MODULE 3

1. What is the output of the following code?

```
main()
{
        int a=1, b=10;
        swap(a,b);
        printf("\n%d%d", a,b);
}
swap(int x, int y)
{
        int temp;
        temp=x;
        x=y;
        y=temp;
}
```

   A. 1 1
   B. 1 10
   C. 10 1
   D. None of these

2. The default parameter passing mechanism is
   A. Call by value
   B. Call by reference
   C. Call by value result
   D. None of the above

3. Use of functions
   A. helps to avoid repeating a set of statements many times
   B. enhances the logical clarity of the program
   C. helps to avoid repeated programming across programs
   D. all of the above

4. What does the following function print?

```
func(int i)
{
        if(i%2) return 0;
        else return 1;
}
main()
{
        int i=3;
        i=func(i);
        i=func(i);
        printf("%d", i);
}
```

   A. 3
   B. 1
   C. 0
   D. 2

5. What is the following function determining?

```
int fn(int a, int b)
{
        if (b==0) return 0;
        if (b==1) return a;
        return a+fn(a, b-1);
}
```

   A. a+b where a and b are integers
   B. a+b where a and b are non-negative integers
   C. a*b where a and b are integers
   D. a*b where a and b are non-negative integers

6. The following program

```
 main()
{
inc(); inc(); inc();
}
inc()
{
static int x;
printf("%d", ++x);
}
```

   A. prints 012
   B. prints 123
   C. prints 3 consecutive, but unpredictable numbers
   D. prints 111

7. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   static int x = 3;
   x++;
   if (x <= 5)
   {
      printf("hi");
      main();
   }
}
```

   A. Run time error
   B. hi
   C. Infinite hi
   D. hi hi

8. Which function definition will run correctly?

   A.
```
int sum(int a, int b)
    return (a + b);
```

B. int sum(int a, int b)
{return (a + b);}
C. int sum(a, b)
return (a + b);
D. Both (a) and (b)

9. The value obtained in the function is given back to main by using _____ keyword?
A. return
B. static
C. new
D. volatile

10. The pointer ptr points to which string?

```
char *ptr;
char myString[]="abcdefg";
ptr=myString
ptr+=5;
```

A. fg
B. efg
C. defg
D. cdefg

11. What is the output of this code having void return-type function?

```
#include <stdio.h>
void foo()
{
   return 1;
}
void main()
{
   int x = 0;
   x = foo();
   printf("%d", x);
}
```

A. 1
B. 0
C. Runtime error
D. Compile time error

12. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   m();
   printf("%d", x);
}
int x;
void m()
{
   x = 4;
```

```
}
```

A. 4
B. Compile time error
C. 0
D. Undefined

13. What is the output of this C code?

```
#include <stdio.h>
int x;
void main()
{
   printf("%d", x);
}
```

A. Junk value
B. Run time error
C. 0
D. Undefined

14. Which of the following is an external variable?

```
#include <stdio.h>
int func (int a)
{
   int b;
   return b;
}
int main()
{
   int c;
   func (c);
}
int d;
```

A. a
B. b
C. c
D. d

15. What is the output of this C code?

```
#include <stdio.h>
static int x = 5;
void main()
{
   x = 9;
   {
      int x = 4;
   }
   printf("%d", x);
}
```

A. 9
B. 4
C. 5

D. 0

16. What is the output of this C code?

```
#include <stdio.h>
int x = 5;
void main()
{
   int x = 3;
   m();
   printf("%d", x);
}
void m()
{
   x = 8;
   n();
}
void n()
{
   printf("%d", x);
}
```

a) 8 3
b) 3 8
c) 8 5
d) 5 3

17. What is the output of this C code?

```
#include <stdio.h>
int x;
void main()
{
   m();
   printf("%d", x);
}
void m()
{
   x = 4;
}
```

a) 0
b) 4
c) Compile time error
d) Undefined

18. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   {
      int x = 8;
   }
   printf("%d", x);
```

```
}
```

a) 8
b) 0
c) Undefined
d) Compile time error

19. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   m();
   m();
}
void m()
{
   static int x = 5;
   x++;
   printf("%d", x);
}
```

a) 6 7
b) 6 6
c) 5 5
d) 5 6

20. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   static int x;
   printf("x is %d", x);
}
```

a) 0
b) 1
c) Junk value
d) Run time error

21. What is the output of this C code?

```
#include <stdio.h>
static int x;
void main()
{
   int x;
   printf("x is %d", x);
}
```

a) 0
b) Garbage value
c) Run time error
d) Nothing

22. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   static int x;
   if (x++ < 2)
   main();
}
```

a) Infinite calls to main
b) Run time error
c) Varies
d) main is called twice

23. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   foo();
   foo();
}
void foo()
{
   int i = 11;
   printf("%d ", i);
   static int j = 12;
   j = j + 1;
   printf("%d\n", j);
}
```

a) 11 12 11 12
b) 11 13 11 14
c) 11 12 11 13
d) Compile time error

24. Comment on the output of this C code?

```
#include <stdio.h>
void func();
int main()
{
   static int b = 20;
   func();
}
void func()
{
   static int b;
   printf("%d", b);
}
```

a) Output will be 0
b) Output will be 20
c) Output will be a garbage value
d) Compile time error due to re-declaration of static variable

25. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   char *p = NULL;
   char *q = 0;
   if (p)
      printf(" p ");
   else
      printf("nullp");
   if (q)
      printf("q\n");
   else
      printf(" nullq\n");
}
```

a) nullp nullq
b) Depends on the compiler
c) x nullq where x can be p or nullp depending on the value of NULL
d) p q

26. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   int i = 10;
   void *p = &i;
   printf("%d\n", (int)*p);
   return 0;
}
```

a) Compile time error
b) Segmentation fault/runtime crash
c) 10
d) Undefined behaviour

27. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   int *ptr, a = 10;
   ptr = &a;
   *ptr += 1;
   printf("%d,%d/n", *ptr, a);
}
```

a) 10,10
b) 10,11
c) 11,10
d) 11,11

28. What is the output of this C code?

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

```
int x = 0;
void main()
{
   int *ptr = &x;
   printf("%p\n", ptr);
   x++;
   printf("%p\n ", ptr);
}
```

a) Same address
b) Different address
c) Compile time error
d) Varies

29. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   int x = 0;
   int *ptr = &x;
   printf("%d\n", *ptr);
}
```

a) Address of x
b) Junk value
c) 0
d) Run time error

30. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   int i = 97, *p = &i;
   foo(&i);
   printf("%d ", *p);
}
void foo(int *p)
{
   int j = 2;
   p = &j;
   printf("%d ", *p);
}
```

a) 2 97
b) 2 2
c) Compile time error
d) Segmentation fault/code crash

31. What is the difference between call by value and call by reference?
32. Write a program to swap two integers using call-by-value method of passing arguments to a function.
33. Write a program to swap two integers using call-by-reference method of passing arguments to a function.

34. What is the output of this C code?

```
#include <stdio.h>
void m(int *p)
{
    int i = 0;
    for(i = 0;i < 5; i++)
    printf("%d\t", p[i]);
}
void main()
{
    int a[5] = {6, 5, 3};
    m(&a);
}
```

a) 0 0 0 0 0
b) 6 5 3 0 0
c) Run time error
d) 6 5 3 junk junk

35. What is the output of this C code?

```
#include <stdio.h>

void main()

{

   char *s= "hello";

   char *p = s;

   printf("%c\t%c", *(p + 3),  s[1]);

}
```

a) h e
b) l l
c) l o
d) l e

36. What is the output of this C code?

```
#include <stdio.h>

int main()

{

   const int ary[4] = {1, 2, 3, 4};

   int *p;

   p = ary + 3;

   *p = 5;

   printf("%d\n", ary[3]);

}
```

a) 4
b) 5
c) Compile time error
d) 3

37. What is the output of the code given below?

```
#include <stdio.h>

int main()

{

   int ary[4] = {1, 2, 3, 4};

   int p[4];

   p = ary;

   printf("%d\n", p[1]);

}
```

a) 1
b) Compile time error
c) Undefined behaviour
d) 2

38. What is the output of this C code?

```
#include <stdio.h>

void main()

{

   char *s= "hello";

   char *p = s + 2;

   printf("%c\t%c", *p, s[1]);

}
```

a) l e
b) h e
c) l l
d) h l

39. What is the output of this C code?

```
#include <stdio.h>
int main()
{
   char *str = "hello world";
   char strary[] = "hello world";
   printf("%d %d\n", sizeof(str), sizeof(strary));
   return 0;
}
```

a) 11 11
b) 12 12

c) 4 12
d) 4 11

40. What is the output of this C code?

```
#include <stdio.h>
int mul(int a, int b, int c)
{
   return a * b * c;
}
void main()
{
   int (*function_pointer)(int, int, int);
   function_pointer  =  mul;
   printf("The product of three numbers is:%d",
   function_pointer(2, 3, 4));
}
```

a) The product of three numbers is:24
b) Run time error
c) Nothing
d) Varies

41. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   int k = 5;
   int *p = &k;
   int **m  = &p;
   printf("%d%d%d\n", k, *p, **m);
}
```

a) 5 5 5
b) 5 5 junk value
c) 5 junk junk
d) Run time error

42. What is the output of this C code?

```
#include <stdio.h>

void main()

{

   int k = 5;

   int *p = &k;

   int **m  = &p;

   **m = 6;

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

}
```

a) 5
b) Compile time error
c) 6
d) Junk

43. What is the output of this C code?

```
#include <stdio.h>
void main()
{
   int k = 5;
   int *p = &k;
   int **m  = &p;
   printf("%d%d%d\n", k, *p, **p);
}
```

a) 5 5 5
b) 5 5 junk value
c) 5 junk junk
d) Compile time error

44. Write a C program to create, initialize, assign and access a pointer variable.
45. Write a C program to read array elements and print with addresses.
46. Write a C program to count vowels and consonants in a string using pointer.
47. Write a C program to change the value of constant integer using pointers.
48. Write a program using pointer variables to read a character until * is entered. If the character is in upper case, print it in lower case and vice versa. Also count the number of upper and lower case characters entered.
49. Write a program that uses an array of function pointers, to add, subtract, multiply, or divide two given numbers.
50. Write a function to calculate roots of a quadratic equation. The function must accept arguments and return result using pointers.
51. Write a program using pointers to insert a value in an array.
52. Write a program using pointers to search a value from an array.
53. Write a program to compare two arrays using pointers.
54. Write a program to add two integers using functions
55. Write a program using pointers to read and print an array of n numbers, then find out the smallest number. Also print the position of the smallest number.
56. Write a program using pointers which reads a string and then scans each character to count the number of upper and lower case characters entered.
57. Write a program using pointers to read and print a text. Also count the number of characters, words, and lines in the text.
58. Write a program to copy a character array in another character array using pointers
59. Write a program to copy n characters of a character array from the mth position in another character array using pointers.
60. Write a program to subtract two floating point numbers using functions.
61. Write a function that accepts a string using pointers. In the function, delete all the occurrences of a given character and display the modified string on the screen.
62. Write a program to reverse a string using pointers.
63. The operator * signifies a
    a) Referencing operator
    b) Dereferencing operator

c) Address operator
d) None of these

64. What is the output of following program?

```
# include <stdio.h>
void fun(int x)
{
   x = 30;
}

int main()
{
  int y = 20;
fun(y);
  printf("%d", y);
  return 0;
}
```

A. 30
B. 20
C. Compiler Error
D. Runtime Error

65. What is the output of following program?

```
#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

66. Output of following program?

```
# include <stdio.h>
void fun(int *ptr)
{
   *ptr = 30;
}

int main()
{
  int y = 20;
  fun(&y);
  printf("%d", y);

  return 0;
}
```

A. 20
B. 30
C. Compiler Error
D. Runtime Error

67. 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

68. Output of following program?

```
#include <stdio.h>
int main()
{
   int *ptr;
   int x;

   ptr = &x;
   *ptr = 0;

   printf(" x = %d\n", x);
   printf(" *ptr = %d\n", *ptr);

   *ptr += 5;
   printf(" x  = %d\n", x);
   printf(" *ptr = %d\n", *ptr);
```

```
    (*ptr)++;
    printf(" x = %d\n", x);
    printf(" *ptr = %d\n", *ptr);

    return 0;
}
```

A.
```
x = 0
*ptr = 0
x = 5
*ptr = 5
x = 6
*ptr = 6
```

B.
```
x = garbage value
*ptr = 0
x = garbage value
*ptr = 5
x = garbage value
*ptr = 6
```

C.
```
x = 0
*ptr = 0
x = 5
*ptr = 5
x = garbage value
*ptr = garbage value
```

D.
```
x = 0
*ptr = 0
x = 0
*ptr = 0
x = 0
*ptr = 0
```

69.
```
#include<stdio.h>
int main()
{
    int arr[] = {10, 20, 30, 40, 50, 60};
    int *ptr1 = arr;
    int *ptr2 = arr + 5;
    printf("Number of elements between two pointer are: %d",
                        (ptr2 - ptr1));
    printf("Number of bytes between two pointers are: %d",
                      (char*)ptr2 - (char*) ptr1);
    return 0;
}
```

Assume that an int variable takes 4 bytes and a char variable takes 1 byte

A. Number of elements between two pointer are: 5.
   Number of bytes between two pointers are: 20

B. Number of elements between two pointer are: 20.
   Number of bytes between two pointers are: 20

C. Number of elements between two pointer are: 5.
   Number of bytes between two pointers are: 5

D. Compiler Error
E. Runtime Error

70. Write a C function to sort an array of ten integer values in ascending order using pointers.
71. Write a C program to print the elements of the array in reverse order using a pointer.
72. Write a C program to find the max of an integral data set. The program will ask the user to input the number of data values in the set and each value. Then your program will show the max of the data set. Your C program will use a function that accepts the array of data values and its size. The return from the function is the pointer that points to the max value.
73. Write a program to find maximum number in array using pointer.
74. Write a C program to swap three numbers in cyclic order using call by reference.

Find the output (or errors) for the following programs:

75.
```
void main()
{
    int const * p=5;
    printf("%d",++(*p));
}
```

76.
```
main()
{
static int var = 5;
printf("%d",var--);
if(var)
    main();
}
```

77.
```
main()
{
    int c[ ]={2.8,3.4,4,6.7,5};
    int j,*p=c,*q=c;
   for(j=0;j<5;j++)
   {
       printf(" %d ",*c);
       ++q;
    }
  for(j=0;j<5;j++)
  {
       printf(" %d ",*p);
       ++p;
   }
}
```

78.
```
main()
{
```

```
        char string[]="Hello
        World"; display(string);
}
void display(char *string)
{
        printf("%s",string);
}
```

79.
```
main( )
{
void *vp;
char ch = 'g', *cp =
"goofy"; int j = 20;
vp = &ch;
printf("%c", *(char *)vp);
vp = &j;
printf("%d",*(int *)vp);
vp = cp;
printf("%s",(char *)vp + 3);
}
```

80.
```
main ( )
{
    static char *s[ ] = {"black", "white", "yellow",
"violet"};
    char **ptr[ ] = {s+3, s+2, s+1, s}, ***p;
     p = ptr;
     **++p;
     printf("%s",*--*++p + 3);
 }
```

81.
```
main()
{
        char *str1="abcd";
        char str2[]="abcd";
        printf("%d %d %d",sizeof(str1),sizeof(str2),sizeof("abcd"));
}
```

82.
```
main()
{
        char *p; p="%d\n";
        p++;
        p++;
        printf(p-2,300);
```

```
}
```

83.
```
void main()
{
        static int i=5;
        if(--i)
        {
        main();
        printf("%d ",i);
        }
}
```

84.
```
main()
{
        int i=5,j=10;
         i=i&=j&&10;
        printf("%d %d",i,j);
}
```

85.
```
main()
{
        int i = 257;
        int *iPtr = &i;
        printf("%d %d", *((char*)iPtr), *((char*)iPtr+1) );
}
```

86.
```
void main()
{
        int count=10,*temp,sum=0; temp=&count;
        *temp=20; temp=&sum;
        *temp=count;
        printf("%d %d %d ",count,*temp,sum);
}
```

87.
```
main()
{
        static i=3;
         printf("%d",i--);
        return i>0 ? main():0;
}
```

88.
```
char* myFunc (char *ptr)
{
        ptr += 3;
        return (ptr);
}
int main()
{
```

```
        char *x, *y;
        x = "HELLO";
        y = myFunc (x);
        printf ("y = %s \n", y);
        return 0;
    }
```

89. In the following program add a stmt in the function fun such that the address of ‘a’ gets stored in ‘j’.

```
main()
{
        int * j;
        void fun(int **);
        fun(&j);
}
void fun(int **k)
{
        int a =0;
        /* add a stmt here*/
}
```

90. Define pointer to function that take argument as character pointer and return void pointer.