//Program development in C, structured programming, using algorithm and flow chart

1. Computer programmers often refer to memory addresses using ____ notation, or base 16.
   a. binary
   b. indirect
   c. mathematical
   d. hexadecimal
   Ans: d.
   It's a number system based on 16
2. After a programmer plans the logic of a program, she will next ____.
   a. understand the problem
   b. test the program
   c. translate the program
   d. code the program
   Ans: d.
   code the program
3. The process of walking through a program's logic on paper pictorially before
   you actually write the program is called ____.
   a. desk-checking
   b. flowcharting
   c. pseudocoding
   d. testing
   Ans: b.
   A flowchart is a graphical tool to construct the logic behind the program.
4. What is the problem with the following statement?
   100=grade
   a. 100 is not a reasonable grade
   b. 100 should be in quotes
   c. data types don't match
   d. value on the left must be a variable name
   Ans: d.
   Most programming languages do not allow a non-variable to sit on the left hand of an assignment statement.

5. What symbol is used to represent output in a flowchart?
   a. square
   b. circle
   c. parallelogram
   d. triangle
   Ans: c.
   Both Input/Output are represented as a parallelogram. Examples: Get X from the user; display X.
6. What is the standard terminal symbol for a flowchart?
   a. circle
   b. lozenge
   c. diamond
   d. square
   Ans: b.
   Start and end symbols, represented as lozenges, ovals or rounded rectangles, usually containing the word "Start" or "End".
7. C Programming was created at______________ by Dennis Ritchie.
   a. Stanford lab
   b. Haward University
   c. AT & T Bell Laboratory
   d. LPU university
   ans: c
8. C Programming was created in Year.
   a 1980
   b 1990
   c 1972
   d1982
   ans c

9. #include is called
 a. Preprocessor directive
 b. Inclusion directive
 c. File inclusion directive
 d. None of the mentioned
ans. a

10 C preprocessors can have compiler specific features.

A. true
B. false
C. Depends on the standard
D. Depends on the platform
Ans A

11
Preprocessor feature that supply line numbers and file names to compiler is called?
A. Selective inclusion
B. macro substitution
C. Concatenation
D. Line control
Ans D

12
A preprocessor is a program.
A. That processes its input data to produce output that is used as input to another program
B. That is nothing but a loader
C. That links various source files
D. All of the mentioned
Ans A

13
#include statement must be written.
A. Before main()
B. Before any scanf/printf
C. After main()
D. It can be written anywhere
Ans B

14
The C-preprocessors are specified with _________symbol.
A. #
B. $
C. " "
D. None of the mentioned.
Ans A

15

The preprocessor provides the ability for ________________.

A. The inclusion of header files
B. The inclusion of macro expansions
C. Conditional compilation and line control.
D. All of the mentioned
Ans D

The preprocessor provides the ability for the inclusion of header files, macro expansions, conditional compilation, and line control.

//The C character set, Identifiers and keywords, Data Types, Constants and variables, Expressions

1. Given the following code, what is printed?

```
   int r;
   r = 14;
   printf("%d %d", r, r++);
```

a.13 14
b.15 14
c.14 14
d.14 15

ANSWER:C

2. Which of the following expressions is not a unary expression?

a. x
b. -x
c. sizeof(x)
d. +x

ANSWER: A

3. Which of the following is not a valid assignment expression?

a. a = b = 0
b. a *= b
c. a = b + 1

d. a + b = c

ANSWER: D

4. Given x is 3, y is 4, and z is 5, what is the value of the expression
--x * (3 + y) / 2 - z++ * y?
a. -9
b. 8
c. -10
d.-13

ANSWER: d

5. The following code contains an error. What is it?

```
 float value;
 scanf(“%f”, value);
```

a. The format specifier is wrong
b. The newline character is missing
c. The width is not specified
d. The parameter is a value and it should be an address

ANSWER: D

6. Which of the following input formatting statements would input the following data?

```
 13 y 14.10
```

a. scanf("%d%c%f", i, c, f);
b. scanf("%i%c%f", i, c, f);
c. scanf("%d%c%f", &i, &c, &f);
d. scanf("%d %c %c", &i, &c, &f);

ANSWER:C

7. Which of the following statements about identifiers is false?
a. Identifiers must start with a letter or underscore
b. Identifiers may not use spaces and hyphens as part of the identifier
c. Identifiers symbolically represent data locations
d. Good programming style uses an underscore as the first character of an identifier

ANSWER: D

8. Which of the following is an invalid variable declaration?
a. int x_f;
b. char status;
c. double value1;
d. int emp-count;

ANSWER: D

9. Which of the following is an invalid initialization?
a. char c = “H”;
b. float value = 20.0;
c. int length = 4000;
d. double pi = 3.141592;

ANSWER: A

10. Which line have error in the following code:

```
L1:int main(){
L2:const int count=50;
L3:count++;
L4:printf("%d",count);
L5:getch();
L6:return 1;}
```

a. L1

b. L2
c: L3
d. L6

Answer: C

11. What would be printed on the output device if the character input is '1' :

```
int main(){
char c;
c=getche();
printf("%c",c);
getch();
return 0;}
```

a. 11
b. 1
c. 65
d. c

Answer: A

//Arithmetic operators, Unary, Relational, Logical, Assignment and conditional operators, Bitwise operators, The C standard library

Q1What is the output of this C code?

```
    #include <stdio.h>

    int main()

    {
      int c = 10 ^ 9;

      printf("%d\n", c);

    }
```

a) 3

b) 8

c) 9

d) 0

ans-a

2.What is the output of this C code?

```
#include <stdio.h>

int main()

{

    int a = 2;

    b= a >> 1;

       printf("%d\n", b);

}
```

a) 0

b) 1

c) 2

d) No Output.

Ans b

3 Suppose that count is an int variable and count = 1. After the statement count++; executes, the value of count is _____.

a. 1

b. 3

c. 2

d. 4

ans=c

4 What will be the output of the
following program code?

```
void main()
{
        int a=3, b=4;
        switch(b%a)
            {
             case 1 :    printf(“A”);
                             break;
             case 3 :    printf(“B”);
                             break;
            default:       printf(“Wrong choice”);
          }
}
```

a) A

b) B

c) Wrong choice

d) Error

ANSWER-A

5 What will be the output of the following program code?

```
void main()
{
        int a=3, b=4;
        switch(a!=b)
            {
             case 1 :    printf(“A”);
                             break;
             case 0 :    printf(“B”);
                             break;
            default:       printf(“Wrong choice”);
          }
}
```

a) A

b) B

c) Wrong choice

d) Error

ANSWER-A

6 What will be the value of d in the following program?

```
    #include <stdio.h>

    int main()

    {

      int a = 10, b = 5, c = 5;

      int d;

      d = b + c == a;

      printf("%d", d);
}
```

a) Syntax error

b) 1

c) 5

d) 10

ans-b

7 What is the output of this C code?

```
    #include <stdio.h>

    int main()

    {

      int i = -3;
```

```
            int k = i % 2;

            printf("%d\n", k);

        }
```

a) Compile time error
b) -1
c) 1
d) Implementation defined
ans-b
8 What is the output of this C code?

```
    #include <stdio.h>
    void main()
    {
        int y = 3;
        int x = 5 % 2 * 3 / 2;
        printf("Value of x is %d", x);
    }
```

a) Value of x is 1
b) Value of x is 2
c) Value of x is 3
d) Compile time error
ans=a
9 What is the output of this C code?

```
    #include <stdio.h>
    int main()
    {
        int x = 1, y = 0, z = 3;
        x > y ? printf("%d", z) :printf("hello");
    }
```

a) 3
b) 1

c) Compile time error

d) Run time error

ans=a

10 What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int x = 1;
    int y = x << 3;
    printf(" %d\n", y);
}
```

a) -2147483648

b) -1

c) Run time error

d) 8

ans=d

11 What is the output of this C code?

```
    #include <stdio.h>

    void main()

    {

        int x = 97;

        int y = sizeof(x);

        printf("y is %d", y);

    }
```

a) y is 97

b) y is 4

c) y is 8

d) Run time error

12 What is the output of this C code?

```
#include <stdio.h>

int main()

{

    int x = 2, y = 1;

    x *= x + y;

    printf("%d\n", x);

    return 0;

}
```

a) 5

b) 6

c) Undefined behaviour

d) Compile time error

ans-b

//Memory Based

13. Which of the following is not a logical operator?

a.if

b.and

c. not

d. or

Ans: A

14 **Which of the following statements about the logical AND operator is true?**

a. The and operator is a unary operator.

b. The result is true only when both operands are true.

c. The and operator is a multiway selection.

d. The and operator selects between two values.

Ans b

15 **Which of the following statements about bitwise operators is true?**

a. Bitwise AND and OR operators compare two bits in two operands to determine the result.

b. The bitwise operators are all binary operators.

c. Bitwise operators can only be applied to integer values.

d. Bitwise operators return a Boolean, true or false.

ANSWER: A

16 The bitwise ____ operator that sets a bit to 1 only if both operands are 1.

a. and

b. inclusive or

c. exclusive or

d. complement

ANSWER: A

17 The bitwise ____ operator reverses bits.

a. and

b. inclusive or

c. exclusive or

d. complement

ans d

18Which of the following is not a relational operator?

a. =

b. >=

c. <

d. >

ANSWER: A

19Which of the following is not an arithmetic operation?

A. a *= 20;

B. a /= 30;

C.a %= 40;

D. a != 50;

ANS-D

20The precedence of arithmetic operators is (from highest to lowest)?

A. %, *, /, +, -

B. %, +, /, *, -

C. +, -, %, *, /

D. %, +, -, *, /

ANS-A

21Logical expressions produce _____________ type results.

a. explicit

b. garbage

c. bool

d. static

Ans-c

// Error finding Based

22 Which of the following data type will throw an error on modulus operation(%)?

A. char

B. short

C. float

D. int

ANS-C

23 Find the error in the program

```
void main()
   {
      float a = 10.6;
      int x = (a % 10);
      printf("%d\n", x);
   }
```

A Datatype of x should be float
B Datatype of a should be int
C both a and b
D no error
Ans-b
24 Find the error in the program

```
#include <stdio.h>
    int main()

    {
        int a |=10;


        printf("%d\n", a);
    }
```

AProgram is using invalid assignment operator

B Expression a|=10 can be written as a&=10

C both a and b

D no error

Ans-a

//Syntax Based

25What is the correct syntax of using sizeof() operator for finding integer variable size
a printf(“size of int is %f”,size_of(int));
b printf(“size of int is %d”,sizeof(int));
c printf(“size of int is %c”,sizeof(int));
d printf(“size of int is %f”,sizeof());
ans-b

//If, If else, Switch case statements, Nested if

Q:1. The output of the code below is

```
#include <stdio.h>
void main()
{
    int x = 5;
    if (x < 1)
        printf("hello");
    if (x == 5)
        printf("hi");
    else
        printf("no");
}
```

a) hi

b) hello

c) no

d) None of the mentioned

Ans: a

Q:2 The output of the code below is

```
#include <stdio.h>
int x;
void main()
{
    if (x)
        printf("hi");
    else
        printf("how are u");
}
```

a) hi

b) how are you

c) Compile time error

d) None of the mentioned

Answer:b

Q:3 Comment on the following code below

```
#include <stdio.h>
void main()
{
    int x = 5;
    if (true);
        printf("hello");
}
```

a) It will display hello
b) It will throw an error
c) Nothing will be displayed
d) Compiler dependent

Answer:b

Q:4 The output of the code below is

```
#include <stdio.h>
void main()
{
    int x = 0;
    if (x == 0)
        printf("hi");
    else
        printf("how are u");
        printf("hello");
 }
```

a) hi
b) how are you
c) hello
d) hihello

Answer:d

Q:5 The output of the code below is

```
#include <stdio.h>
void main()
{
    int x = 5;
    if (x < 1);
        printf("Hello");

}
```

a) Nothing
b) Run time error
c) Hello

d) Varies

Answer:c

Q:6 The output of the code below is(when 1 is entered)

```
#include <stdio.h>
void main()
{
    double ch;
    printf("enter a value btw 1 to 2:");
    scanf("%lf", &ch);
    switch (ch)
    {
    case 1:
         printf("1");
         break;
     case 2:
         printf("2");
         break;
     }
  }
```

a) Compile time error(switch quantiy not an integer)

b) 1

c) 2

d) Varies

Answer: a

Q:7 The output of the code below is(When 1 is entered)

```
#include <stdio.h>
void main()
{
    char *ch;
    printf("enter a value btw 1 to 3:");
    scanf("%s", ch);
    switch (ch)
    {
```

```
    case "1":
        printf("1");
        break;
     case "2":
        printf("2");
        break;
     }
}
```

a) 1

b) 2

c) Compile time error

d) No Compile time error

Answer:c

Q 8 When 1 is entered, The output of the code below is?

```
#include <stdio.h>
void main()
{
    int ch;
    printf("enter a value btw 1 to 2:");
    scanf("%d", &ch);
    switch (ch)
    {
    case 1:
        printf("1\n");
     default:
        printf("2\n");
     }
 }
```

a) 1

b) 2

c) 1 2

d) Run time error

Answer: c

Q 9 When 2 is entered, The output of the code below is?

```
#include <stdio.h>
void main()
{
    int ch;
    printf("enter a value btw 1 to 2:");
    scanf("%d", &ch);
    switch (ch)
    {
    case 1:
         printf("1\n");
         break;
         printf("Hi");
     default:
         printf("2\n");
     }
 }
```

a) 1

b) Hi 2

c) Run time error

d) 2

Answer:d

Q 10: What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 1;
    if (x > 0)
        printf("inside if\n");
    else if (x > 0)
        printf("inside elseif\n");
}
```

a) inside if

b) inside elseif

c) inside if
   inside elseif

d) Compile time error

Answer:a

Q 11: What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 0;
    if (x++)
        printf("true\n");
    else if (x == 1)
        printf("false\n");
}
```

a) true

b) false

c) Compile time error

d) Undefined behaviour

Answer:b

Q:12 What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 0;
    if (x == 1)
        if (x == 0)
            printf("inside if\n");
        else
            printf("inside else if\n");
     else
         printf("inside else\n");
 }
```

a) inside if

b) inside else if

c) inside else

d) Compile time error

Answer:c

Q: 13

What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 0;
    if (x == 0)
        printf("true, ");
    else if (x = 10)
        printf("false, ");
    printf("%d\n", x);
}
```

a) false, 0

b) true, 0

c) true, 10

d) Compile time error

Answer:b

Q: 14

What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 0;
    if (x == 1)
        if (x >= 0)
            printf("true\n");
        else
            printf("false\n");
}
```

a) true

b) false

c) Depends on the compiler

d) No print statement

Answer:d

Q:15 What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int a = 1;
    if (a--)
        printf("True");
        if (a++)
            printf("False");
}
```

a) True

b) False

c) True False

d) No Output

Answer:a

Q:16

Comment on the output of this C code?

```
#include <stdio.h>
int main()
{
    int a = 1;
    if (a)
        printf("All is Well ");
        printf("I am Well\n");
    else
        printf("I am not a River\n");
}
```

a) Output will be All is Well I am Well

b) Output will be I am Well I am not a River

c) Output will be I am Well

d) Compile time errors during compilation(else without previous if)

Answer:d

Q: 17

What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int a = 1, b = 1;
    switch (a)
    {
    case a*b:
        printf("yes ");
    case a-b:
          printf("no\n");
          break;
      }
  }
```

a) yes
b) no
c) Compile time error
d) yes no

Answer:c

Q:18 What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int x = 97;
    switch (x)
    {
    case 'a':
        printf("yes ");
        break;
      case 97:
          printf("no\n");
          break;
      }
  }
```

a) yes
b) yes no
c) Duplicate case value error

d) Character case value error

Answer:c

Q:19

What is the output of this C code?

```
#include <stdio.h>
int main()
{
    float f = 1;
    switch (f)
    {
    case 1.0:
        printf("yes\n");
        break;
    default:
        printf("default\n");
    }
}
```

a) yes

b) yes default

c) Undefined behaviour

d) Compile time error

Answer:d

Q:20 What is the output of this C code?

```
#include <stdio.h>
const int a = 1,  b = 2;
int main()
{
    int x = 1;
    switch (x)
    {
    case a:
        printf("yes ");
    case b:
        printf("no\n");
        break;
        }
    }
```

a) yes no

b) yes

c) no

d) Compile time error

Answer:d

//While, For, Do-while loops, Nested loops(By Sanjeev Sir)

1. Fill in the blanks:

   ```
   ______(<expression>)
   {
   //code block
   }
   ```

   a. for
   b. while
   c. do-while
   d. while-do

Ans: b

2. Fill in the blanks:

   ```
   ___
   {
   //code block
   }______(<expression>);
   ```

   a. for, for
   b. while, while
   c. do, while
   d. while, do

Ans: c

3. Fill in the blanks:

```
_____(<expression1>;<expression2>;<expression3>)
{
//code block
}
```

   a. for

b. while

c. do-while

d. while-do

Ans: a

4. Find the correct order:

```
for(_____;______;_____)
{
//code block
}
```

a. Condition, initialization, updating

b. Updating , condition, initialization

c. Initialization, condition, updating

d. Condition, updating, initialization

Ans: c

5. The keyword *break* is used for

a. Skipping the instructions in the next lines and starting the next iteration

b. Terminating the internal loop

c. Jumping to the predefined label

d. Returning the control of the program

Ans: b

6. The keyword *continue* is used for

a. Skipping the instructions in the next lines and starting the next iteration

b. Terminating the internal loop

c. Jumping to the predefined label

d. Returning the control of the program

Ans: a

7. The keyword *goto* is used for

a. Skipping the instructions in the next lines and starting the next iteration

b. Terminating the internal loop

c. Jumping to the predefined label

d. Returning the control of the program

Ans: c

8. The keyword *return* is used for
   a. Skipping the instructions in the next lines and starting the next iteration
   b. Terminating the internal loop
   c. Jumping to the predefined label
   d. Returning the control of the program

Ans: d

9. The while loop is a looping structure
   a. Where the condition occurs first
   b. Where the updating occurs first
   c. Where the initialization occurs at first
   d. Where all the three expressions are in lined

Ans: c

10. The for loop is representation of while loop
    a. Where the condition occurs first
    b. Where the updating occurs first
    c. Where the initialization occurs at last
    d. Where all the three expressions are in lined

Ans: d

11. The do-while loop is different than while loop
    a. Where the condition occurs first
    b. Where the updating occurs first
    c. Where the code will execute without checking the condition at first
    d. Where all the three expressions are in lined

Ans: c

12. How many time *hello* will be displayed for this code

```
#include<stdio.h>
int main()
{
int a=1;
while(a<10)
{
printf(“hello”);
a++;
}
}
```

a. 10
b. 9
c. 8
d. 11

Ans: b

13.How many times *hello* will be displayed for this code

```
#include<stdio.h>
int main()
{
int a=10;
  while(a>0)
  {
printf(“hello”);
    a--;
  }
}
```

a. 10
b. 11
c. 9
d. 8

Ans: a

14.How many times *hello* will be displayed for this code

```
#include<stdio.h>
int main()
{
int a;
for(a=0;a<10;a++)
  {
printf(“hello”);
  }
}
```

a. 11
b. 10
c. 9
d. 8

Ans: b

15.How many times *hello* will be displayed for this code

```
#include<stdio.h>
```

```
int main()
{
int a=10;
do
{
printf(“hello”);
a--;
} while(a>0);
}
```

a. 10
b. 11
c. 9
d. 8

Ans: a

16.Guess the output of the code

```
#include <stdio.h>
int main ()
{
  /* local variable definition */
  int i, j;
  for(i =1; i<=5; i++)
  {
    for(j = 1; j <=i; j++)
    {
       printf("*");
    }
 }
  return 0;
}
```

```
a. *            b.         *  c. *****    d. *****
   **                     **        ****       ****
   ***                   ***         ***       ***
   ****                 ****          **       **
   *****               *****           *       *
```

Ans: a

17.What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int k = 0;
    for (k)
        printf("Hello");
}
```

a) Compile time error
b) hello
c) Nothing
d) Varies

Ans:a

18.Guess the output of the code

```
#include <stdio.h>
int main ()
{
   /* local variable definition */
   int i, j;
   for(i =5; i>=1; i--)
   {
      for(j = 1; j <=i; j++)
      {
         printf("*");
      }
 }
   return 0;
}
```

a.

```
*
**
***
****
*****
```

b.

```
  *
 **
***
```

c.

```
 ****
*****
*****
 ****
  ***
   **
    *
```

d.

```
*****
****
***
**
*
```

Ans: d

19.Guess the output of the code

```
#include<stdio.h>
#include<conio.h>
int main(){
int n,i,j,k;
printf("Enter the no of lines: ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
        for(j=0;j<n-i-1;j++)
        {
                printf(" ");
        }
        for(k=0;k<=i;k++)
        {
                printf("*");
        }
printf("\n");
}
getch();
return 0;
}
```

a.
```
*
**
***
****
*****
```

b.
```
    *
   **
  ***
 ****
*****
```

c.
```
*****
 ****
  ***
   **
    *
```

d.
```
*****
****
***
```

```
    **
    *
```

Ans: b

//Break and continue statements, Goto, Return, Type conversion and type modifiers, Designing structured programs in C, Top down design and stepwise refinement, Formatted and unformatted Input/Output, functions like printf(), Scanf(), Puts(), Gets() etc.

Q1. The set of functions that are used for formatted input and output

a. getchar() and putchar()
b. getch() and putch()
c. scanf() and printf()
d. all of the above

Ans: c

Q2. The output could be displayed as

a. left justified
b. right justified
c. both of the above
d. none of the above

Ans: c

Q3. Each format/conversion specification contains

a. % symbol
b. Format/conversion specifier
c. Any one of the above
d. Both of the above

Ans: d

Q4. Various format for floating point input are

a. 314.1
b. 3.141e+03
c. 31.41e-03
d. All of the above

Ans: d

Q5. In general, the structure of printf() and scanf() is

a. (<arguments> , <format control string>)

b. (<format control string> , <argument list>)
c. (<arguments>)
d. None of the above

Ans: b

Q6. To print out a and b given below, which of the following printf() statement will you use?

```
#include<stdio.h>
float a=3.14;
double b=3.14;
```

a. printf("%f %lf", a, b);
b. printf("%Lf %f", a, b);
c. printf("%Lf %Lf", a, b);
d. printf("%f %Lf", a, b);

Ans: a

Q7. What will be the output of the program ?

```
#include<stdio.h>
int main()
{
   float a=3.15529;
   printf("%.1f\n", a);
   return 0;
}
```

a. 3.00
b. 3.15
c. 3.2
d. 3

Ans: c

Q9. Which among the following is odd one out?

a. printf
b. putch
c. putchar
d. scanf

Ans: d

Q10. Which among the following is odd one out?

a. scanf
b. getch
c. printf
d. getchar

Ans: c

Q11. What is the return value of putchar()?

a. The character written on the screen
b. The next input character each time it is called
c. Nothing
d. Both a & b

Ans: a

Q12. What is the return value of getchar()?

a. The character written on the screen
b. The next input character each time it is called
c. Nothing
d. Both a & b

Ans: b

Q13. Guess the output of the following code:-

```
#include<stdio.h>
int main()
{
int x=1;
while(1)    /*infinite loop*/
{
  if(x==5)
    break;
  printf("%d",x++);
}
return 0;
}
```

a. 1 2 3 4
b. 2 3 4 5
c. 5

d. No output

Ans: a

Q14. Guess the output of the following code:-

```
#include<stdio.h>
int main()
{
int x=1;
while(1)    /*infinite loop*/
{
  if(x>5)
     break;
  printf("%d",x++);
}
return 0;
}
```

a. 1 2 3 4 5
b. 2 3 4 5 6
c. Infinte loop
d. No output

Ans: a

Q15. Guess the output of the following code:-

```
#include<stdio.h>
int main()
{
int x;
for(x=1;x<=10;x++)
{
   if(x%2==1)  /*checking for odd number */
      continue;
   printf("\t%d",x);
}
return 0;
}
```

a. 1 3 5 7 9
b. 2 4 6 8 10

c. Infinite loop
d. 1 2 3 4 5 6 7 8 9 10

Ans: b

Q16. What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int a = 0,i;
    for (i = 0;i < 5; i++)
     a++;
    printf("%d",a);
}
```

a. 2
b. 3
c. 4
d. 5

Ans: d

Q17. What is the output of this C code?

```
#include <stdio.h>
int main()
{
    int a = 0, i ;
    for (i = 0;i < 5; i++)
    {
        a++;
        if (i == 3)
            break;
    }
      printf(“%d”,a);
}
```

a. 1
b. 2
c. 3
d. 4

Ans: d

Q18. The keyword ‘break’ cannot be simply used within

a. do while
b. if-else
c. for
d. while

Ans: b

Q19. Which keyword is used to come out of a loop only for that iteration?

a. break
b. continue
c. return
d. none of the above

Ans: b

Q20. What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int i , j ;
    for (i = 0;i < 5; i++)
    {
        for (j = 0;j < 4; j++)
        {
            if (i > 1)
                break;
        }
        printf("Hi \n");
    }
}
```

a. Hi is printed 5 times
b. Hi is printed 9 times
c. Hi is printed 7 times
d. Hi is printed 4 times

Ans: a

Q21. What is the output of this C code?

```
#include <stdio.h>
```

```
void main()
{
    int i ;
    int j ;
    for (i = 0;i < 5; i++)
    {
        for (j = 0;j < 4; j++)
        {
            if (i > 1)
                continue;
                printf("Hi \n");
        }
    }
}
```

a. Hi is printed 5 times
b. Hi is printed 8 times
c. Hi is printed 7 times
d. Hi is printed 4 times

Ans: b

Q22. What is the output of this C code?

```
#include <stdio.h>
void main()
{
    int i ;
    for (i = 0;i < 5; i++)
        if (i < 4)
        {
            printf("Hello");
            break;
        }
}
```

a. Hello is printed 5 times
b. Hello is printed 4 times
c. Hello
d. Hello is printed 3 times

Ans: c

//Function prototypes, Function definition, Function call including passing arguments by value and passing arguments, by reference, Math library functions, Recursive functions

Q1. What will be output when you will execute following c code?

```
#include <stdio.h>
   int main()
   {
      void abc();
      printf("1 ");
      abc();
   }
   void abc()
   {
      printf("2 ");
   }
```

Choose the correct answer:

A. 1 2
B. 2 1
C. 1 1
D. 2 2

Ans: A

Q2. What will be output when you will execute following c code?

```
#include <stdio.h>
   int main()
   {
      void f1();
      void f()
      {
         f1();
      }
      f();
   }
   void f1()
   {
      printf("2 ");
   }
```

Choose the correct answer:

A. 1
B. 2
C. 0
D. None of the above

Ans: B

Q3. What will be output when you will execute following c code?

```
#include <stdio.h>
#include <math.h>
void main()
{
    printf("fmod(8,5) = %f\n", fmod(8,5));
}
```

A. fmod(8,5) = 2.000000
B. fmod(8,5) = 3.000000
C. fmod(8,5) = 0.000000
D. fmod(8,5) = 1.000000

Ans:- B

Q4. What will be output when you will execute following c code?

```
#include <stdio.h>
int sum(int);
int main()
 {
  printf("%d",sum(2));
  }
 int sum(int n)
 {
         if (n==0)
         return n;
         else
         return n + sum(n-1);
}
```

Choose the correct answer:

A. 3
B. 2
C. 0
D. 4

Ans: A

Q5. What will be output when you will execute following c code?

```
#include <stdio.h>
#include <math.h>
void main()
{
    printf("ceil(2.12) = %f\n", ceil(2.12));
}
```

E. ceil(2.12) =2.120000
F. ceil(2.12) =3.000000
G. ceil(2.12) =2.500000
H. ceil(2.12) =2.000000

Ans:- B

Q6. What will be output when you will execute following c code?

```
#include <stdio.h>
   int main()
   {
      void func();
      printf("work");
      func();
   }
   void func()
   {
      printf("hard");
   }
```

Choose the correct answer:

A. hardwork
B. workhard
C. workwork
D. hardhard

Ans: B

//Syntax Based

Q7. Fill in the blank

```
#include <stdio.h>
```

```
int sum(int n);
int main()
{
   int number, result;
   printf("Enter a positive integer: ");
   scanf("%d", &number);
   result = ________ (number);      //function call
   printf("sum=%d", result);
}
int sum(float num)
{
   if (num!=0)
      return num + sum(num-1);
   else
      return num;
}
```

A. int
B. result
C. sum
D. None of the above

Ans: C

Q8. Fill in the blank

```
#include <stdio.h>
int factorial(________)
 {
      if(i <= 1)
      return 1;
      else
      return i * factorial(i - 1);
  }
 int  main()
{
     int i;
     scanf("%d",&i);
     printf("Factorial of %d is %d\n", i, factorial(i));
     return 0;
```

}

A. int i, int j
B. int i
C. int,int
D. float i

Ans: B

//Memory Based

Q9. What is the use of functions?

A. Helps to avoid repeating a set of statements many times
B. Helps to avoid repeated programming across programs.
C. Makes the debugging task easier.
D. All of the above

Ans: D

Q10. Which of the following is not a math library function?

A. Sqrt()
B. Pow()
C. Sum()
D. Floor()

Ans:- C

Q11. Recursion is

A. When a function is called within the same function
B. calling different function
C. defining a function
D. None

Ans:- A

Q12. Which header file is used for math library functions?

A. ctype.h
B. stdio.h
C. math.h
D. None

Ans:- C

Q13. Which of the following returns largest whole number not greater than x?

A. Ceil()

B. Floor()
C. Fmod()
D. Fabs()

Ans:- B

Q14. Which of the following returns the smallest whole number that is not less than x?

A. Ceil()
B. Floor()
C. Fmod()
D. Fabs()

Ans:- A

Q15. Which of the following function is used to calculate square root of a number?

A. squart()
B. sqrt()
C. squaroot()
D. None

Ans:- B

Q16. Calling function is present in

A. main()
B. definition of the function
C. declaration of the function
D. None

Ans:- A

// Error finding Based

Q17. Point out the error in the program

```
f(int a, int b)
{
   int a;
   a = 20;
   return a;
}
```

A. Missing parenthesis in return statement

B. The function should be defined as int f(int a, int b)

C. Re declaration of a

D. None of the above

Ans:- C

Q18. Point out the error in the program

```
f(int x);
{
   x = 20;
   return x*x;
}
```

A. Missing parenthesis in return statement

B. x is not declared within block of function

C. Semicolon after function definition

D. None of the above

Ans:- C

Q19. Point out the error in the program

```
#include <stdio.h>
int sum(int n)
int main()
{
   int number, result;
   printf("Enter a positive integer: ");
   scanf("%d", &number);
   result = sum(number);
   printf("sum=%d", result);
}
int sum(float num)
{
   if (num!=0)
      return num + sum(num-1);
   else
      return num;
}
```

A. conflicting data types for function 'sum'

B. missing semicolon in function declaration

C. both A and B

D. None of the above

Ans:- C

Q20. Point out the error in the program

```
#include <stdio.h>
#include <math.h>
int main ()
{
        float  pi = 3.1415926;
        printf("pow(2,3) = %f\n, pow(2,3)”);
}
```

A. Double quotes missing in printf

B. Conversion specifier is not correct

C. Both A and B

D. None of the above

Ans:- A

//Scope rules (local and global scope), Storage classes in C namely auto, Extern, Register, Static storage classes, Declaring and initializing arrays in C

Ques: What will be printed after execution of the following code?

```
void main()
{
     int arr[10] = {1,2,3,4,5};
     printf("%d", arr[5]);
}
```

(a) Garbage Value   (b) **0**

(C)   6   (d) None of these

Ans: (b)

Ques: What will be the output of the program ?

```
#include<stdio.h>
int main()
{
        int a[5] = {51, 1, 5, 20, 25};
        int x, y, z;
        x = ++a[1];
        y = a[1]++;
        z = a[x++];
        printf("%d, %d, %d", x, y, z);
        return 0;
}
```

(a) 2, 3, 20 (b) 2, 1, 5
(c) 1, 2, 5 (d) 3, 2, 5
Ans: (d)

Ques: What will the output for this program?

```
#include <stdio.h>
int fun()
{
  static int num = 16;
  return num--;
}
 int main()
{
  for(fun(); fun(); fun())
    printf("%d ", fun());
  return 0;
}
```

(a) Infinite loop (b) 13 10 7 4 1
(c) 14 11 8 5 2 (d) 15 12 8 5 2
Ans : (c)

Ques: What will be the output of the program ?

```
#include"stdio.h"
void main()
{
float arr[] = {12.4, 2.3, 4.5, 6.7};
printf("%d", sizeof(arr)/sizeof(arr[0]));
}
```

(a) 5 (b)4
(c) 6 (d) 7
Ans: 4

Ques: What is the output of the following program snippet?

```
#include<stdio.h>
```

```
void main()
{
        static num = 4;
        printf("%d ",--num);
        if(num)
                main();
}
```

(a) 4 3 2 1 0 (b) 3 2 1 0

(c) Infinite Times 4 (d) Compile or Run Time Error

Ans: (b)

Ques: What will the output this code?

```
#include<stdio.h>
void fn()
{
        static int i=5;
        printf("%d\t",++i);
}
int main()
{
        fn();
        fn();
        return 0;
}
```

(a) 5 5 (b) 6 6

(c) 6 7 (d) 7 7

Ans: (c)

Ques: What will be the output?

```
int main()
{
  int x = 032;
  printf("%d", x);
  return 0;
}
```

(a) 32 (b) 0

(c) 26 (d) 5

Ans: (c)

Ques: What will be the output for following code snippet?

```
 #include <stdio.h>
int main()
{
  int x = 1, y = 2, z = 3;
  printf(" x = %d, y = %d, z = %d \n", x, y, z);
  {
       int x = 10;
       float y = 20;
       printf(" x = %d, y = %f, z = %d \n", x, y, z);
       {
             int z = 100;
             printf(" x = %d, y = %f, z = %d \n", x, y, z);
       }
  }
  return 0;
}
```

(a) x = 1, y = 2, z = 3

x = 10, y = 20.000000, z = 3

x = 1, y = 2, z = 100

(b) Compiler Error

(c) x = 1, y = 2, z = 3

x = 10, y = 20.000000, z = 3

x = 10, y = 20.000000, z = 100

(d) x = 1, y = 2, z = 3

x = 1, y = 2, z = 3

x = 1, y = 2, z = 3

Ans: (c)

// SYNTAX BASED

Ques: An array elements are always stored in ________ memory locations.

(a) Sequential (b)Random

(c) Sequential and Random (d) None of the above

Ans: (a)

Ques: What is the maximum number of dimensions an array in C may have?

(a) 2 (b) 8

(c) 20 (d) Theoretically no limit. The only practical limits are memory size and compilers

Ques : Regarding the scope of the variables identify the incorrect statement:

(a) Automatic variables are automatically initialized to 0

(b)Static variables are automatically initialized to 0

(c) The address of a register variable is not accessible

(d)Static variables cannot be initialized with any expression

Ans: a

Ques: Due to variable scope in c

(a) Variables created in a function cannot be used another function

(b) Variables created in a function can be used in another function

(c) Variables created in a function can only be used in the main function

(d) None of above

Answer: (a)

Ques: Which is not a storage class?

(a)Auto (b) Struct

(c) Extern (d) Static

Ans: (b)

Ques: Which of s, t and u are available to a function present in another file?

```
extern int s;
int t;
static int u;
main()
{

}
```

(a) Only s (b) S & u

(c) S, t, u (d) None

Ans: (a)

Ques: Size of the array need not be specified, when

(a) Initialization is a part of definition

(b)It is a declaratrion

(c)It is a formal parameter

(d) All of these

Ans: (d)

Ques: Variables inside parenthesis of functions declarations have _____ level access.

(a) Local (b) Global

(c) Module (d) Universal

Ans: (a)

Ques: Consider the following C program, which variable has the longest scope?

```
int a;
int main()
{
   int b;
   // ..
   // ..
}
int c;
```

(A) a (B) b

(C) c (D) All have same scope

Ans: (a)

// Error finding

Ques: Find the error in below code snippet:

```
int main()
{
  {
     int x = 10;
  }
  {
     printf("%d", x);
  }
  return 0;
}
```

(a) 'X' undeclared

(b) No error

(c) Should use void main

(d) Return 0 is optional

// Memory Based

Ques: Set of consecutive memory locations is called as ________.

(a) Function (b) Loop

(c) Array (d) Pointer

Ans: (c)

Ques: Array can be considered as set of elements stored in consecutive memory locations but having __________.

(a) Different Data Type (b) None of these

(c) Same Data Type (d) Same Scope

Ans: (c)

Ques: Array is an example of _______ type memory allocation.

(a) Compile Time

(b) Run Time

Ans: (a)

Ques: Array is ______ data type in C Programming language.

(a) Derived Data Type (b) None of these

(c) Custom Data Type (d) Primitive Data Type

Ans: (a)

Ques: If you pass an array as an argument to a function, what actually gets passed?

(a) First element of the array (b) Value of elements in array

(c)Address of the last element of array (d) Base address of the array

Ans: (d)

Ques: Higher priority is given to ________ variable than _______ variable.

(a) Local , Global

(b) Global , Local

Ans: (a)

// Searching including linear and binary search methods, Sorting of array using bubble sort

Q1. The following C code contains the logic of which searching technique

```
for ( i = 0; i < size; i++ ) {
   if ( key == list[i] )
     break;
 }
```

```
  if ( i < size ) {
    found = true;
    rec = &list[i];
  }
  return found;
}
```

Choose all that apply:

E. Binary
F. Linear
G. Hashing
H. None of the Above

Ans:- B

Q2. The following C code contains the logic of which searching technique

```
int MIDDLE,
 LEFT = 0,
 RIGHT = array_size-1;
 while (( LEFT<= RIGHT ) && (!found))
 { MIDDLE = (LEFT + RIGHT )/ 2; // Get middle index
if ( array[MIDDLE] == search_key)
{ index = MIDDLE;
 found = true;
}
 else if (array[MIDDLE] > search_key)
 RIGHT= MIDDLE– 1; // search is focused
 // on the left side of list
 else
 LEFT= MIDDLE+ 1 // search is focused
 // on the right side of the list
 } //end while
```

Choose all that apply:

A. Binary
B. Linear
C. Hashing
D. None of the Above

Ans:- A

Q3. Fill in the blank

```
void sequential_search(int array[], int size, int n)
{
   int i;
    ____( i = 0; i < size; i++)
    {
    if (array[i] == n)
          {
             printf("%d found at location %d.\n", n, i+1);
             break;
          }}
```

E. do while

F. while

G. for

H. void search

Ans: C

Q4. Binary search algorithm cannot be applied to ...

A. sorted linked list

B. sorted binary trees

C. sorted linear array

D. pointer array

Ans: A

Q5. Linear search algorithm is also known as

A. Fast Search

B. Quick Search

C. Sorted Search

D. Sequential search

Ans: D

Q6. The operation of processing each element in the list is known as

A. Sorting

B. Merging

C. Inserting

D. Traversal

Ans: D

Q7. Finding the location of the element with a given value is:

A. Traversal
B. Search
C. Sort
D. None of above

Ans: B

Q8. The elements of an array are stored successively in memory cells because

A. by this way computer can keep track only the address of the first element and the addresses of other elements can be calculated
B. the architecture of computer memory does not allow arrays to store other than serially
C. both of above
D. none of above

Ans: A

Q9. Which searching technique can be performed recursively?

A. Linear
B. Binary
C. Both
D. None

Ans C

Q10. Which searching technique can be performed iteratively?

A. Linear
B. Binary
C. Both
D. None

Ans C

Q11. Binary search is more suitable for:

A. Array
B. Stack
C. Linked list
D. Any of the above

Ans A

Q12. Pick the odd one out:

A. Linear Search
B. Binary Search
C. Sequential Search
D. Quick Search
Ans D

Q13. Binary search is also known as
A. Sequential Search
B. Quick Search
C. Logarithmic search
D. None of the above
Ans C

Q14. In a bubble sort structure, there is/are?
A. Single for loop
B. Three for loops, all separate
C. While loop
D. Two for loops, one nested in the other
Ans D

Q15. What is an external sorting algorithm?
A. Algorithm that uses tape or disk during the sort
B. Algorithm that uses main memory during the sort
C. Algorithm that involves swapping
D. Algorithm that are considered 'in place'
Ans A

Q16. What is an external sorting algorithm?
A. Algorithm that uses tape or disk during the sort
B. Algorithm that uses main memory during the sort
B. Algorithm that involves swapping
C. Algorithm that are considered 'in place'
Ans B

Q17. The following C code contains the logic of which sorting technique

```
 for(int j=arr.length-1; j>=0; j--)
{
        for(int k=0; k<j; k++)
        {
                if(arr[k] > arr[k+1])
                {
                        int temp = arr[k];
                        arr[k] = arr[k+1];
                        arr[k+1] = temp;
                }
        }
}
```

Choose all that apply:

A. Selection
B. Quick
C. Bubble
D. Radix

Ans:- C

Q18. Why is bubble sort method often presented when learning sorting method.

A. It is fastest
B. It is most efficient
C. It uses fewer loops than others
D. It is easy to understand.

Ans C

Q19. How many passes required in bubble sort to sort any array containing five elements.

A. 4
B. 5
C. 3
D. 1

Ans : A

Q20.Bubble sort is also known as

A. Sinking Sort
B. Comparison Sort
C. Both A and B
D. None of the above
Ans C