Q1. What will be the output of the C program?

```c
#include<stdio.h>

int main()
{
    const int a = 5;
    const int *ptr;
    ptr = &a;
    *ptr = 10;
    printf("%d\n", a);
    return 0;
}
```

A. Compilation error-

B. Garbage Value

C. Address

D. 5

Q2. Fill the question mark to get "void pointer" as an output?

```c
#include<stdio.h>
int main(){
 char *ptr = "void pointer";
 void *vptr;
 vptr = &ptr;
 printf("%s" , ?);
 return 0;
}
```

A. *(char **)vptr -

B. (char **)vptr

C. *(char *)vptr

D. (char *)vptr

Q3. What will be the output of the C program?

```c
#include<stdio.h>
int main(){
    signed a;
    unsigned b;
    a = 6u + -16 + 16u + -6;
    b = a + 1;
    if(a == b)
    printf("%d %d",a,b);
    else
    printf("%u %u",a, b);
    return 0;
}
```

A. Compilation error

B. 0 0

C. 0 1-

D. address address

Q4. What will be the output of the C program?

```c
#include<stdio.h>
int main()
{
    typedef int num;
    num num1 = 5;
    printf("%d", num1);
    return 0;
}
```

A. Compilation error

B. 1

C. 5-

D. Runtime error

Q5. What will be the output of the C program?

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

```
{
    int x = 2;
    (x & 1) ? printf("true") : printf("false");
    return 0;
}
```

A. Compilation error

B. true

C. false-

D. Runtime error

Q6. What will be the output of the C program?

```
#include<stdio.h>
int main()
{
    int a = 1, b = 3, c;
    c = b <<  a;
    b = c * (b * (++a)--);
    a = a >> b;
    printf("%d",b);
    return 0;
}
```

A. 36-

B. Compilation error

C. 30

D. 24

Q7. What will be the output of the C program?

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

```
    a = 250;
}
```

A. 250

B. 0

C. 1-

D. Some Garbage value

Q8. What will be the output of the C program?

```
#include<stdio.h>
#include<stdlib.h>
int* fun();
int main()
{
    int *ptr = fun();
    printf("%d", *ptr);
    return 0;
}
int* fun()
{
    int *ptr1 = (int*) malloc(sizeof(int));
    *ptr1 = 25;
    return ptr1;
}
```

A. Compilation error

B. 25-

C. address

D. Runtime error

Q9. What will be the output of the C program?

```
#include<stdio.h>
void fun(int*, int);
void (*ptr[1])(int*, int);
int main()
{
    int a = 2;
    int b = 4;
    ptr[0] = fun;
```

```
    ptr[0](&a, b);
    printf("%d %d ", a, b);
    return 0;
}
void fun(int *p, int q)
{
    int tmp = *p;
    *p = q;
    q = tmp;
}
```

A. 2 2

B. 4 2

C. 4 4-

D. 2 4

Q10. Guess the output?

```
#include <stdio.h>
void mysteriousFunction(int x) {
    x += 10;
}
int main() {
    int num = 5;
    mysteriousFunction(num);
    printf("The value of num is: %d\n", num);

    return 0;
}
```

A. The value of num is: 5-

B. The value of num is: 15

C. The value of num is: 10

D. The program will not compile.

Q11. Which among the given options compares atmost n characters of string ch to string s?

a) int strncmp(ch, s, n)-
b) int strcmp(ch, s)
c) int strncmp(s, ch, n)
d) int strcmp(s, ch)

Q12.What will be the output of the following C code?

```
char str1[15];
char str2[15];
int mat;
strcpy(str1, "abcdef");
strcpy(str2, "ABCDEF");
mat= strncmp(str1, str2, 4);
if(mat< 0)
printf("str1 is not greater than str2");
else if(mat> 0)
printf("str2 is is not greater than str1");
else
printf("both are equal");
```

a) str1 is not greater than str2
b) str2 is not greater than str1-
c) both are equal
d) error in given code

Q13.Point out the error (if any) in the following C code?

```
#include<stdio.h>
enum hello
{
    a,b,c;
};
main()
{
    enum hello m;
    printf("%d",m);
}
```

a) No error
b) Error in the statement: a,b,c; -
c) Error in the statement: enum hello m;
d) Error in the statement: printf(“%d”,m);

Q14. Guess the output.

```c
#include <stdio.h>

enum Days {
    SUNDAY,
    MONDAY,
    TUESDAY,
    WEDNESDAY,
    THURSDAY,
    FRIDAY,
    SUNDAY,
    SATURDAY
};

int main() {
    enum Days today = WEDNESDAY;
    printf("The value of today is: %d\n", today);

    return 0;
}
```

A. No error, the code will compile and run successfully.
B. Compilation error due to repeating variables in the enum.-
C. The value of `today` will be 2.
D. The value of `today` will be 6.

Q15.What is the value of the variable a after executing this program? Please choose the correct option:

```c
#include <stdio.h>

int main() {
    int a = printf("hi");
    printf("\nThe value of a is: %d\n", a);
```

```
    return 0;
}
```

A. The value of a is 2.

B. The value of a is 4.

C. The value of a is 6.

D. The value of a is the number of characters printed by the printf function.-

Q16. What will be the output of the following C code?

```
#include <stdio.h>
   struct p
    {
       int x;
       int y;
   };
   int main()
   {
       struct p p1[] = {1, 92, 3, 94, 5, 96};
       struct p *ptr1 = p1;
       int x = (sizeof(p1) / 5);
       if (x == 3)
           printf("%d %d\n", ptr1->x, (ptr1 + x - 1)->x);
       else
           printf("false\n");
   }
```

a) Compile time error

b) 1 5

c) Undefined behaviour

d) false-

Q17. What will be the output of the C program ?

```c
#include<stdio.h>
#define clrscr() 50
int main()
{
    clrscr();
    printf("%d\n",clrscr());
    return 0;
}
```

A. Compilation error

B. Runtime error

C. 50-

D. none of the above

Q18. What will be the output of the C program?

```c
#include<stdio.h>
#define a =
int main()
{
    int num a 6;
    printf("%d",num);
    return 0;
}
```

A. 6-

B. Compilation error

C. Garbage value

D. Runtime error

Q19. What will be the output of the following C code?

```c
#include<stdio.h>
void main()
{
    #ifndef max
    printf("hello");
    #endif
    printf("hi");
}
```

a) hello
b) hellohi
c) error
d) hi

Q20. What will be the output of the C program?

```c
#include<stdio.h>
int main()
{
    EOF++;
    printf("%d", EOF);
    return 0;
}
```

A. Compilation Error-

B.-1

C. 0

D. 1

Q21. If an error occurs, the function fseek() will returns

A. NULL

B. 0

C.-1

D. non-zero values-

Q22. Write a C program to find the intersection point of two linked lists.

Q23. Write a C program to swap nodes in a linked list without swapping the data.

Q24. Create a C program to determine the minimum and maximum number of nodes between critical points in a linked list.

Q25. Write a C program to delete N nodes after every M nodes in a linked list.

Q26. Implement a C program to merge two sorted linked lists into a single sorted list.

Q27. Develop a C program to print the kth node from the end of a linked list.

Q28. Write a C program to sort a linked list containing only 0s, 1s, and 2s.

Q29. Create a C program to double a number represented as a linked list.

Q30. Write a C program to find the middle node of a linked list.

Q31. Implement a C program to check whether a linked list is circular or not.

Q32. Develop a C program to find the starting point of a circular linked list.

Q33. Write a C program to find the minimum number of bracket reversals needed to make a string balanced.

Q34. Create a C program to remove all adjacent duplicates in a given string.

Q35. Implement a C program to solve the Celebrity Problem.

Q36. Write a C program to find the next greater element for each node in a linked list.

Q37. Develop a C program to implement N stacks in a single array.

Q38. Create a C program to implement the Online Stock Span problem as described in Leetcode.

Q39. Write a C program to check if a given word is valid after performing a series of substitutions as per Leetcode.

Q40. Implement a C program to decode strings as per the rules defined in Leetcode.

Q41. Write a C program to solve the Car Fleet- I problem as described in Leetcode.

Q42. Develop a C program to solve the Car Fleet - II problem as described in Leetcode.

Q43. Create a C program to simplify a given Unix-like file path as per Leetcode.

Q44. Write a C program to remove k digits from a given number to form the smallest possible number.

Q45. Implement a C program to find the length of the longest valid parentheses substring.

Q46. Develop a C program to implement a queue using stacks.

Q47. Create a C program to implement stacks using queues.

Q48. Write a C program to implement "K" queues in a single array.

Q49. Develop a C program to find the sum of the minimum or maximum elements in all subarrays of size "k".

Q50. Write a C program to find the inorder successor and inorder predecessor of a given node in a Binary Search Tree (BST).

Q51. Create a C program to construct a Binary Search Tree (BST) from its preorder traversal.

Q52. Develop a C program to convert a normal Binary Search Tree (BST) into a Balanced BST.

Q53. Implement a C program to merge two Binary Search Trees (BSTs) into a single BST.

Q54. Write a C program to count pairs from two Binary Search Trees (BSTs) whose sum is equal to a given value "X".

Q55. Develop a C program to find the median of a Binary Search Tree (BST) in O(n) time and O(1) space.

Q56. Create a C program to count the number of nodes in a Binary Search Tree (BST) that lie in a given range.

Q57. Write a C program to replace every element in an array with the least greater element on its right side.

Q58. Implement a C program to check whether a given array represents a valid preorder traversal of a Binary Search Tree (BST) or not.

Q59. Develop a C program to check whether a Binary Search Tree (BST) contains a dead end or not.

Q60. Write a C program to flatten a Binary Search Tree (BST) into a sorted linked list.