CHAPTER NO 4: - (DATA & REPETITIONS)

SHORTS QUESTIONS/ANSWERS

1. ***Define data structure?***
   A data structure is a specialized format for organizing, processing, retrieving, and storing data. Common data structures include arrays, linked lists, stacks, queues, trees, and graphs. Each data structure is designed to organize data to suit a specific purpose so that it can be accessed and worked with in appropriate ways.

2. ***Define an array?***
   An array is a data structure consisting of a collection of elements (values or variables), each identified by at least one array index or key. Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

3. ***Differentiate between array initialization & array declaration?***
   Array Declaration: *This is when you specify the type and size of the array. Example in C:* `int arr[10];`
   Array Initialization: *This is when you assign values to the array at the time of declaration or later. Example in C:* `int arr[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};`

4. ***Define accessing array elements?***
   Accessing array elements involves referencing an array element using its index. For example, `arr[0]` *accesses the first element of the array arr.*

5. ***Define looping structure?***
   A looping structure is a control flow statement that allows code to be executed repeatedly based on a condition. Loops are used to perform repeated tasks efficiently.

6. ***How many types of looping structure?***
   There are mainly three types of looping structures:
   - *For Loop*
   - *While Loop*
   - *Do-While Loo*

7. ***Define for loop?***
   A for loop is a control flow statement that allows code to be executed repeatedly based on a condition. It is commonly used when the number of iterations is known before entering the loop.
   Syntax: -
   ```
   for (initialization; condition; increment/decrement) {
       // code to be executed
   }
   ```

8. ***Write down the syntax/structure of for loop?***
   Syntax/Structure: -
   ```
   for (initialization; condition; increment/decrement) {
       // code to be executed
   }
   ```

9. ***Write down the rules of for loop?***
   Initialization*: Sets the starting value for the loop control variable.*
   Condition*: Determines whether the loop will execute.*
   Increment/Decrement*: Updates the loop control variable after each iteration.*

10. *Write a program that display the table of "2"?*

```
#include <stdio.h>
using namespace std;
int main() {
   int i;
   for (i = 1; i <= 10; i++) {
      printf("2 x %d = %d\n", i, 2 * i);
   }
   return 0;
}
```

11. *Define nested loop?*

A nested loop is a loop inside another loop. The inner loop runs completely every time the outer loop runs once.

Syntax: -

```
for (initialization; condition; increment/decrement) {
   for (initialization; condition; increment/decrement) {
      // code to be executed
   }
}
```

12. *Write down the syntax of nested loop?*

Syntax: -

```
for (initialization; condition; increment/decrement) {
   for (initialization; condition; increment/decrement) {
      // code to be executed
   }
}
```

13. *When do we use nested loop?*

Nested loops are used when we need to perform repetitive tasks within repetitive tasks. They are often used for multidimensional array operations.

14. *Define loop and array?*

A loop is a sequence of instructions that is repeated until a certain condition is reached. An array is a collection of elements, typically of the same data type, stored in contiguous memory locations.

15. *Define array name?*

The array name is a reference to the memory address of the first element of the array.

16. *Define array size?*

The array size is the number of elements that an array can hold. It is defined at the time of array declaration.

17. *What is the advantage of initializing an array at the time of declaration?*

Initializing an array at the time of declaration ensures that the array has known values from the start, which can prevent errors related to uninitialized data.

18. *How can you declare an array? Briefly describe the three parts of array declaration?*

int arr[10];

Data type: *The type of elements that will be stored in the array (e.g., int).*

Array name: *The identifier for the array (e.g., arr).*

Size: *The number of elements the array can hold (e.g., 10).*

19. Write a program that display the value from 1-to-10 on the computer screen?

```
#include<stdio.h>
using namespace std;
int main(){
    for(int num=1;num<=10; num+=1){
            printf("%d\n",num);
    }
    return 0;
}
```

20. Write a program that display the tables of "2,3,4,5 & 6"?

```
#include<stdio.h>
using namespace std;
int main(){
    int number;
    printf ("Plz enter the number:");
    scanf("%d",&number);
    printf("Table of %d:\n", number);
    for(int i=0;i<=10;++i){
            printf("%dx%d=%d\n",number,i,number*i);
    }
    printf("The table of any number is here:");
    return 0;
}
```

21. Define counter variable in for loop?

A counter variable in a for loop is used to control the number of times the loop executes. It is typically initialized before the loop starts, tested in the loop's condition, and updated in each iteration

22. How can we declare of an array of type int?

int arr[10];

23. How can we declare of an array of float type?

float arr[10];

24. How can we use a loop to take input from user in an array of size 1-to10?

```
#include <stdio.h>
using namespace std;
int main() {
   int arr[10];
   int i;
   for (i = 0; i < 10; i++) {
      printf("Enter value for element %d: ", i + 1);
      scanf("%d", &arr[i]);
   }
   return 0;
}
```

25. Write down the code to display the element of an array having 100 elements?

```
#include <stdio.h>
using namespace std;
int main() {
   int arr[100];
   int i;

   // Assuming the array is already filled with values
   for (i = 0; i < 100; i++) {
      printf("Element %d: %d\n", i + 1, arr[i]);
   }
   return 0;
}
```

26. What is the advantages of initialization array at the time of declaration?

1. *Ensures that the array has defined values from the start.*
2. *Can save time and reduce errors by not requiring separate initialization code.*
3. *Helps in writing more concise and readable code.*

27. Write down the syntax of while loop?

```
while (condition) {
   // code to be executed
}
```

28. Write down the syntax of do-while loop?

```
do {
   // code to be executed
} while (condition);
```

29. Write a program using FOR loop to print "PUNJAB" 10 times?

```
#include<stdio.h>
using namespace std;
int main(){
    for(int i=0;i<=10; i++){
            printf("PUNJAB");
    }
    return 0;
}
```

30. Write a program using FOR loop to print even numbers from 2 to 20?

```
#include<stdio.h>
using namespace std;
int main(){
    for(int num=2;num<=20; num+=2){
            printf("%d\n",num);
    }
    return 0;
}
```

31. Write a program using FOR loop to print odd numbers from 1 to 19?

```
#include<stdio.h>
using namespace std;
int main(){
    for(int num=1;num<=19; num+=2){
            printf("%d\n",num);
    }
    return 0;
}
```

32. Write a program using FOR loop to print sum of first 10 natural number?

```
#include<stdio.h>
using namespace std;
int main(){
    int sum=0;
    for(int i=1;i<=10;++i){
            sum +=1;
    }
    printf("The sum of the first 10 natural numbers is:%d\n", sum);
    return 0;
}
```

33. ***Write a program using FOR loop to print product of first 10 natural numbers?***

```
#include<stdio.h>
using namespace std;
int main(){
      int product=0;
      for(int i=1;i<=10;++i){
                product *=i;
      }
      printf("The product of the first 10 natural numbers is:%lld\n", product);
      return 0;
}
```

34. ***Write a program using FOR loop to print Alphabets from A to Z?***

```
#include<stdio.h>
using namespace std;
int main(){
      char alphabet;
      for (alphabet = 'A'; alphabet <= 'Z'; ++alphabet) {
        printf("%c ", alphabet);
    }
  printf("\n");
return 0;
}
```

35. ***Write a program using FOR loop to print alphabets from a to z?***

```
#include<stdio.h>
using namespace std;
int main(){
      char alphabet;
      for (alphabet = 'a'; alphabet <= 'z'; ++alphabet) {
        printf("%c ", alphabet);
    }
  printf("\n");
return 0;
}
```

36. ***Write a program using FOR loop which will print numbers in following series 50 , 45 , 40 …. 10?***

```
#include<stdio.h>
using namespace std;
int main(){
      for (int num = 50; num >= 5; num -= 5) {
        printf("%d ", num);
    }
    printf("\n");
    return 0;
}
```

LONG QUESTIONS

1. ***Write a note on array?***

   An array is a data structure consisting of a collection of elements (values or variables), each identified by at least one array index or key. Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

Array Declaration: *This is when you specify the type and size of the array. Example in C:* `int arr[10];`

Array Initialization: *This is when you assign values to the array at the time of declaration or later. Example in C:* `int arr[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};`

Advantages of initialization array at the time of declaration

1. *Ensures that the array has defined values from the start.*
2. *Can save time and reduce errors by not requiring separate initialization code.*
3. *Helps in writing more concise and readable code.*

Array accessing element:

Accessing array elements involves referencing an array element using its index. For example, `arr[0]` *accesses the first element of the array arr.*

Array name

The array name is a reference to the memory address of the first element of the array.

Array size

The array size is the number of elements that an array can hold. It is defined at the time of array declaration.

How can you declare an array? Briefly describe the three parts of array declaration?

`int arr[10];`

Data type: *The type of elements that will be stored in the array (e.g.,* `int`*).*

Array name: *The identifier for the array (e.g.,* `arr`*).*

Size: *The number of elements the array can hold (e.g.,* `10`*).*

2. Write a note on for loop?

A for loop is a control flow statement that allows code to be executed repeatedly based on a condition. It is commonly used when the number of iterations is known before entering the loop.

Syntax: -

```
for (initialization; condition; increment/decrement) {
   // code to be executed
}
```

Rules of for loop?

Initialization*: Sets the starting value for the loop control variable.*

Condition*: Determines whether the loop will execute.*

Increment/Decrement*: Updates the loop control variable after each iteration.*

Example program that display the table of "2"?

```
#include <stdio.h>
using namespace std;
int main() {
   int i;
   for (i = 1; i <= 10; i++) {
      printf("2 x %d = %d\n", i, 2 * i);
   }
   return 0;
}
```

3. Write a note on nested loop?

A nested loop is a loop inside another loop. The inner loop runs completely every time the outer loop runs once.

Syntax of nested loop?

Syntax: -

```
for (initialization; condition; increment/decrement) {
   for (initialization; condition; increment/decrement) {
      // code to be executed
   }
```

```
}
```

Use nested loop?

Nested loops are used when we need to perform repetitive tasks within repetitive tasks. They are often used for multidimensional array operations.

LONG CODING QUESTIONS

1. ***Write a program that takes input the marks obtained in matriculation by 30 students of a class. The program should display the average marks of the class?***

```
#include <stdio.h>
using namespace std;
int main() {
   int marks[30];
   int sum = 0;
   float average;
   printf("Enter the marks obtained by 30 students:\n");
   for (int i = 0; i < 30; ++i) {
      printf("Student %d: ", i + 1);
      scanf("%d", &marks[i]);
      sum += marks[i];
   }
   average = sum / 30.0;
   printf("The average marks of the class is: %.2f\n", average);
   return 0;
}
```

2. ***Write a program that takes two number input and display their Greatest Common Divisor (GCD) using Education method?***

```
#include<stdio.h>
using namespace std;
int main(){
      int m,n;
      printf("Enter the two integer:");
      scanf("%d \n %d", &m, &n);
      while(n>0){
                  int r=m%n;
                  m=n;
                  n=r;
      }
      printf("GCD=%d\n", m);
      return 0;
}
```

3. ***Write a program that declare and initialized an array of 7 elements and tells how many elements in the array are greater than 10.***

```
#include <stdio.h>
using namespace std;
int main() {
   int arr[7] = {3, 12, 9, 18, 5, 20, 15};
   int count = 0;
   for (int i = 0; i < 7; ++i) {
      if (arr[i] > 10) {
         count++;
      }
   }
   printf("Number of elements greater than 10: %d\n", count);
```

```
    return 0;
}
```

4. ***Write a program that takes 10 number as an input in array and display the product of first and last element on console.***

```
#include <stdio.h>
using namespace std;
int main() {
    int arr[10];
    printf("Enter 10 numbers:\n");
    for (int i = 0; i < 10; ++i) {
        printf("Element %d: ", i + 1);
        scanf("%d", &arr[i]);
    }
    int product = arr[0] * arr[9];
    printf("The product of the first and last element is: %d\n", product);
    return 0;
}
```

5. ***Write a program to display the factorial number 1 to 7. (Use nested Loop)***

```
#include <stdio.h>
using namespace std;
int main() {
    for (int i = 1; i <= 7; ++i) {
        unsigned long long factorial = 1;
        for (int j = 1; j <= i; ++j) {
            factorial *= j;
        }
        printf("Factorial of %d is %llu\n", i, factorial);
    }
    return 0;
}
```

ERRORS & OUTPUTS QUESTIONS

Q4 Identify the errors in the following code segments.

a)
```
int a[] = {{2},{3},{4}};
```
Error: Parenthesis is not used in array initialization and single curly braces are used for 1d array.

b)
```
for (int i = 0; i< 10; i++)
        printf("%d\n", i);
```
Error: Semicolon is missing inside the loop.

c)
```
int a[ ] = {1,2,3,4,5};
for (int j = 0; j< 5; j + +)
    printf ("%d ", a(j));
```
Error: Square bracket is used instead of parenthesis in printf statement.

d)
```
float f[] = {1.4, 3.5, 7.3, 5.9};
int size = 4;
for (int n -1; n < size; n--)
    printf ("%f\n", f[n]);
```
Error: Loop condition never be false.

e)
```
int count = 0;
for(int i=4; i < 6; i - - )
for(int j=i; j < 45; j + +)
        {
        count++;
        printf("%count",count)
        }
```
Error: Outer loop never be false. Semicolon is missing at the end of printf statement.

Q5. Write down output of the following code segments.

a)
```
int sum = 0, p;
for (p = 5; p <= 25; p = p + 5)
sum = sum + 5;
printf ("sum is %d ", sum);
```
Output: 25

b)
```
int i;
for (i = 34; i<= 60; i * 2)
printf("*");
```
Output: *

c)
```
for (int i = 50; i<= 50; i++)
        {
        for (j = i; j >= 48, j - -)
            printf ("j =%d \ n", j);
            printf ("i = %d\n", i);
        }
```

Output:
j=50
j=49
j=48
i=50

d)
```
int i, arr[ ] = {2, 3, 4, 5, 6, 7, 8};
        for (i = 0; i <7; i++)
        {
            printf ("%d\n" arr[i] * arr[i]);
            i++;
        }
```

Output:
4
16
36
64

e)
```
int i, j;
        float ar1[ ] = {1.1, 1.2, 1.3};
        float ar2[ ] = {2.1, 2.2,  2.3};
        for (i = 0; i < 3; i ++)
        for (j = i ; j < 3; j++)
        printf("%f\n", ar1[i] * ar2[j] * i * j)
```

Output:
0.000000
0.000000
0.000000
2.640000
5.520000
11.959999

MCQ'S (MULTIPLE CHOICE QUESTIONS)

Exercise

Q1 Multiple Choice Questions

Sr. No	Question	A	B	C	D
1	An array is a ____structure.	Loop	Control	Data	Conditional
2	Array element stored at _____memory locations.	Contiguous	Scattered	Divided	None
3	If the size of an array is 100, the range of indexes will be___	0-99	0-100	1-100	2-102
4	_____structures allows repetition of a set of instruction.	Loop	Conditional	Control	Data
5	___is the unique identifier used to refer to the array.	Data type	Array name	Array size	None
6	Array can be initialized _____declaration.	At the time of	After	Before	Both a & b
7	Using loops inside loop is called ____.	For	While	Do-while	Nested
8	_____part of for loop is executed first.	Condition	Body	Initialization	Increment/Decrement
9	_____make it easier to read and write values in array.	Loops	Conditions	Expression	Functions
10	To initialize the array in a single statement, initialize it_____ declaration	At the time of	After	Before	Both a & b

Key

1	2	3	4	5	6	7	8	9	10
C	A	A	A	B	A	D	C	A	A

MCQs

Sr. No	Question	A	B	C	D
1	_____is a container to store collection of data items in a specific layout.	Software	Hardware	Data structure	Arrays
2	An _____is one of the most commonly used data structures.	Loop	Array	Flowchart	Algorithm
3	An _____is a data structure that can hold multiple values of same data type	Array	Memory	Loop	None
4	How we can declare an array of type int that holds the daily wages of a labors for seven days?	int daily_wage [7]	int daily_wage [7];	int daily_wage (7);	int daily_wage [[7]];
5	An array Index starts with___	-1	0	1	2
6	Which one of the following is the size of int arr[9] assuming that int is of 4 bytes?	9	40	35	none
7	How can we initialize an array in C language?	int arr[2]=(10,20)	int arr(2)={10,20}	int arr[2] = {10, 20};	int arr(2) = (10, 20)
8	Assigning values to an array for the first time, is called_____	Array filling	Array finalization	Array declaration	Array initialization
9	An array can be initialized at the time of its_____	Declaration	Initialization	Finishing	None
10	If we do not initialize an array at the time of declaration, then we need to initialize the array elements_____	one by one	two by two	three by three	together
11	An _____array can hold multiple integer values.	While	Int	Float	nested
12	A _____array can hold multiple real values.	For	Simple	Float	While
13	An important property of array is that it _____inside the computer memory.	stores all the values at consecutive locations	stores all the values at memory locations	stores only one values at consecutive locations	does not stores all the values at consecutive locations
14	Each element of an array has an index that can be used with the _____as array_name[index] to access the data stored at that particular index.	Index	Loop	array location	array name
15	A very important feature of arrays is that we can use _____as array indexes.	Numbers	Variables	Constants	Integer
16	If we need to repeat one or more statements, then we use___	Array	Repetition	Loops	All

17	C language provides _____kind of loop structures.	Three	Four	Five	Six
18	Which on from the following is not a type of loops?	For loop	Do While loop	While loop	Check loop
19	_____is the first part to be executed in a for loop.	Initialization	Declaration	Finalization	None
20	Each run of a loop is called an_____.	Declaration	Repetition	Iteration	Running
21	When we use a loop inside another loop, it is called _____structure.	Else loop	Nested loop	While loop	Do while loop
22	When we want to repeat a pattern for multiple times, then we use_____.	Repetition loop	Do while loop	Else loop	Nested loops
23	We can use _____ inside loop structures.	Loop Structure	Sequence structure	While structure	Nested structure
24	We can use _____inside if structures in any imaginable manners.	While structure	Data structure	loop structures	if structures
25	As _____can be used as array indexes.	Variables	Loop	Data structure	Constants
26	We can use loops to perform different operations on_____.	Information	Arrays	Loops	Data
27	Using_____, we can easily take input in arrays.	For loop	Nested loop	Arrays	Loops
28	_____help us in reading the values from array.	Loops	Array	Compiler	None
29	What is correct syntax of for loop?	for (initialization ; condition; increment /decrement)	for (increment/ decrement; initialization ; condition)	for (initialization , condition, increment / decrement)	none
30	Can for loop contain another for loop?	No	Yes	Compilation Error	Runtime Error

KEY:

1	2	3	4	5	6	7	8	9	10
C	B	A	B	B	B	C	D	A	A
11	12	13	14	15	16	17	18	19	20
B	C	A	D	B	C	A	D	A	C
21	22	23	24	25	26	27	28	29	30
B	D	A	C	A	B	B	A	A	B