iv) switch statements

4. Rewrite the program given in example 4 using if statement. attempt the following parts:

- (v) Assuming  $x$  is 10.0 and  $y$  is 15.0, what are the values of the following conditions:

- |                     |      |   |
|---------------------|------|---|
| a) $x != y$         |      |   |
| ture 1              |      |   |
| b) $x < x$          | true | 1 |
| c) $x > = y - x$    | True | 1 |
| d) $x == y + x - y$ | True | 1 |

5. Write an expression to test each of the following relationship.

**Ans. (a)** age from 18 to 25.

```
If(age>=18 && age<=25)
```

Temperature is less than  $4^{\circ}\text{C}$

```
if(temp>25 && temp<40)
```

year is divide by 4

```
year%4==0
```

speed is not

Speed  $\leq 80$

$v$  is greater

(c)  $y$  is greater than  $x$  and less than  $z$ .  
 $\forall x \& \exists z: 1 < z$

it is either

(1)  $w$  is either equal to 0 or not greater than 3.  
 $w = 0 \vee w \leq 3$

$$W=0 \parallel W>0$$



6. Assigns the value of 1 to 10 variable test if k is in the range -m through +m inclusive. Otherwise, assigns a value of zero.

Ans. If  $k > -m$  &  $k <= +m$

V=1

Else

V=0

7. A year is a leap year if it is divisible four, except that any year divisible by 100 is a leap year only if it is divisible by 400. Write a program that inputs a year such as 1996, 1800 and 2010, and displays "Leap year" if it is leap year, otherwise display : not a leap year".

Ans. #include <stdio.h>

int main()

{

int year;

printf("Enter a year: ");

scanf("%d", &year);

if(year%4 == 0)

{

if( year%100 == 0)

{

// year is divisible by 400, hence the year is a leap year

if ( year%400 == 0)

printf("%d is a leap year.", year);

else

printf("%d is not a leap year.", year);

}

else

printf("%d is a leap year.", year);

}

else

printf("%d is not a leap year.", year);

return 0;

}

8. Write a program that inputs two numbers and ask for the choice of the user, if users enters 1 then displays the sum of numbers if users enters 2 then displays result of subtraction of the numbers, if user enters 3 then display the result of the multiplication of the numbers and if the users enters 4 then displays results of the division of the numbers.

Ans. #include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int input, a, b, result;

char option;

do{

printf("calculator:\n");

**Calculator:**

Enter 1 for addition;  
Enter 2 for subtraction;  
Enter 3 for multiplication;  
Enter 4 for divisions;

1

Enter a number;

5

Enter second number;

95

The addition is : 100

Do you want to continue ? (y/n)

```
printf("\nEnter 1 for addition:\n ");
printf("Enter 2 for subtraction:\n ");
printf("Enter 3 for multiplication:\n ");
printf("Enter 4 for division:\n ");
scanf("%d",&input);
printf("Enter a number:\n");
scanf("%d",&a);
printf("Enter second number:\n");
scanf("%d",&b);
switch(input){
case 1 : result=a+b;
 printf("The addition is : %d\n",result);
 break;
case 2 : result=a-b;
 printf("The Subtraction is : %d\n",result);
 break;
case 3 : result=a*b;
 printf("The Multiplication is : %d\n",result);
 break;
case 4 : result=a/b;
 printf("The division is : %d\n",result);
 break;
default: printf("wrong input\n");
}
printf("Do you want to continue ? (y/n)\n");
option=getche();
}while(option=='y');
return 0;
}
```

**From Punjab Board:**

1. How instructions are executed in sequence structure. (FSD-19)

Ans. Usually, **C programs** are **sequential** in nature. Generally, **sequence** of statements are written in order to accomplish a specific activity. So statements are **executed** in the order they are specified in the **program**. This way of **executing** statements **sequentially** is known as **Sequential** control statements.

2. What is condition? (AJK-19) (SGD-19)

Ans. The definition of **condition** is the state something or someone is in or can also refer to a specific illness. An **example of condition** is a brand new sofa with no defects. An **example of a condition** is a harsh work environment. An **example of a condition** is a cold or the flu.

3. How does a selection structure select a statement to execute? (D.G.K - 19)

Ans. The **selection structure** tests a condition, then executes one sequence of statements instead of another, depending on whether the condition is true or false. A condition is **any variable or expression** that returns a Boolean value ( TRUE or FALSE )

4. Define sequential structure. (MTN/G1-15)

Ans. In a **C program**, the flow of execution is **sequential** by default, that is, one statement after another in their order as in the source code of a function. With various control structures, the order of execution can be different from the **sequential** order. ... Control structures are usually used with compound statements

5. What is control structure? (BWP-17) (SGD-14) (FSD/G1-15) (SWL-18) (FSD/G1-16) (MTN/G1-16) (SWL-19)

Ans. In programming, "control structures" are the structures used to control the flow of a program. The two main classes are conditionals and loops. In C, blocks (also called compounds statements) group **multiple** statements together so they can be treated as one. They also coping mechanism.

**OR** A control structure is a block of programming that analyzes variables and chooses a direction in which to go based on given parameters. ... Hence it is the basic decision-making process in computing; flow control determines how a computer will respond when given certain conditions and parameters.

6. List any four types of control structure. (GUJ-17)(FSD-18)

Ans. Flow of control through any given function is implemented with three basic types of control structures:

- Sequential: default mode.
- Selection: used for decisions, branching choosing between 2 or more alternative paths.
- Repetition: used for looping, i.e. repeating a piece of code multiple times in a row.

7. Distinguish between if else and switch statements? (SWL/G1-15) (BWP-15) (RWP/G1-15) (KSMR/G1-16) (LHR-18) (MTN/G1-16) (BWP-14) (RWP/G1-16) (AJK-17) (FSD/G1-16) (MUL-18)

Ans. A **switch statement** works much faster than equivalent **if-else** ladder. It is because compiler generates a jump table for a **switch** during compilation. Consequently, during execution, instead of checking which case is satisfied, it only decides which case has to be executed.

**Switch** is generally faster than a long list of ifs because the compiler can generate a jump table. The longer the list, the **better** a **switch statement** is over a series of **if** statements. Far more important than the performance benefits of **switch** (which are relatively slight, but worth noting) are the readability issues.

8. Write the syntax of switch-case statement. (D.G.K/G1-15) (MTN/G1-15)

Ans. Switch statement. In computer programming languages, a switch statement is a type of selection control mechanism used to allow the **value** of a variable or expression to change the **control** flow of program execution via search and map.

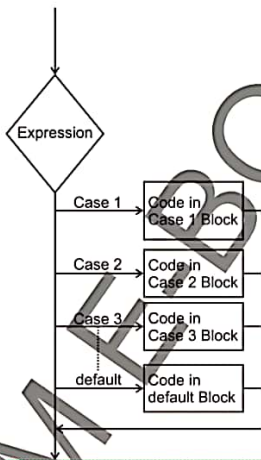
```
switch(expression)
{
 case val - 1:
 statements - 1;
 break;
 case val - 2:
 statements - 2;
```

```

break;

case val – n:
statements – n;
break;
default:
statements;

```



9. What is compound statement?

(FSD/G1-16)(MTN-19)(RWP-19)(LHR-19)

**Ans.** A **compound statement** (also called a "block") typically appears as the body of another **statement**, such as the **if statement**. Declarations and Types describes the form and meaning of the declarations that can appear at the head of a **compound statement**.

Simply we can say:- A **compound statement** is a **statement** which results from the application of one or more logical connectives to a collection of simple **statements**. (or) The set of statement surrounded by curly braces { } is called compound statement.

10. Define selection statement. List two basic selection statements in c. (SGD/G1-16)

**Ans.** A selection statement causes the program control to be transferred to a specific flow based upon whether a certain condition is true or not. Three types of selection statements.

**if statement:**

Performs an action, if a condition is true; skips it, if false.

Single-selection statement—selects or ignores a single action (or group of actions).

**if...else statement:**

Performs an action if a condition is true and performs a different action if the condition is false.

Double-selection statement—selects between two different actions (or groups of actions).

**switch statement:**

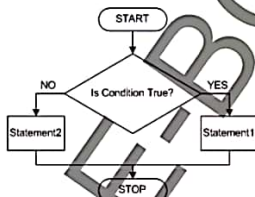
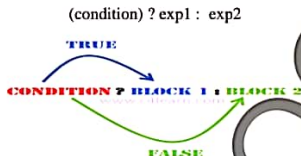
Performs one of several actions, based on the value of an expression.

Multiple-selection statement—selects among many different actions (or groups of actions).

11. Define conditional operator? Write its syntax and example. (GRW/G1-16) (LHR/G1-16)  
(D.G.K/G1-16) (LHR/G1-15) (RWP-14) (D.G.K-14) (LHR-17) (SWL/G1-15)

Ans. The conditional operator (`? :`) is a ternary operator (it takes three operands). The conditional operator works as follows: The first operand is implicitly converted to bool . ... If the first operand evaluates to true (1), the second operand is evaluated.

General syntax of conditional operator is:



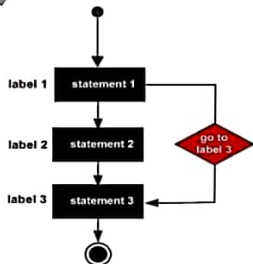
12. What do you mean by condition? (D.G.K/G1-15)

Ans. The definition of **condition** is the state something or someone is in or can also refer to a specific illness. An **example of condition** is a brand new sofa with no defects. An **example of a condition** is a harsh work environment. An **example of a condition** is a cold or the flu.

13. Define goto statement. (SWL/G1-15)

Ans. A goto statement provides an unconditional jump from the **goto** to a labeled **statement** in the same function.

NOTE – Use of **goto** statement is highly discouraged because it makes difficult to trace the control flow of a program, making the program hard to understand and hard to modify.



## 14. Explain if statement.

(RWP-14) (GRW-14)

Ans.

```
if(boolean_expression) {
 /* statement(s) will execute if the
 boolean expression is true */
} If the Boolean expression
evaluates to true, then the block of
code inside the 'if' statement will
be executed.
```

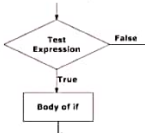


Fig: Operation of if statement

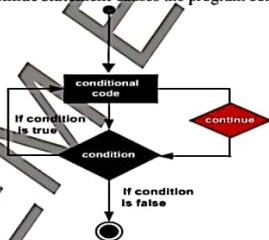
## 15. Write an expression in C Language for "Number is divisible by 5" (LHR-14)

Ans. C program to find printf("%d is Number is divisible by 5

```
{
int no;
printf("Enter a No");
scanf("%d", &no);
if ((no % 5) == 0)
printf("%d is Number is divisible by 5 \n", no);
else
printf("%d is Number is not divisible by 5 \n", no);
}
```

## 16. What is Continue statement? (MUL-18)

Ans. The **continue statement** in C programming works somewhat like the **break statement**. Instead of forcing termination, it forces the next iteration of the loop to take place, skipping any code in between. ... For the while and do...while loops, **continue statement** causes the program control to pass to the conditional tests.

Programming

(D.G.K/G1-16)

## 1. Find out errors:

```
include <stdio.h>
void main ()
{
 if (50 > 20) then
 printf (" Islamic country ");
 getch ();
}
```

Correction:

- (1) .h extension is not mention.
- (2) Delimiter { } are used not this sign is used [ ]
- (3) Then keyword is not use in C-Lang
- (4) Statement terminator; is missing;
- (5) Delimiter { } are always used

**2. Write output. (D.G.K/G1-16)**

```
int p,q,r;
p = 10;
q = 3;
if (p % q == 3)
r = 0;
else
r = 1;
printf (" %d ", r);
```

**Output**

1

**3. Trace the errors in the following codes:****(BWP/G1-16)**

```
(1)Void main ()
int x, y = 5;
If (x > y);(2)
printf (" x is largest ");
(3)Else
printf (" y is largest ");
getch ();
}
```

**Correction:**

- (1) V & E are capital letters that is wrong.  
(2) ; is not used after if ( )

**4. Trace the output of the following codes:****(BWP/G1-16)**

```
int a = 4, b = 2, c = 5;
if (a > b)
a = 5;
if (c == a)
a = 6;
a = 7;
printf (" %d ", a);
```

**Output**

7

**5. Trace the output:****(FSD/G1-16)**

```
if (1==2)
printf("Hello");
else
printf("Equality");
```

**Output**

Equality

**6. Trace the error:****(FSD/G1-16)**

```
void main ()
{
if (16 > 10) then(1)
printf ("%c",(3) "Pakistan");
getch ();
}
```

**Correction:**

- (1) Then keyword is not used in C-Lang  
(2) (" is missing in printf ( ) before Pakistan  
(3) if we want print Pakistan then we use %s format specifier for string



7. Trace the error in the following code:

(FSD/G1-16)

```
void main ()
{
 int x = 0 (1)
 if (x = (2) 1)
 printf (" Hello ");
 else
 printf (" Bye ");
}
```

**Correction:**

- (1) Statement terminator; is missing; .  
(2) Expression is wrong correct is if (x == 1)

8. Predict the output for the following code:

(FSD/G1-16)

```
int a, b, c ;
a = 10;
b = 3;
if (a % b == 1)
 c = 0 ;
else
 c = 1 ;
printf (" % d " ,c) ;
```

**Output**

0

9. Trace errors in the following code:

(GRW/G1-16)

```
void main (void);(1)
{
 int x = 10;
 if (x > 5 && (2) < 15)
 printf ((3) "x is valid");
 else
 printf ("invalid");
}
```

**Correction:**

- (1) ; is not used after main ( ) this is not syntax  
(2) Condition is wrong  
(3) printf statement is not proper ( " is not used in printf )

10. Trace error in the following code:

(KSMR/G1-16)

```
void main (void)
{
 int x = 10 ;
 if (x != 10) (1)
 x ++ ;
 else (2) (x == 10)
 x -- ;
}
```

**Correction:**

- (1) ; is not used after if ( ).  
(2) If keyword is missing after else.

11. Write down output of the following code;

```
void main (void)
{
 int x = 10;
 (x % 2 == 0) printf("Even"); printf(" odd ");
}
```

**Output**

Even

12. Predict the output of the following code:

(LHR/G1-16)

```
if (4%2 == 0)
printf (" programming makes the life interesting \n");
else
printf (" Programming is difficult to learn");
```

**Output**

programming makes the life interesting

13. Trace the errors:

(LHR/G1-16)

```
int p = 20, (1)
if (price(1) == 20)
price = 0, (2)
else
price = 2, (3)
```

**Correction:**

- (1) Price is undefined symbol
- (2) Statement terminator; is missing and comma is wrong used here
- (3) Statement terminator; is missing and comma is wrong used here

14. Trace error:

(MTN/G1-16)

```
void main ()
{
 int a = 2;
 if (a == 1)
 printf ("OK");
 else
 printf ("Cancel") (1)
 getch () (2)
}
```

**Correction:**

- (1) & (2) Statement terminator; is missing;

15. Trace output:

(MTN/G1-16)

```
int a = 5, b = 10;
if (a > b)
printf ("Low Triangle");
else
printf ("Huge Triangle");
```

**Output**

Huge triangle

16. What is the error in the following code?

(RWP/G1-16)

```
int x=10, y = 20;
if(x>10 &&(1) y<30)
printf ("%d", x + y);
```

**Correction:**

- (1) Condition is wrong AND operates is denote with && correct condition is if (x > 10 && y < 30)

17. Trace the error from the following code:

(SGD/G1-16)

```
(1)Void main ()
{
 int a = 2 ;
 if (a = 1);(2)
 printf ("ok");
 (3)Else
 printf ("No");
}
```

**Correction**

- (1) Here V is used in capital letter C is a case sensitive language, so v is used in small letter  
(2) ; is not used here  
(3) Wrong keyword correct is else mean use in small letters

18. Predict the output for the following code:

(SGD/G1-16)

```
int a = 1;
int b = 6;
if (a + b < 7)
 printf (" % d", a);
else
 printf (" % d", b);
```

**Output**

6

19. Trace the errors in the following code.

(SWL/G1-15)

```
include < stdio.h >
void main ();(1)
{
 If (16 < 18)(2) then
 printf (" % c(4) ", OK(3));
 getch ();
}
```

**Correction:**

- (1) ; is not used after if ( )  
(2) Then keyword is not used in C-Language  
(3) ( " ") is missing correct is ("OK")  
(4) Format specifier In this program no any character data type is declared, if we want to print OK then take %s format specifier for string.

20. What is the output of the following code?

(SWL/G1-15)

```
int m , n ;
m = 0 ;
n = m ;
if (m == n)
 printf (" BWP ");
else
 printf (" LHR ");
```

**Output**

BWP

## 21. Trace the output.

(RWP/G1-15)

```
void main ()
{
 int p, q, r;
 p=10;
 q=3;
 if (p % q == 3)
 r=0;
 else
 r=1;
 printf ("%d ", r);
 getch ();
}
```

**Output**

1

## 22. Trace the error.

(RWP/G1-15)

```
void main ()
{
 int R(2);
 r=25(3)
 if (r>0)
 printf("Good")(3)
 getch ();
}
```

**Correction:**

- (1) ; is not use after main ( )
- (2) R is declare in capital letters.
- (3) ; is missing after printf()

## 23. Trace the output.

(RWP/G1-15)

```
float f=3.14159;
printf ("f=%5.1f",f);
```

**Output:**

f=3.1

## 24. Determine the output of the following.

(AJK/G1-14)

```
if (1 == 2)
 printf ("Hello");
else if (2 == 1)
 printf ("world");
```

**Output**

Nothing any output because both if conditions are false.

## 25. What is the output of given code? (MTN/G1-15)

```
int b = 6, c = 5;
if (b++ == 7 & & ++c == 5)
{
 b = c;
 printf ("%d \n", b++);
}
```

```
else
 printf ("%d \n", b--);
```

**Output**

7

26. Trace the error of the following code.

```
#include < CONIO-H(1) >
void main () ;(2) {
 int r = 25;
 If (R(4) > 8)
 R = R * 3.141593(3)
 getch () ; }
```

**Correction:**

- (1) Wrong header file is written correct is <conio.h> use in small letters.
- (2) ; is not use after main ( ) this is against the structure of C-Program
- (3) ; is missing here and R is used in capital letter.
- (4) int r Variable declare in small letter , but use R.

27. Find the output of following code.

```
int marks = 50 ;
if (marks >= 33)
 printf ("Pass") ;
else
 printf (" Fail") ;
```

**Output**

Pass

28. Trace the errors.

```
#include(1) <stdio.h>
void main (void)
{
 int x = 4 ;
 int y = 3 ;
 if(x><(2) y)
 printf (" x is greater ");
}
```

**Correction:**

- (1) Less than "<" is missing
- (2) Run time error or condition is not true

29. Trace the errors in the following code.

```
int word = 10;
if (word != 10)
 word = 0(1)
else
 word += 2(2)
```

**Correction:**

- (3) And (2) ; missing + logical error exist in condition

30. What is the output of the following code:

```
int x, y;
x = 5;
y = 7;
if (x < y)
 printf (" Hello");
else
 printf ("Pakistan");
```

**Output**

Pakistan

(D.G.K/G1-15)

(D.G.K/G1-15)

(D.G.K/G1-15)

(LHR/G1-15)

(LHR/G1-15)

**31. Trace output of the following program.****(BWP-15)**

```
int a = 4, b = 2, c = 5;
if (a > b)
a = 5;
if (c == a)
a = 6 ;
else a = 7 ;
printf ("%d ", a);
```

**Output**

6

**32. Find the errors in the given code: (BWP-15)**

```
void main ()
{ int a = 2 (1)
(3)if (a == 1)
If printf (" OK ");
(2)Else
(2)Printf(" Cancel ");
getch () ;}
```

**Correction:**

- (1) ; is missing after the Statement
- (2) Use upper letter correct is else and printf
- (3) Condition is wrong, correct is if(a == 1)

**33. Trace the errors of following code.(RWP-14)**

```
include (stdio.h)(1)
{
main (2)() ;(3)
```

**Correction:**

- (1) <> sign is not used in header files
- (2) \* is not used after main ( )
- (3) ; is not used after main function

**34. Trace the output:****(FSD/G1-15)**

```
if (1 == 2)
printf("Hello");
else
printf("Equality");
```

**Output**

Equality

**35. Trace the error:**

```
void main ()
{
if (16 > 10)(1) then
printf("%c(3)", (2)Pakistan");
getch ();
}
```

**Correction:**

- (1) then keyword is not used in C-Language
- (2) ( " ) is missing
- (3) if we print Pakistan, we use %s for string

## 36. Find output of following code

(RWP-14)

```
int price = 10;
if (price != 10)
price = 0;
else
price += 2;
printf ("%d", price);
```

**Output:**

## 37. Write the purpose of "%c" format specifier.

(SWL-19)

The **format specifier** is used during input and output. It is a way to tell the compiler what type of data is in a variable during taking input using `scanf()` or printing using `printf()`. %c format specifier is **used** to represent characters. It is **used** in `printf` function to print the character stored in variable. The value stored in char variable can be retrieved and printed using %c format specifier. %s format specifier is **used** to represent strings

## 38. Trace the Error.

(D.G.K-14)

```
int p, q, r;
(1)p = 50000, 10(2)
printf("%d ", p);
```

**Correction:**

- (1) P declare is in Capital letter C is a case sensitive language
- (2) ; is missing at the end of statement
- (3) if we want to print Pakistan, we use %s for string

## 39. Trace the out put.

(D.G.K-14)

```
int x = 1;
int y = 2 ;
int z = 3;
if ((x == y) || (y == z) || (z == 2))
printf("yes");
else
printf("no");
```

**Output**

## 40. What is the output of the following code?

(FSD-14)

```
int x = 1;
int y = z = 3;
if(x == y) || (y == z) || (z == 2))
printf("Yes");
else
printf("No");
```

**Output**

## 41. Trace the error in the following code:

```
void main (void);(1)
{ int a, b;
a = -10;(2)
b = 40;(2)
if(a<0)
b = SQRT (a);
printf("Result = %f",b);
getch (); }
```



**Correction:**

- (1) ; is not using after main ( )
- (2) ; is not using after end of the statement
- (3) Wrong format specifier %f is used because data type is int here correct is %d.

**42. What is the output of the following code?**

(SWL-14)

```
Int x = 1;
Int y = z = 3;
If((x == y) || (y == z) || (z == 2))
printf("yes");
else
printf("No");
```

**Output**

yes

**43. Trace the output.**

(SGD-14)

```
void main () {
 int p, q, r ;
 p = 10 ; q = 3 ; r = -2;
 if ((p+q) < 14 && (r < q -3))
 printf ("%d" , q) ;

else
 printf ("%d" , p) ;
 getch () ; }
```

**Output**

3

**44. Trace the error**

(SGD-14)

```
void main ()
{
 int a, b(1)
 a = -10(2)
 b = 40(2)
 if (a < 0) ;(3)
 b = sqrt (a) ;
 printf ("Result = %f" , b) ;
 getch () ; }
```

**Correction:**

- (1) ; is missing
- (2) ; is missing
- (3) ; is not use after if ( )

**45. Trace the output.**

(SGD-14)

```
void main () {
 int a, b ;
 for (a = 10; a <= 30; a++)
 {
 b = ++a;
 printf ("%d\n", b);
 getch () ; }
```

**Output**

31

## 46. Trace the output.

(BWP-14)

```
void main ()
{
 int marks ;
 printf ("\n enter your marks) ;
 scanf(" % d", & marks);
 if (marks >= 40)
 PRINTF (" \n Congratulation");
}
```

## Output

```
enter your marks 40
Congratulations
```

## 47. Trace the error in the following code.

(BWP-14)

```
int a ; (1) b ;
a = 10, b = 5 (2)
if (a < b)
 printf (" \n A is less than b") ;
```

## Correction:

- (1) ; is not used here , is used here  
(2) ; is missing

## 48. Find errors in the following code;

(AJK-17)

```
void main ()
{
 int x = 10;
 if (x = 10);(1)
 printf (" int x is valid");
 else printf ("invalid");
```

## Correction:

- (1) ; is not use after if ( )

## 49. Predict the output of the following piece of code;

(AJK-17)

```
if (7 / 2 == 1)
 printf (" programming makes the life boring");
```

## Output

```
Nothing any Output become output because condition is false
```

## 50. What is the output of the following code?

(AJK/G1-15)

```
Int p , q , r ;
p = 10 ;
q = 3 ;
if (p % q == 3)
 r = 0 ;
else
 r = 1 ;
printf (" % d " , r) ;
```

## Output

```
1
```

51. Trace out the errors from the following.

(AJK/G1-15)

```
(1)Void main ()
{
int a = 2 (2)
if (a = 1)
printf (" cancel ");
(1)Else
printf (" ok ");
getch ();
}
```

**Correction:**

- (1) V and E use small letter because C is a case sensitive.  
(2) ; is missing

52. Trace error in the following code; (AJK/G1-16)

```
void main (void)
{
int x = 10 (1)
if (x != 10); (3)
x ++ ;
else (2)(x == 10)
x -- ;
}
```

**Correction:**

- (1) ; is not use here  
(2) If condition or keyword of if is missing  
(3) ; is not used after if condition

53. Write down output of the following code;

(AJK/G1-16)

```
void main (void)
{
int x = 10 ;
(x % 2 == 0) ? printf ("Even") : printf ("odd");
}
```

**Output**

Even

54. What will be output of the following: (LHR-14)

```
int x = 5, y = 10;
if (x > y)
y = 2;
y = y + 1;
printf ("Value of y = %d" , y);
```

**Output:**

Value of y = 11

55. Find error.

(GRW-14)

```
int price = 10 (1)
if (price != 10)
price = 0 (2)
```

**Correction:**

- (1) & (2) statement ; is missing

## 56. Trace the output.

(GRW-14)

```
if (1 == 2)
printf ("hellow");
else
printf ("hello");
else
printf ("correct it");
Output
```

illegal else without if

## 57. Find errors:

(D.G.K-17)

```
include (stdio. N (2))(1)
include <conio.c(2)>
void main ()
{
if (16<10) then (3)
Printf ("%C", "Pakistan");
getch ();
}
```

**Correction:**

- (1) < > sign is used here
- (2) The extension of header file is .h, here use (.N & C)
- (3) Then keyword is not use in C-Language

## 58. Find output of the following code: (GUJ-17)

```
int a = 1, b =6;
if (a + b < 7);
printf ("%d" ,b);
Output
```

.No output because condition is false and ; is used after if()

## 59. Trace errors from the following code: (GUJ-17)

```
if (7 = ! 10 (1))
printf ("hello")(2)
```

**Correction:**

- (1) '=' closing bracket is missing
- (2) Statement terminator ; is missing

## 60. Find the error in the following code.(LHR-17)

```
include <stdio.h>
void main ()
{
scanf ("%d", &r)(3);
area = 3.14 * r * r;
printf ("area = %f", area);
}
```

**Correction:**

- (1) Format specifier is missing
- (2) " is missing in printf ( )
- (3) Statement terminator ; missing

**61. Find the output of the following code.(LHR-17)**

```
#include <stdio.h>
void main ()
{
 char grade = 'c';
 if (grade == 'a' || grade == 'b' && grade == 'c')
 printf ("Fail");
 else
 printf ("Pass")
}
```

**Output**

Pass

**62. Show Errors;**

(MTN-17)

```
in p = 10;
if (p! = 10)
 p = 0(1)
else
```

<sup>(2)</sup>pt = 5;

**Output**

- (1) Statement terminator; missing
- (2) pt undefined symbol

**63. Find errors;**

(MTN-17)

```
(1)Void main ()
{
 int a;(2) b;
 a = 10, b = 5 (3)
 if (a < b)
(1)Printf ('A is less then B');
 getch ();
}
```

**Correction:**

- (1) V and P is upper case , because C is a case sensitive language
- (2) ; is not used (here , is used
- (3) ; is missing

**64. Find errors from following code:**

(RWP-17)

```
int y;(1)z;
if (y == z)
 printf ("Yes")
```

**Correction:**

- (1) , is used like int y , z ; (not ; is used between y and z ) if we use ; then we declare (int y ; int z) like this.

**65. Find output from the following.**

(RWP-17)

```
int price = 19;
if (price! = 10)
 printf ("price");
```

**Output**

price

66. Write down output of the following code.

(SWL-17)

```
void main ()
{int x =10;
if (x != 10)
printf ("Hello");
}
```

**Output**

no output

67. Trace the errors from following code segments.

(SWL-17)

```
void main ()
{
int x = 10 (1)
int y = 15(2)
if (x = y)
printf ("x is equal")(3)
}
```

**Correction:**

(1) ; is missing

(2) ; is missing

(3) ; is missing

68. Determine the output of the following code;

(SGD-17)

```
int a, b;
b = 20; a =10;
if (b<=a)
printf ("No");
else
printf ("Yes");
```

**Output**

Yes

69. Find errors in the following code segment:

(SGD-17)

```
float y = 3.14 (1)
if (x(2) == 3)
printf ("Errors");
```

**Correction:**

(1) ; is missing

(2) x is undefined symbol

70. Determine the output: (FSD-18)

```
int x = 50;
int y = 25;
if (x % y == 0)
 printf ("Result = %d", x % y);
else
 printf ("No result");
```

**Output**

Result = 0

71. Find errors; (FSD-18)

```
void main (void)
{
 int x = 10;
 if (x = 10)
 printf ("True");
 end if(1)
}
```

**Correction:**

(1) This is no syntax of C-Language

72. Show output (GUJ-18)

```
int n = 1, m = 2;
if (n == m)
 printf ("%d", n);
else
{
 if (n == m - 1)
 printf ("%d %d", n, m);
}
```

**Output**

12

73. Trace errors in the following code: (GUJ-18)

```
int n = 7;
if (n > 4 && (1) < 10)
 printf ("n is valid"(2))(4)
(3)Else
 printf ("n is invalid");
```

**Correction:**

- (1) Condition is wrong
- (2) (") is missing in printf ( )
- (3) small 'E' is used in else keyword
- (4) Statement terminates ; is used not (:) is used



**74. Trace the errors in the following code: (LHR-18)**

```
if (7! = 10(4))
 printf ("Hello") (1)
else
 print (2)((3)Welcome);
```

**Correction:**

- (1) ; is missing
- (2) print function is not use in C-Language. Correction function name is printf ( )
- (3) (" ") is missing in printf ( )
- (4) Closing ) is not used after if.

**75. Find out two errors from following code:**

(RWP-18)

```
if (x(1) == y(1))
 printf ("(2)hello") (3)
```

**Correction:**

- (1) x and y are not declare
- (2) (" ") is missing in printf()
- (3) ; is missing

**76. Find output from following code: (RWP-18)**

```
int price = 10;
if (price == 10)
{ printf ("%d", price);
}
```

**Output**

10

**77. Determine the output.**

(SWL-18)

```
int x = 1, y = 2, z = 3;
If ((x == y) (y == z) (z == 2))
 printf ("Yes")
else
 printf ("No")
```

**Output**

Call of non function error is occurred

**78. Find errors**

(SWL-18)

```
int x ;
If (x = 1 or(1) 2)
 printf ("% d" ,x)(2)
```

**Correction:**

- (1) The or is " || " Operator is always used in C.
- (2) ; is missing

**79. Write the error from the following code(SGD-18)**

```
#include< > (1)
void main ()
{ float Y = 3.14 (2)
 if (Y = 3.14)(3)
 printf ("%d", Y) (2)
}
```

**Correction:**

- (1) header file is not included / mentioned
- (2) ; is missing
- (3) Closing ) is missing

**80. Write the output of the following code(SGD-18)**

```
int p = 3, q = 5;
if (p > q)
 printf ("%d", p);
else
 printf ("%d", q);
```

**Output**

5

**SECTION III****LONG QUESTIONS**

1. Define nested if statement. Explain with the help of diagram and example. (SWL-19)
2. Briefly explain the basic control structure for writing programs. (FSD-19)
3. Write down the syntax of "if ( ) - else / switch" statement, diagram and example. (SWL/G1-15)  
(BWP-15) (KSMR/G1-16) (LHR-18) (MTN/G1-16) (BWP-14) (RWP/G1-16) (AJK-17)  
(FSD/G1-16) (MUL-18) (RWP-17) (LHR-19) (GRW-19) (AJK-19) (MTN-19) (D.G.K - 19)
4. Write a program which inputs two numbers and tells whether these numbers are equal or not equal. (SGD-14)
5. Write down a program in which user enters a number and program tells that number is either positive, negative or zero. (RWP-14) (SGD-17) (RWP-19)
6. Write a program that inputs a number of week days and displays the name of the day using Switch Statement. For example if the user enters 1, it displays "Friday" and so on. (GRW-14)
7. Write a program in C Programming Language that accept three numbers from the Keyboard and display the largest number. (BWP-15)
8. Write a program that inputs a year and finds whether it is a leap year or no using if-else structure (Hint : Leap years divisible by 4). (FSD/G1-15) (MTN/G1-16) (LHR/G1-18) (FSD/G1-16) (BWP/G1-16)
9. Write a program that input marks from user and display 'pass; If marks are greater or equal to 40 otherwise 'fail'? (LHR/G1-15) (BWP-14)
10. Write a program that inputs a number and find whether it is even or odd, using if-else structure. (LHR-14) (D.G.K-14) (AJK/G1-14) (D.G.K/G1-15) (AJK/G1-16) (RWP/G1-15)  
(SWL/G1-15) (KSMR/G1-16) (SGD-17)
11. Write a program that inputs two numbers and finds whether second number is square of first number or not. (GRW/G1-16)
12. Write a program in C that inputs the number of the month of the year and display the number of days of the corresponding month using if - else - statement. (e.g. if user enters 2, it will display 28 or 29). (FSD/G1-16)
13. Define Selection Structure. Describe selection structure. How does a selection structure select a statement to execute? (RWP/G1-15) (D.G.K-17) (BWP-15) (AJK/G1-15) (RWP-18) . (BWP-14)



# LOOP CONSTRUCTS

## SECTION I

### MULTIPLE CHOICE QUESTIONS

**From PTB Exercise:**

No MCQs in this Chapter by Punjab Text Book Board.

**From Punjab Board:**

- In a 'for' statement, this expression is executed only once. (SGD-14)(SWL-19)  
(a) test  
(b) initialization  
(c) validation  
(d) increment / decrement
- Which is post test loop? (GRW-19)  
(a) do-while  
(b) while  
(c) for-next  
(d) for
- What is the value of x after executing the code: for (x = 1; x < 10; x++)  
(a) 9  
(b) 10  
(c) both a & b  
(d) 12
- Semicolon is placed at the end of condition in \_\_\_\_\_. (D.G.K - 19) (LHR-19)  
(a) do-while loop  
(b) for loop  
(c) Hybrid loop  
(d) while loop
- This is a control structure that causes a statement or group of statements to repeat (D.G.K - 19)  
(a) Decision statement  
(b) Sequential  
(c) Loop  
(d) Logical
- A loop within a loop is called: (MTN-18)(MTN-18)(BWP-15)(LHR-14)  
(a) Nested loop  
(b) Complex loop  
(c) Infinite loop  
(d) Dual loop
- In counter controlled loop it is important to declare \_\_\_\_\_ and counter variable. (RWP-18)  
(a) Identify  
(b) Initialize  
(c) Size  
(d) Specify
- Functions that are the part of language are called: (MTN/G1-15)  
(a) Intrinsic  
(b) Built in function  
(c) Language defined  
(d) All these
- Which is unconditional control transfer statement? (GRW/G1-15)  
(a) FOR  
(b) GOTO  
(c) IF  
(d) WHILE

10. How many types of loop structure are available in C language? (RWP/G1-15)  
(a) 4 (b) 3  
(c) 2 (d) 6
11. This means to increase a value by one: (RWP/G1-15)(RWP-15)  
(a) Modulus (b) Increment  
(c) Decrement (d) None
12. How many types of loop structure are available in C. (D.G.K/G1-15)  
(a) 4 (b) 2  
(c) 3 (d) 6
13. Which loop structure always executes at least once. (D.G.K/G1-15) (SGD-18) (LHR-19) (GRW-19) (FSD-19)  
(a) While (b) do-while  
(c) For (d) None
14. One execution of a loop is known as a (n). (SWL/G1-15) (BWP-15) (AJK-19)  
(a) Cycle (b) Duration  
(c) Iteration (d) Test
15. Which of the following Loop is called Counter Loop: (BWP-15)(SWL-14)  
(a) While (b) do While  
(c) for ( ) (d) While ( )
16. What is the final value of X after executing the following code? (MTN/G1-15)  
For (int x = 0; x < 5 ; x ++)  
(a) 0 (b) 4  
(c) 5 (d) 6
17. The loop which never ends is called. (AJK/G1-14) (SGD-18) (BWP-15) (AJK-19) (MTN-19) (SWL-14)  
(a) Infinite loop (b) Running loop  
(c) Continuous loop (d) Nested loop
18. A loop with in another loop is called: (AJK/G1-14) (RWP-14)  
(a) Outer loop (b) Try loop  
(c) Complex loop (d) Nested loop
19. Which statement is used to move the control on the start of loop body? (GRW-14)(AJK/G1-15)(AJK/G1-15)  
(a) Break (b) Continue  
(c) Switch ( ) (d) Getch ( )
20. In which loop the condition comes after the body of the loop? (AJK/G1-16) (MTN-19)  
(a) For ( ) (b) While ( )  
(c) Do-while ( ) (d) Nested for ( )
21. Which one is a loop structure? (D.G.K-14)  
(a) if ( ) (b) if ( ) ... else  
(c) switch ( ) (d) for ( )
22. The do ... while ( ) loop structure always ends with: (D.G.K-14)  
(a) Comma (b) Semi colon  
(c) Colon (d) Brace
23. While loop is also called: (FSD-14) (LHR-14) (SGD-19) (FSD-19)  
(a) conditional loop (b) Do - while loop  
(c) For loop (d) All these
24. This statement causes a loop to terminate early: (LHR-14) (RWP-19) (SWL-19)  
(a) Exit (b) Terminate  
(c) Break (d) All of these
25. Which is an example of multiple branches from single expression? (SWL-14)  
(a) If statement (b) Switch statement  
(c) While loop (d) For-loop

## SECTION II

## SHORT QUESTIONS ANSWERS

**From PTB Exercise:**

1. Define a loop. How many loops are available in C? Compare the following loops :

**Ans.** In computer programming, a **loop** is a sequence of instructions that is continually repeated until a certain condition is reached. Typically, a certain process is done, such as getting an item of data and changing it, and then some condition is checked such as whether a counter has reached a prescribed number.

There are **three types** of loops: for, while, and do...while. Each of them has their specific uses. They are all outlined below. The variable initialization allows you to either declare a variable and give it a value or give a value to an already existing variable.

**a) while loop and do- while loop**

Only **difference** between these two **loops** is that, in **while loops**, test expression is checked at first but, in **do...while loop** code is executed at first then the condition is checked. So, the code are executed at least once in **do...while loops**.

**b) while loop and for loop**

In most computer programming languages, a **while loop** is a control flow statement that allows code to be executed repeatedly based on a given Boolean condition. The **while loop** can be thought of as a repeating if statement.

2. What sentinel controlled loop and how it is implemented? Discuss some of the situations where it can be useful. (LHR-19)(AJK-19)

**Ans.** Sentinel-controlled repetition is sometimes called indefinite repetition because it is not known in advance how many times the **loop** will be executed. It is a repetition procedure for solving a problem by using a **sentinel** value (also called a signal value, a dummy value or a flag value) to indicate "end of data entry". Sometimes, **loop control** may need to be based on the value of what we are processing. In this case, we **would** **sentinel-controlled** repetition. **Sentinel-controlled**.

3. Write the output of the following program fragments:

**Ans.** k=0;

while (k<=5)

```
{
printf("%3d\n",k,10-k)
```

```
k++;
```

```
}
```

**Output:**

0 1 2 3 4 5

4. Trace the output of the following price of code.

```
i=10;
```

```
for (int i=j;i<=5;i++)
```

```
printf("%d %d\n",i,j);
```

```
j-=2;
```

```
}
```

**Output:** This programmes generate syntax errors because in loop we do not initialize with any value and secondly j is undefined symbol mean no declare its data type. In this programme expression syntax error is occurred because for() loop is not properly complete.

4. Write a program that inputs a numbers and displays the message "Prime number" if it is a prime number, otherwise display "Not a prime number".

Ans. #include <stdio.h>  
#include <conio.h>

```
void
main(void)
{
 int n, d, p = 1;
 clrscr();
 printf("Enter a number ?");
 scanf("%d", &n);
 d = 2;
 while(d <= n/2)
 {
 if(n%d == 0)
 p = 0;

 d++;
 }
 if(p == 0)
 printf("Not prime number");
 else
 printf("Prime number");
}
```

Output  
Enter a number ? 7  
Prime number

5. Write a program that displays the first 15 even numbers.

Ans. #include <stdio.h>  
#include <conio.h>  
void main(void)  
{  
 int n;  
 clrscr();  
 for(n = 15, n>=1; n--)

Output of the  
program  
14  
12  
10  
8  
6  
2

```
{
 if(n%2 == 0)
 printf("%d\n", n);
}
```

6. Write a program that inputs a number , and display its table according to the following format:  
Suppose the number entered is 8, even the output will be as follows :

```
8 x 1 = 8
8 x 2 = 16
8 x 10 = 80

#include <stdio.h>
#include <conio.h>
void main(void)
{
 int num, c;
 clrscr();
 printf("Enter a number ? ");
 scanf("%d", &num);
 c = 1;
 while(c <= 10)
 {
 printf("%d X %d = %d\n", num, c, num*c);
 c = c + 1;
 }
}
```

Output of the program  
Enter a number ? 8  
8 X 1 = 8  
8 X 2 = 16  
8 X 3 = 24  
8 X 4 = 32  
8 X 5 = 40  
8 X 6 = 48  
8 X 7 = 56  
8 X 8 = 64  
8 X 9 = 72  
8 X 10 = 80

7. Write a program using do-while loop that repeatedly prompts for and takes input until a value in the range 0 through 15 inclusive is input. The program should add all the values before exiting the loops and display there sum at the end.

Ans.

```
#include <stdio.h>
#include <conio.h>
main()
{
 int c, s;
 clrscr();
 c = 1;
 s = 0;
 while(c<=5)
 {
 printf("%d\n", c);
 s = s + c;
 c++;
 }
 printf("sum of five numbers = %d\n", s);
}
```

Output of the Program

```
1
2
3
4
5
Sum of five number =
15
```

From Punjab Board:

- 1) Write down the syntax of do-while-loop.

Ans. Syntax of 'do-while' loop with example.

(MTN/G1-15)(RWP-14)

The general syntax of do-while is as follows:

```
do
{
 group of statement;
 ++ or --
}while (condition);
```

To display numbers from 1 to 5, a set of statement using do-while loop is written as:

```
int c = 1;
do
{
 printf("%d", c);
 c++;
}while(c<=5);
```

- 2) While does a programmer decide to use a while loop versus a for loop?

(KSMR/G1-16)

Ans. The for loop is often used when you usually know how many times you would like the program, which means it will run that program until the number of times is complete before it terminates itself. The while loop works in a similar manner but requires a conditional statement. benefit of the while loop is when you are unsure how many iterations are required to complete the given expression.

- 3) Define nested loop.

(SWL/G1-15)(AJK/G1-14)(SGD-14)

Ans. A loop inside the body of another loop is called nested loop. The loop that contains another loop in its body is called outer loop. The loop used the body of outer body of outer loop is called inner loop.

- 4) Define While Loop. (BWP-14)

Ans. The "while" loop is a conditional loop structure. It is used to execute a statement or a set of statement as long as the given condition remains true. This loop structure is used when the programmer does not know in advance the number of repetition of loop.

The general syntax of "while" is as follows:

```
while (condition)
{
 group of statements;
}
```

Example:- To display numbers from 1 to 15, a set of statement using while loop is written as;

```
int c = 1
```



```

while(c<5)
{
 printf("%d\n", c);
 c++;
}

```

5) Write two uses or advantages of loop. (AJK-17)

Ans.

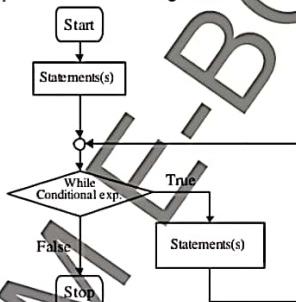
- I. The advantages of loops are
- II. Redundancy of code is less.
- III. Used to reduce the code.
- IV. Effectively solves all the difficult codes using **loops** very easily, especially pattern programs.
- V. Generally we use **loops** in the case of repetitive occurrence of some processes.

6) Make a flowchart of while loop.

(SWL-14) (LHR-14) (FSD-14)

Ans. Flowchart of 'while' Loop:

The flowchart of 'while' loop structure is shown in figure below.



### Flowchart of the while loop

#### Working of 'while' Loop:

When a 'while' loop statement is executed, the condition is evaluated first. If the condition is true, then **body of loop** is executed. After executing the body of loop, the execution control goes back to the 'while' statement and evaluates the condition again. If the given condition is still true, the **body of loop** is executed again. This process is repeated. When the given condition becomes false at any stage during execution, the loop is terminated. After terminating the loop the control shifts to the statement that comes immediately after the body of loop.

The body of loop must contain a statement used to control the repetition of loop. Usually, a statement is given in the body of loop to change the value of control variable so that the condition becomes false, the loop never terminates. It becomes an **infinite loop**.

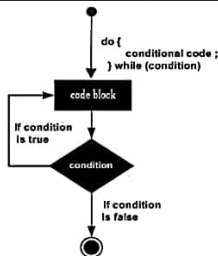
7) Define "do-while" loop. (MTN/G1-15)

Ans. In most computer programming languages, a **do while loop** is a control flow **statement** that executes a block of code at least once, and then repeatedly executes the block, or not, depending on a given boolean condition at the end of the block. The **do while** construct consists of a process symbol and a condition.

```

do
{
 group of statement;
 ++ or --;
}while (condition);

```



To display numbers from 1 to 5, a set of statement using do-while loop is written as;

```

int c = 1;
do
{
 printf("%d\n", c);
 c++;
} while(c<=5);

```

8) Write the syntax of while loop.

Ans. The general syntax of "while" is as follows:

```

while (condition)
{
 group of statements;
 ++ or --;
}

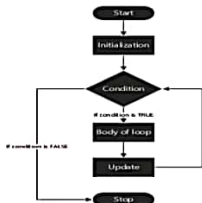
```

Example:- To display numbers from 1 to 15, a set of statement using while loop is written as;

```

{
int c = 1;
while(c<=15)
{
 printf("%d\n", c);
 c++;
}
}

```



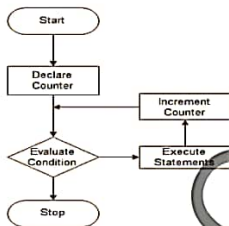
9) What is goto statement (FSD/G1-16) (FSD/G1-15)

Ans. **goto** is a jumping statement in c language, which transfer the program's control from one statement to another statement (where label is defined). **goto** can transfer the program's within the same block and there must a label, where you want to transfer program's control.

10) What is a counter-controlled loop?

(AJK/G1-16)(AJK/G1-15)(KSMR/G1-16)(FSD-19)

Ans. Counter-controlled repetition requires a control variable (or loop counter) the initial value of the control variable. The increment (or decrement) by which the control variable is modified each time through the loop (also known as each iteration of the loop)



11) What is for loop and syntax of "for loop".

(BWP-15)(MTN/G1-15)(D.G.K/G1-15)(GRW-14)(AJK/G1-14)(LHR/G1-15)(GRW-19)

Ans. Syntax of For Loop

for (initialization; condition; increment/decrement)

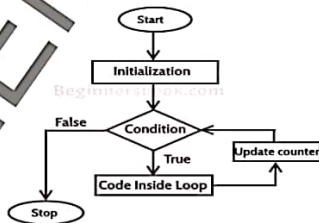
{ group of statements;

}

To display numbers from 1 to 5, a set of statement using for loop is written as;

```
for(c=1; c<=5; c++)
```

```
printf("%d\n", c);
```



**Definition of for loop:** In Loop, the statement needs to be written only once and the loop will be executed 10 times as shown below. In computer programming, a loop is a sequence of instructions that is repeated until a certain condition is reached.

12) Define Loop. List three loops in C.

(BWP-15) (GRW/G1-15)

Ans. Loops are control structures used to repeat a given section of code a certain number of times or until a particular condition is met. There are three loops in C programming: for loop while loop and do-while loop.

## 13) Differentiate between while loop and do while loop.

(LHR/G1-15)

Ans.

| The 'while' Loop Structure                                                                                  | The 'do-while' Loop Structure                                                                                                                       |
|-------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Test condition comes before the body of loop.                                                            | 1. Test condition comes after the body of loop.                                                                                                     |
| 2. At the beginning of loop, condition is evaluated and then body of loop is executed if condition is true. | 2. The condition is evaluated after executing the body of loop. The body of the loop must be executed at least once even if the condition is False. |
| 3. Semicolon (;) is not given after the while (condition).                                                  | 3. Semicolon (;) is given after the while(condition).                                                                                               |

**While versus Do-While Loops**

## 14) What is iteration?

(AJK/G1-14)

Ans. With respect to computing, **iteration** is the process of going through a set of operations that deal with computer code. For example, in a computer program, one form of **iteration** is a loop. A loop repeats code until a certain condition is met. Each time the computer runs through a loop, it is referred to as an **iteration**.

## 15) What is loop?

(AJK/G1-14)

Ans. In computer programming, a **loop** is a sequence of instructions that is continually repeated until a certain condition is reached. Typically, a certain process is done, such as getting an item of data and changing it, and then some condition is checked such as whether a counter has reached a prescribed number.

## 16) What are the differences of for ( ) loop as compared to while ( ) loop. (D.G.K/G1-15)

Ans.

|            | FOR Loop                                                                            | WHILE Loop                                                                                             |
|------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| Definition | To execute the certain block of code repeatedly until some conditions are satisfied |                                                                                                        |
| Used for   | When you know the number of iterations beforehand                                   | When you have an idea about the range of values on which to iterate but are unsure of the exact number |
| Loop       | Will loop until the number of mentioned iterations are completed                    | Will loop until the condition presented is completed                                                   |

The only difference between a for loop and a while loop is readability. If you have an initialization phase, a termination condition, and an increment condition, a for loop is better although a range-for may have better optimization. ... So in principle the while is both more efficient and more powerful than the for.

in general a while loop is used if you want an action to repeat itself until a certain condition is met i.e. if statement. An for loop is used when you want to iterate through an object. i.e. iterate through an array.

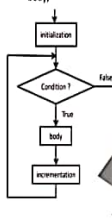
## 17) Write the purpose of continue statement. (GRW-19)

Ans. **Continue statement** is mostly used inside loops. Whenever it is encountered inside a loop, control directly jumps to the beginning of the loop for next iteration, skipping the execution of statements inside loop's body for the current iteration.

## 18) Define do-while Loop. (SGD-19)

Ans. The **do while loop** checks the condition at the end of the loop. This means that the statements inside the loop body will be executed at least once even if the condition is never true. The **do while loop** is an exit controlled loop, where even if the test condition is false, the loop body will be executed at least once.

for( initialization; condition; incrementation )  
body;



## 19) Write three repetition statements. (AJK/G1-14)

Ans. A **repetition statement** (also called a **looping statement** or a **loop**) allows you to specify that a program should repeat an action while some condition remains true. There are 3 types of Loop in C language, namely: while loop, for loop and do while loop,

## 20) What is nested loop? (RWP-19)

Ans. The placing of one loop inside the body of another loop is called **nesting**. When you "nest" two loops, the outer loop takes control of the number of complete repetitions of the inner loop. While all types of loops may be nested, the most commonly nested loops are for loops.

```

for(num2 = 0; num2 < 9; num2++)
{
 for(num1 = 0; num1 < 9; num1++)
 {
 cout << num2 << " " << num1 << endl;
 }
}

```

outer loop (points to the outer for loop)  
inner loop (points to the inner for loop)

## 21) Define infinite loop. (MTN-19)

Ans. An **infinite loop** (sometimes called an **endless loop**) is a piece of coding that lacks a functional exit so that it repeats indefinitely. ... Usually, an **infinite loop** results from a programming error - for example, where the conditions for exit are incorrectly written.

## 22) What is counter control loop? (SWL-19)

Ans. A **count-controlled** repetition will exit after running a certain number of times. The **count** is kept in a variable called an **index** or **counter**. When the index reaches a certain value (the **loop bound**) the loop will end.)

Example: A while loop is an example of counter controlled loop.

Programming

1. Write output.

```
int x = 5 , y = 3 ;
do
{
x = x * 2 ;
```

```
y = y + 2 ;
}
while (y < 7) ;
printf (" % d " , x) ;
```

**Output**

20

(D.G.K/G1-16)

2. Convert following loop code into while loop code.

```
for (i = 10 ; i > 0 ; i --)
{
printf (" i = % d " , i) ;
}
```

**Correction:**

```
i = 10;
while (i >= 0)
{
printf ("i = %d", i);
i --;
}
```

(D.G.K/G1-16)

3. Trace the errors of the following code:

```
(1)Void main ()
{
int x , y = 5 ;
for (x = 0 ; x < 3 ; x ++) ; (2)
if (y >= 5) ; (3)
printf (" % d " , x) ;
}
```

**Correction:**

- (1) V used in upper case.  
 (2) and (3) syntax error occurs because ; is not used after both statements.

(BWP/G1-16)

4. Convert the following do - while loop in for loop.

```
int c = 2 ;
do
{
printf (" % c " , c) ;
c ++ ;
}while (c <= 5) ;
```

(BWP/G1-16)

**Conversion:**

| Do-while                                                           | For-loop                                                                  |
|--------------------------------------------------------------------|---------------------------------------------------------------------------|
| <pre>int c = 2; do {printf ("%c",c); c++; }while (c&lt;= 5);</pre> | <pre>{ int c ; for (c = 2 ; c &lt;= 5 ; c + +) printf ("%c" , c); }</pre> |

5. Predict the output from the following code:

```
int n ;
clrscr () ;
for (n = 5 ; n >= 1 ; n --)
printf (" % d \n " ,n) ;
getch () ;
```

**Output**

|   |
|---|
| 5 |
| 4 |
| 3 |
| 2 |
| 1 |

(FSD/G1-16)

6. Convert following code into while loop?

```
void main (void)
{
int i;
for (i = 1; i <= 10 ; i ++)
printf ("loop");
}
```

**Convert:**

```
int i;
i = 1;
while (i <= 10)
{
printf ("loop");
i ++ ;
}
```

(GRW/G1-16)

7. Write down output of the following code:

```
void main (void)
{
int i;
for (i = 1; i <= 2 ; i ++)
printf (" % d " , i * i);
}
```

**Output**

|    |
|----|
| 14 |
|----|

(GRW/G1-16)

8. Predict the output of the following piece of code:

```
int i = 1;
while (i <= 5)
printf (" Pakistan ");
i ++ ;
}
```

**Output**

|                                          |
|------------------------------------------|
| PakistanPakistanPakistanPakistanPakistan |
|------------------------------------------|

(LHR/G1-16)



9. Convert the following loop in do-while loop:

```
for (i = 3 ; i <= 39 ; i += 6)
```

```
{
 printf ("%d \n", i);
}
```

**Convert:**

```
i = 3 ;
```

```
do
```

```
{
 printf ("%d \n", i);
 i += 6;
 while (i <= 39);
}
```

10. Trace output:-

(MTN/G1-16)

```
int i, j = 10;
```

```
for (i = 1 ; i <= 5; I++)
```

```
{
 printf ("\n Pakistan");
}
```

**Output :**

Pakistan

Pakistan

Pakistan

Pakistan

Pakistan

11. Convert the following code into while loop:-

(MTN/G1-16)

```
for (int i = 1; i <= 10 ; i++)
```

```
{
 Printf ("\n Pakistan");
}
```

**Convert:**

```
int i = 1;
```

```
while (i <= 10)
```

```
{
 printf ("\n Pakistan");
 n ++ ;
}
```

12. What is the output of following code?

(RWP/G1-16)

```
int x = 10;
```

```
for (x = 8; x >= 1; x--)
```

```
printf ("% d\n", x % 2);
```

**Output:**

|   |
|---|
| 0 |
| 1 |
| 0 |
| 1 |
| 0 |
| 1 |
| 0 |
| 1 |

13. Predict the output from the following code:

```
int n;
n = 1;
clrscr();
while (n <= 10)
{
 printf("%d\n", n);
 n++;
}
```

(SGD/G1-16)

Output

```
1
2
3
4
5
6
7
8
9
10
```

14. Trace the output.

(RWP/G1-15)

```
void main()
{
 int a;
 for (a=1; a<=100; a=a+3)
 printf("%d\n", a);
 getch();
}
```

Output

```
1
4
7
10
.
.
.
100
```

15. Trace the output

```
int k = 0;
while (k <= 5)
{
 printf("%3d %3d\n", k, 10-k);
 k++;
}
```

Output

```
0 10
0 10
0 10
0 10
```

```


```

Continue infinite

16. Determine the output of the following code:

```
for (int i = 2 ; i <= 10 ; i = i + 2)
 printf (" % d \n" , i);
```

Output

```
2
4
6
8
10
```

(AJK/G1-14)

17. What is the output of the following code?

```
void main (void)
{
 int n = 1;
 while (n <= 5)
 {
 printf("Islam Zindabad")
 n = n + 1;
 getch () ;
 }
}
```

Output

```
Islam ZindabadIslam ZindabadIslam ZindabadIslam ZindabadIslam Zindabad
```

(FSD-14)

18. Trace the output.

(GRW-14)

```
int a = 1;
while (a <= 6)
{
 printf ("a = %d" , a);
 a++ ;
}
```

Output

```
a = 1
a = 2
a = 3
a = 4
a = 5
a = 6
```

19. What is output of the following code: (SWL-14)

```
void main (void)
{
 int n = 1;
 while (n <= 5)
 {
 printf ("Pakistan");
 n = n + 1
 }
 getch () ;
}
```

Output

```
PakistanPakistanPakistanPakistanPakistan
```

20. What is the final value of x after executing the following code:

(BWP-14)

```
for (int x = 0 ; x < 10, x ++)
```

Output

```
9
```

21. Predict the output of the following code:

(AJK-17)

```
int i;
for (i = 1 ; i <= 10 ; i + 2)
 printf (" %d" , i);
```

Output

```
1
3
5
7
9
```

**22. Convert the following while loop to for-loop.**

(AJK-17)

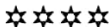
```
int c = 0;
while (c <= 5)
{
 printf("\n %d ", c);
 c++;
}
```

**Convert:**

```
{
 int c;
 for (c = 0; c <= 5; c++)
 {
 printf("\n %d ", c);
 }
}
```

**SECTION III****LONG QUESTIONS**

- 1) Write a program in C-Language to print the sum of following series. (D.G.K - 19)  
1    3        6        9        12        15
- 2) Write a program that displays counting from 10 to 1 using for Loop. (GRW-19)
- 3) Write a program that inputs a number from the user and displays the factorial of that number. (LHR-19) (SGD-19)
- 4) Write a program in C that input a number from the user and displays the table of that number using for loop. (MTN-19) (FSD-19)
- 5) Define while loop and differentiate while and do-while loop in detail. (RWP-19)
- 6) What is Do-While loop? Discuss flowchart syntax and example of Do-While loop in detail. (RWP-19)
- 7) Write a program that print odd numbers from 11 to 49 using loop. (SWL-19)
- 8) Write a program in C that displays the following series using while loop. (AJK-19)  
1    3    5    7    9    11    13    15    17    19    21    23
- 9) What are loops? Discuss different loops provide by C. (LHR/G1-15)
- 10) What is sentinel controlled loop and how it implemented? Discuss some of the situations where it can be useful? (SWL/G1-15)
- 11) Define "while" loop. Write its syntax, Draw flow chart and explain its working with the help of example. (SWL-14) (SGD/G1-16) (RWP/G1-16) (RWP/G1-15) (D.G.K-14) (BWP-15) (GRW-14) (LHR-14) (D.G.K/G1-15) (MTN/G1-16)
- 12) What is nested loop? Explain with example in C-language. Also describe a situation where nested loop can be applied? (MTN/G1-15) (BWP-14) (AJK/G1-16) (AJK/G1-14) (FSD/G1-15) (MTN/G1-15) (KSMR/G1-16) (LHR/G1-16) (GRW/G1-16) (FSD/G1-16) (FSD/G1-15) (FSD/G1-16) (AJK/G1-15) (FSD-14) (RWP/G1-16)
- 13) Define for loop? Give its syntax and flow chart. Also explain its working using an example \ give reason programmer decide to use a while loop versus a for loop? (D.G.K/G1-16) (RWP-14) (AJK/G1-16) (FSD/G1-16) (BWP/G1-16) (SGD-14) (AJK-17)



# FUNCTIONS

## SECTION I

### MULTIPLE CHOICE QUESTIONS

#### From PTB Exercise:

- Function prototypes for built-in functions are specified in:  
(a) source files (b) header files  
(c) object files (d) image file
- Global variables are created in: (D.G.K/G1-15) (SWL/G1-15) (MTN-19) (RWP-19)  
(a) RAM (b) ROM  
(c) Hard Disk (d) Cache
- Which of the following is true about a function call?  
(a) stops the execution of the program  
(b) transfers control to the called function  
(c) transfers control to the main function.  
(d) resumes the execution of the program.
- Which of the following looks for the prototypes of functions used in a program?  
(a) linker (b) loader  
(c) compiler (d) parser

#### From Punjab Board:

- Another name for built-in function is: (AJK-19)  
(a) User-defined function (b) Library function  
(c) Arithmetic function (d) Logical function
- Multiple arguments to functions are separated by: (SWL-19)  
(a) comments (b) semicolons  
(c) colons (d) commas
- The parameters in function declaration. (GRW-19)  
(a) actual parameters (b) formal parameters  
(c) returned parameters (d) call parameters
- The statement that activates a function is known as: (D.G.K - 19)  
(a) Function design (b) Function definition  
(c) Function declaration (d) Function
- The process of sending an argument to a function is called: (FSD-19) (SGD-19)  
(a) Sending (b) Filtering  
(c) Delivering (d) Passing
- The scope of variable refers to its: (SGD-14)  
(a) Selected (b) set selected  
(c) Index (d) Set index

7. The scope of variable refers to its: (LHR-19)  
(a) Length (b) Name  
(c) Accessibility (d) Data type
8. Function prototype for built-in functions are variable denotes: (LHR/G1-15)  
(a) Source file (b) Header file  
(c) Object file (d) Image file
9. A built-in function. (RWP-18)  
(a) cannot be redefined (b) can be redefined  
(c) cannot return value (d) should be redefined
10. Local variables are also called; (AJK-17)(SWL-14)  
(a) Static variable (b) Automatic variable  
(c) Register variable (d) Run time variable
11. Memory is allocated to a local variable at the time of its: (FSD-14)/ (LHR-14)  
(a) Declaration (b) Destruction  
(c) Definition (d) First reference

## SECTION II

### SHORT QUESTIONS ANSWERS

#### From PTB Exercise:

1. What is function? How many types of function are used in C? Discuss the difference between them.

Ans. A function is a group of statements that together perform a task. A function declaration tells the compiler about a function's name, return type, and parameters. A function definition provides the actual body of the function. The C standard library provides numerous built-in functions that your program can call. There are two type of functions user defined and built in functions

3. How is function call is made C program? Discuss briefly. (LHR-14) (SWL-14) (FSD-14) (SGD-14) (BWP-15) (BWP-14) (RWP-14) (D.G.K-14) (GRW/G1-16) (GRW-19) (D.G.K - 19)

Ans. A function is a group of statements that together perform a task. A function declaration tells the compiler about a function's name, return type, and parameters. A function definition provides the actual body of the function. The C standard library provides numerous built-in functions that your program can call.

#### DISCUSS BRIEFLY

1. It is a process that is used to invoke (execute) a function to perform a specific task.
  2. A function can be called at any point in the main program.
  3. A function can be called with its name and correct sequence of parameters.
  4. When function call statement is executed. Then the control is transferred to the function body. And the statements in function body are executed.
  5. After executing the last statements in the function, the control is transferred to the calling function.
4. Write a program that call two functions Draw \_ horizontal to draw into parallel horizontal lines, and the function Draw \_ Vertical to draw two parallel verticals lines.

Ans. `void Draw_Horizontal();`

`void Draw_Vertical();`

`void main()`

`{`  
`clrscr()`

`Draw_Horizontal();`

`Draw_Vertical();`

```

Draw_Horizontal();
 getch();
}

void Draw_Horizontal()
{
for(int i=1; i<=25;i++)
 printf("+");
}

void Draw_Vertical()
{
for(int j=1; j<=5;j++)
 printf("*\t\t*\n");
}

}

```

5. Write a program that prompts the user to enter a number and then reverse it. Write a function Reverse to reverse the number. For example if user enters 2765, the function should reverse the number so that becomes 5672. The function should accept the number as an input parameter and return the reversed number

Ans.

```

int reverse(int n);
void main()
{
 int x, y;
 clrscr();
 printf("ENTER AN INTEGER");
 scanf("%d",&x);
 y= reverse(x);
 printf("Actual No Is=%d\n",x);
 printf("Reverse No is =%d",y);
}

```

```

getch();
}

int reverse (int n)

```

```

{
 int t, r;
 t=n;
 r=0;
 for(t!=0;t/10)
 r=10*r+t%10;
 return r;
}

```

#### OUTPUT

```

ENTER AN INTEGER 5678
Actual No Is= 5678
Reverse No is = 8765

```

6. Write a function that has an input parameter num and returns value of if num is prime, otherwise returns a value of 0.

Ans.

```

int is_Prime(int n);
void main(void)

```



```
{
int num,ans;
clrscr(0);
printf("ENTER A NUMBER");
scanf("%d",&num);
ans=is_Prime(num);
printf("YOUR GIVEN NO IS %d",ans);
getch();
}
```

```
int is_prime(int n) {
```

```
Int p,j;
p=1;
for(j=2;j<n;j++)
if(n%j==0)
{
p=0;
break;
}
If(p==1)
return 1;
else
return 0;
}
```

#### OUTPUT

ENTER A NUMBER 17  
YOUR GIVEN NO IS 1

7. Write a program that prompts the user to enter the numbers and call a function Fraction to compute its fractional .Write the function Fractional that has one 1 input parameters and returns the fraction of a number pass to it.

Ans. long factorial(int ):

```
void main()
{
Int n;
long fact;
clrscr();
printf("Please enter a No for Factorial");
scanf("%d", &n);
fact=factorial(n); //function call
printf("Factorial of your number=%d",fact);
getch();
}
```

```
long factorial(int n)
```

```
{
int i ;
long f;
f=1;
```

```
for(i=1;i<=n;i++)
f=f*i;
return f;
}
```

**OUTPUT**

Please enter a No for Factorial 5  
 Factorial of your number=120

8. Write a function GCD that has two input parameters and returns the greatest common divisor of the two numbers passed to it. Write a complete C program that inputs two numbers and call the functions GCD to compute the greatest common divisor of the number entered

Ans. 

```
int gcd(int, int);
void main()
{
 int a,b,r;
 clrscr();
 printf("please enter no of A & B ");
 scanf("%d%d", &a,&b);
 r=gcd(a,b);
 printf("GCD=%d",r);
 getch();
}
```

```
int gcd(int x,int y)
```

```
{
 int i,m;
 m=1;
 for(i=2;i<=x;i++)
 {
 If(x%i==0&&y%i==0)
 m=i;
 }
 return m;
}
```

**OUTPUT**

please enter no of A & B 5 10  
 GCD=5

**From Punjab Board:**

1. What is function and function Call Statement?

(LHR-14) (SWL-14) (FSD-14) (SGD-14) (BWP-15) (BWP-14)  
 (RWP-14) (D.G.K-14) (GRW/G1-16) (GRW-19) (D.G.K-19)

Ans. Inside the function, the address is used to access the actual argument used in the call. This means that changes made to the parameter affect the argument. By default, C uses call by value to pass arguments. In general, it means the code within a function cannot alter the arguments used to call the function.

2. Differentiate between local and global variables?

(KSMR/G1-16) (FSD/G1-16) (LHR/G1-15) (FSD/G1-15) (AJK/G1-16) (GRW/G1-16)  
 (RWP/G1-16) (LHR/G1-16) (D.G.K-14) (BWP-14) (MTN-19) (SGD-19) (GRW-19)

Ans. Local variables are declared inside the function while global variables are declared outside the function. Local variables can be used in the function in which they are declared while global variables can be used by all functions of the program.

3. Define the scope of local variable. (D.G.K/G1-16) (MTN/G1-16)

Ans. The portion of a program where a variable is used (accessed) is known as its scope. A local variable declared in a function can only be used in that function only.

**4. What is life time of local variable? (SWL/G1-15)**

**Ans.** The time period which a local variable exists in memory is called lifetime of a local variable. The local variables are created when control enters into function. They are destroyed from memory when control returns back to calling function.

**5. What are Actual Parameters?(LHR-14) (SGD-14) (LHR-19)**

**Ans.** The parameters used in the function call are called actual parameters/ argument. These are the actual values that are passed to the function. The actual parameters may be in the form of constant values or variable.

**6. Differentiate between function definition and function declaration / function prototype?**

(RWP/G1-15) (BWP/G1-16) (AJK-17) (SGD/G1-16) (D.G.K/G1-16) (FSD-19) (LHR-19) (RWP-19)

**Ans.**

| Function Definition                                                      | Function Declaration                                                                                                                              |
|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. It is a set of statement that is written to perform a particular task | 1. It is a single statement that specifies the name of function, its parameters along with their data types and the return data type of function. |
| 2. It must be defined before or after the main() function.               | 2. Its use is optional if function definition is placed before the main() function.                                                               |
| 3. It cannot be written inside the main() function.                      | 3. It can be written within the main() function.                                                                                                  |

**7. Define function. Why is it used in a program?**

(FSD/G1-16) (FSD/G1-16) (BWP/G1-16) (AJK-19) (RWP-19)

**Ans.** A piece of code within a large program, which is written to perform a specific task, is called function. It is independent of the remaining code of the program. Functions are the building blocks of C program.

Using functions a program can be divided into different units. Each unit can be managed easily. A piece of code for each unit is written as a function. The same piece of code is written only once in the program as a function. This function can be called again and again from multiple places in a program to perform a particular task. The size of program is reduced. The program can easily be modified and debugged.

**8. Write the use of Function declaration / prototype? (KSMR/G1-16) (AJK/G1-16)**

**Ans.** Specifying the structure or model of a user defined function is called the function prototype. The function prototype is also referred to as function declaration. The function declaration statement is used to declare a user defined function. This statement only provides the information to the compiler about the user-defined function.

**9. Define built in function. (LHR/G1-15)(MTN/G1-15)(AJK/G1-15)(SWL-19)(SGD-19)(MTN-19)(D.G.K – 19)**

**Ans.** The predefined function that are that are part of the programming language are called built-in function. The built-in function are called library function. These function can be used for different purposes. C-language has a large number of built-in functions. The functions are defining in the header files. Four built-in function that are commonly used in C are printf(), scanf(), clrscr(), and getch().

**10. What do you know about parameter passing in the function? (D.G.K/G1-15) (RWP/G1-15)**

**Ans.** parameter passing The mechanism used to pass parameters to a procedure(subroutine) or function. The most common methods are to pass the value of the actual parameter (call by value), or to pass the address of the memory location where the actual parameter is stored (call by reference).

**11. List four benefits of using functions.**

(SGD/G1-16) (AJK/G1-14) (SWL/G1-15) (BWP-15) (GRW-14) (RWP-14) (AJK-19)

**Ans.** Some benefits of using in a program are:

1. Easy to write a program
2. Easy to understand and modify the program
3. Eliminate duplicate code
4. Reusability

**12. What are the two types of functions in C language?**

(RWP/G1-16)(SWL-19)(SGD-19)(MTN-19) (D.G.K – 19)

**Ans.** There are two types of function in C-language. These are built in function and user- defines functions

**Built-In Functions** The predefined function that are that are part of the programming language are called built-in function .The built-in function are called library function .These function can be used for different purposes. C-language has a large number of built-in functions. The functions are defining in the header files. Four built-in function that are commonly used in C are printf(),scanf(),clrscr(),and getch().

**User-define function** The function written by a user or programmer are called user- defined function .These function are also called programmer-define function. A user-define function is written to programmer-define functions. A user-defined function is written to perform a specific task.

**13. What is lifetime of a variable? (MTN/G1-16)**

**Ans.** The time period for which a variable exists in memory is called lifetime of a variable. Its lifetime starts when it is created in memory and ends when it is destroyed from memory. A variable can only be used during its lifetime.

**14. Define function body. (LHR/G1-16)**

**Ans.** The set of statement that are written between the curly braces just after the header of function definition is called function body. The statements are written to perform a specific task.

**15. How does a function make programming easier? (AJK-17) (MTN/G1-15)**

**Ans.** The program is divided into small parts known as functions. The complexity of the program is reduced. It makes easier to write program. It is also easy to make changes and detect errors when large program is divided into functions.

**16. Define function header. (AJK/G1-14) (FSD/G1-15) (AJK/G1-15)**

**Ans.** The first line of function definition is called function header. It also called function declaration. It is same as the function declaration but it is not terminates by semicolon (;).

**17. What is meant by scope of a variable? (GRW-14)**

**Ans.** A scope in any programming is a region of the program where a defined variable can have its existence and beyond that variable it cannot be accessed. There are three places where variables can be declared in C programming language = Inside a function or a block which is called local variables.

**18. What is a string? (AJK-19)**

**Ans.** A string is a data type used in programming, such as an integer and floating point unit, but is used to represent text rather than numbers. It is comprised of a set of characters that can also contain spaces and numbers. For example, the word "burger" and the phrase "I ate 3 burgers" are both strings.

**19. What is meant by scope of variable? (SWL-19)**

**Ans.** Scope refers to the visibility of variables. In other words, which parts of your program can see or use it. Normally, every variable has a global scope. Once defined, every part of your program can access a variable. It is very useful to be able to limit a variable's scope to a single function.

**20. What is meant by fgets function? (SWL-19)**

**Ans.** The C library function char \*fgets(char \*str, int n, FILE \*stream) reads a line from the specified stream and stores it into the string pointed to by str. It stops when either (n-1) characters are read, the newline character is read, or the end-of-file is reached, whichever comes first.

**SECTION III****LONG QUESTIONS**

No long questions in this Chapter by Punjab Text Book Board.



# FILE HANDLING IN C

## SECTION I

### MULTIPLE CHOICE QUESTIONS

#### From PTB Exercise:

1. A file is stored in:  
(a) ram (b) hard disk  
(c) rom (d) cache
4. Which of the following functions is used to write a string to file? (GRW-19) (SWL-19)  
(a) puts() (b) putc()  
(c) fputs() (d) fgets()

#### From Punjab Board:

1. In the statement FILE\* fp, the \* represents: (MTN-19) (AJK-19) (D.G.K - 19)  
(a) Variable (b) Parameter  
(c) Argument (d) Pointer
2. A binary stream is sequence of:- (MTN/G1-15)  
(a) Bits (b) Bytes  
(c) Kilobytes (d) Giga bytes
3. The following function is used to read character from a file. (RWP-18)  
(a) putc ( ) (b) getc ( )  
(c) fputs ( ) (d) fgets ( )
4. The fopen ( ) function uses parameters. (SGD-18)  
(a) 1 (b) 4  
(c) 3 (d) 2
5. The character conversion may occur in: (D.G.K-14)  
(a) Text Stream (b) Binary Stream  
(c) Output Stream (d) Input Stream

## SECTION II

## SHORT QUESTIONS ANSWERS

**From PTB Exercise:**

1. What is stream? Illustrate the difference between text and binary streams.

**Ans.** In computer science, a **stream** is a sequence of data elements made available over time. A **stream** can be thought of as items on a conveyer belt being processed one at a time rather than in large batches. We have already operated on a lot of **text files** and a few **binary files**. The major difference between these two is that a **text file** contains textual information in the form of alphabets, digits and special characters or symbols. On the other hand, a **binary file** contains bytes or a compiled version of a **text file**.

2. How many modes are there for opening a file in C? Discuss the characteristics of different file opening modes.

**Ans.** File can be opened in basic 3 modes : Reading Mode, Writing Mode, Appending Mode. If File is not present on the path specified then New File can be created using Write and Append Mode. Generally we used to open following types of file in C

File Opening Mode Chart:

| Mode | Meaning             | open Returns if FILE-                                          |
|------|---------------------|----------------------------------------------------------------|
|      |                     | Exists                                                         |
| r+   | Reading + Writing   | New data is written at the beginning overwriting existing data |
| w+   | Reading + Writing   | Over write on Existing                                         |
| a+   | Reading + Appending | New data is appended at the end of file                        |

**From Punjab Board:**

1. Define stream.

(SGD-14) (SGD-19) (FSD-19)

**Ans.** In C, the **stream** is a common, logical interface to the various devices that comprise the computer. In its most common form, a **stream** is a logical interface to a file. As C defines the term "file", it can refer to a disk file, the screen, the keyboard, a port, a file on tape, and so on.

Simply a logical interface between data file or input/output device and program is called stream. It is a general name given to a flow of data. The data flows as a sequence of bytes. The stream may be input stream or output stream.

2. What do you mean by binary stream? (RWP/G1-15) (SWL/G1-15) (SGD/G1-16) (RWP-14) (GRW/G1-16) (AJK/G1-14) (FSD/G1-15) (FSD/G1-16) (GRW-19)

**Ans.** A binary stream is a sequence of byte. In a binary stream, there is one-to-one relationship between the bytes and those saved on the external device. The number of bytes written or read is the same as the number of bytes on the external device. Or

A **binary stream** consists of one or more bytes of arbitrary information. You can write the value stored in an arbitrary object to a (byte-oriented) **binary stream** and read exactly what was stored in the object when you wrote it.

3. Write the use of new line marker. (BWP/G1-16)

**Ans.** In a C-program, the escape sequence '\n' is used to insert a new line on the output device and in a text data file. It means that '\n' indicates the ends-of-line character or new-line marker. In a text editor (such as TC editor or notepad), when enter key is pressed, a new line character '\n' is automatically inserted at the end of a line.

4. Which function is used to close a file in C language? (RWP/G1-16) (BWP-15) (GRW-14) (LHR/G1-16) (LHR-19)

**Ans.** The file (both text and binary) should be closed after reading/writing. Closing a file is performed using library function `fclose()`. `fclose(fptr);` //fptr is the file pointer associated with file to be closed.



**5. What is the use of data file? (MTN/G1-16)**

**Ans.** A logical interface between data file or input/output device and program is called stream. It is a general name given to a flow of data. The data flows as a sequence of bytes. The stream may be input stream or output stream. A flow of data from input device to computer is called input stream. For example, a program loaded in memory reads data from input device such as keyboard. Similarly, a program reads data from data file stored on disk. The data flows from keyboard or disk to computer memory. A flow of data from computer to output device is called output stream. For example, a program loaded in memory sends information to output device such as monitor, similarly, a program writes data into data file on disk. The data flows from computer memory to monitor or disk.

**Simply in short** One way to get input into a program or to display output from a program is to use standard input and standard output, respectively. All that means is that to read in data, we use scanf() (or a few other functions) and to write out data, we use printf()

**6. Compare binary and text stream / Name two types of streams used in files?**

(KSMR/G1-16) (AJK/G1-16) (RWP-19)(RWP-19)

**Ans.** There are two types of streams.

- Text stream
- Binary stream

**Text Stream:**

- A text stream is a sequences of characters.
- Character translations may occur in a text stream e.g. new line is represented as carriage return.
- There may not be one to one relation between the characters written and those on the external device.

**Binary Stream:**

- It is a sequence of bytes.
- The number of bytes written or read is always the same as those on the external device. It means there is one to one relation between the bytes written or read and those on the external device.

No character translations occur in binary stream.

| TEXT STREAM                                                                                                                                                                                                | BINARY STREAM                                                                                                                                                                               |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| A text file contains textual information in the form of alphabets, digits and special characters or symbols.                                                                                               | A binary file contains bytes or a compiled version of a text file.                                                                                                                          |
| A text file stores data in the form of alphabets, digits and other special symbols by storing their ASCII values and are in a human readable format. For example, any file with a .txt, .c, etc extension. | A binary file contains a sequence or a collection of bytes which are not in a human readable format. For example, files with .exe, .mp3, etc extension. It represents custom data.          |
| A small error in a textual file can be recognized and eliminated when seen. Whereas                                                                                                                        | A small error in a binary file corrupts the file and is not easy to detect.                                                                                                                 |
| A text stream will invoke translation of the system defined newline marker to a \n (on reading), and the reverse \n to system defined on writing.                                                          | Since the data is not human readable it also adds to the security of the content as one might not be able to get data if the structure is not known.                                        |
|                                                                                                                                                                                                            | A binary stream doesn't do this. There is perhaps an implicit assumption that a file opened in text mode might actually consist only of printable characters and cursor control characters. |

**7. What is the purpose of open statement. (D.G.K/G1-15)**

**Ans.** The fopen() function is used to open a file in memory in a specified access mode. A file pointer is associated with the file to be opened.

**Opening Mode:**

It is a string and enclosed in double quotes. Mode is a type of operation that is performed on the file

**Example:**

FILE \*fp

Fp=fopen("c:\\c\\XYZ.DAT","r");

**File Opening Modes:**

- "r" open a text file for reading, each file must exist before reading.
- "w" open a text file for writing, if the file already exists then its contents are overwritten, if it does not exist, it will be created.
- "a" open a text file for opening. Data is appended at the end of existing file. If the file does not exist, it will be created.
- "r+" opens a file for both reading and writing. The file must already exist.
- "w+" open a file for reading and writing, its contents are overwritten. If the file does not exist, it is created.
- "a+" opens a text file for both reading and appending. If the file does not exist, it is created for both reading and writing.
- The fopen function returns the null pointer, if it fails to open the file (most common reason is file does not exist).

**8. What is text file. (D.G.K-14) (BWP-14)**

**Ans.** All text file functions and types in C come from the **stdio** library. When we need text I/O in a C program, and we need only one source for input information and one sink for output information, you can rely on **stdin** (standard in) and **stdout** (standard out), and in a text stream is a sequence of characters. In a text stream, certain character conversion may occur. For example, when a sequence of characters flows from main memory to output device, a carriage return ('\r') is converted into new line. It means that there may not be a one-to-one relationship between the characters.

**9. Define file pointer. (LHR/G1-15)**

**Ans.** **File pointer** is a **pointer** which is used to handle and keep track on the files being accessed. A new data type called "**FILE**" is used to **declare file pointer**. This data type is **defined** in **stdio.h** file.

Simply we can say a file pointer is a point variable of type FILE. FILE is a special data structure, which is defined in the 'stdio.h' header file. It is used to hold the information of a data file opened in memory.

**10. Define EOF marker in file. (LHR-14)**

**Ans.** The data source is usually called a **file** or **stream**. In the C Standard Library, the **character** reading functions such as **getchar** return a value equal to the symbolic value (macro) **EOF** to indicate that an end-of-file condition has occurred.

Simply EOF stands for End-Of-File. It is a special character. It is automatically inserted at the end of each data file. When program is reading data from a sequential file, it reads data upto the required data item or end-of-line character is reached. In C, EOF identifier is used to detect the end-of-line.

**11. Which access method can access the data directly? (D.G.K/G1-16)****Ans. File Access Methods**

**Sequential Access:** A simple access method, information in a file is accessed sequentially one record after another.

**Direct Access:** Sometimes it is not necessary to process every record in a file.

**Indexed Sequential Access:** This access method is a slight modification of the direct access method.

**12. How is the end of text file indicated? (MTN-19) (D.G.K - 19)**

**Ans.** Short for **end-of-file**, EOF is a code placed by a computer after a file's last byte of data. EOF marks are helpful in data transmission and storage. Files are stored in blocks, and the **end marker** helps the computer know it has allocated enough space to store the file.

**SECTION III****LONG QUESTIONS**

No long questions in this Chapter by Punjab Text Book Board.

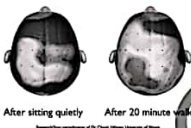




**EXAMS TIPS****1. Walk Before An Exam**

It's been proven that exercise can boost your memory and brain power. Research conducted by Dr. Chuck Hillman of the University of Illinois provides evidence that about **20 minutes exercise before an exam** can improve performance.

Composite of 20 student brains taking the same test

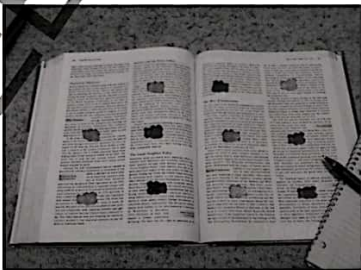
**2. Speak Out Loud Instead of Simply Reading**

Although this may make you look a little crazy, give it a go! You will be surprised how much more you can remember when you've said it out loud. **Warning:** Don't try this in a crowded library!

You're 50% more likely to remember something if you speak it out loud instead of simply reading it over and over.

**3. Reward Yourself With A Treat**

There are many ways to integrate a reward system into your habits so you learn how to study for exams more efficiently. Here's a simple way to **motivate yourself to study** with Gummy Bears:



#### 4. Teach What You Have Learned

The best way to test if you really understand something is to try to teach it to someone else. If you can't get anyone to listen to you **explain the Pythagorean Theorem**, why not teach a class of stuffed animals!



#### Get New Study Hacks on a Daily Basis

Join GoConqr for free and you'll have access to Study Groups in which students from all over the world share their secrets to better academic results

#### 5. Create Mental Associations

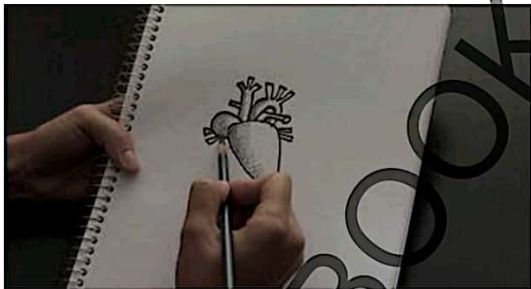
The ability to make connections is not only an easier way to remember information, but it's the **fuel of creativity and intelligence**. Steve Jobs famously said *"Creativity is just connecting things. When you ask creative people how they did something, they feel a little guilty because they didn't really do it, they just saw something"*.

**Mind Maps** are an easy way to connect ideas by creating a visual overview of different connections. Read more about the benefits of using Mind Maps to learn [here](#).



## 6. Draw Diagrams

Drawing diagrams will help you to visualise information which would be hard to describe. This creates a **visual memory in your mind which can be recalled** in an exam. You may even be asked to draw or label diagrams such as the human heart in your exam so get practicing!



## 7. Times New Roman is the Fastest Font to Read

Simply put – there's a reason why Times New Roman is the default font on most applications!

This is easy to read.

This is easy to read.

*This is not easy to read.*

This is not easy to read.

## 8. Use Apps to Block Distracting Sites

The SelfControl app helps you to avoid distractions by blocking websites for a certain amount of time. Discover more student apps to make student life easier in our blog post ["12 Student Apps You Don't Want to Miss!"](#).



## 9. Watch a Documentary on the Topic

Documentaries are an entertaining way of compacting an entire story into a short timeframe. This will help you remember key details from a story plus you may even get extra credit for mentioning that you took the initiative and watched a film about the topic!

Check out the [infographic](#) in this blog post which helps you decide which documentary to watch.

## 10. Search Google Like a Pro

Save time when researching sources online by mastering the biggest search engine in the world; Google. Follow the tips in this image to find what you need at your fingertips:

### Search Google Like a Pro

You know how to Google,  
but do you do it like a pro?

Here are a few simple yet very helpful search operators to help you  
Search Google... like a Pro

#### "Quotation Marks"

"I love you Mom"

Using quotation marks in your search terms lets you search exactly for that word. It means all your results will have your search terms in them.

#### - Dashes

dolphins -football

If you want to exclude a term from your search include a hyphen before that word.

#### ~ Tilde

music ~classes

Use tilde when you want also its synonyms to appear in the result. The above query will search for music classes, lessons, coaching etc.

#### site:

site:ndtv.com

Use this operator to search within a specific website only.

#### | verticle bar

blouse | shirt | chemise

This query will search websites that have any one/two/all of the terms

#### .. Two Periods

movies 1950..1970

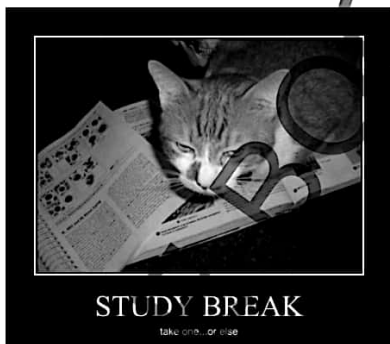
Include two periods when you want to search within two number ranges

### 11. Create Flashcards for Quick Memory Buzz

Quickly test your knowledge of key concepts, definitions, quotes and formulas with flashcards. Sign up to GoConqr now to create your first Flashcard deck like the one below now!

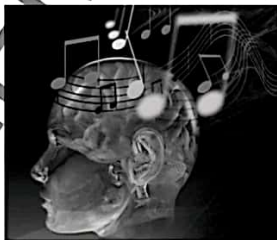
### 12. Take Regular Study Breaks

When your brain is working, you need to take regular study breaks to help your brain absorb more information but also to **keep you motivated and focused** when you are working. Take a short break after 45-50 minutes study as your focus and concentration will become impaired after this period, anything new after 1 hour 30 minutes does not get assimilated.



### 13. Listen to the Correct Type of Music

In our blog post "[Music for Studying: 10 Tips to Pick the Best Study Music](#)" we looked into the area of how the correct types of music can lead to more productive studying by elevating your mood. Have you made your Mozart Spotify playlist yet?



### 14. Make Your Study Space Portable

We may be creatures of habit with favourite seats in the library but information retention actually improves when you vary the places where you study. Check out this [Buzzfeed video](#) for more study hacks for exams:

**15. Practice, Practice, Practice...**

Practicing sample answers to past exam questions can help train your brain to retrieve information. Create realistic, exam-like condition and test your understanding by using our [new Quiz tool](#). Try our general Knowledge Quiz below:

**16. Don't Stay Up All Night Before an Exam**

Make sure to get adequate rest the nights leading up to your exams. When you sleep, **your brain assimilates the information** you have learned when studying so getting a good night's sleep will help you remember those pesky math's formulas you need for your exam!

Don't make this your sleep schedule:

**MY TIME SCHEDULE**

|  |                                              |
|--|----------------------------------------------|
|  | SLEEP                                        |
|  | WAKE UP EXHAUSTED                            |
|  | PLEDGE TO GO TO BED EARLIER TONIGHT          |
|  | SLEEP ALL DAY                                |
|  | FINALLY WAKE UP LATER                        |
|  | WIDE AWAKE!                                  |
|  | IT'S LATE BUT WHO CARES? IT'S TIME TO PARTY! |
|  | IT'S 2 AM I SHOULD PROBABLY GO TO BED.       |

**17. Discover News Ways to Learn**

Trying new study methods can help you find what really works for you. Use technology to your advantage by watching educational TED Talks or downloading useful dictionary apps for example. Read more about the [benefits of learning how to study for exams with technology](#).

**18. Use Scents or Gum to Jog Your Memory**

This may seem a bit random but spraying an unfamiliar scent while you're studying is one of the study methods that can help jog your memory when you spray it again just before an exam. Chewing a strange kind of gum will work the same way.

**19. Study in a Group**

Studying in a group can help you collect new insights to enhance your learning experience. The [GoConqr groups tool](#) is an innovative spin on the traditional study group formula. Our Groups tool helps you share resources, discuss ideas and interact with members of your team or group project. [Sign up here to get started!](#)

**20. Meditate**

Meditation is one of the study methods that can help students stay focused when studying. Not only will meditation help you concentrate when studying but it **will help reduce pre-exam stress** as it improves both mental and physical health.