26. What is the result of this program?
```
#include<stdio.h>
#include<string.h>
intmain ( )
{
char *xyz = "Test\0\0 String\0\n",
printf ("%s\n", xyz);
}
```

A. Test   B. String   C. Blank Line   D. Won't compile

27. What will be the output?
```
int main ( )
{
        printf( "%d", 010  ) ;
}
```

A. 010   B. 00000010   C. 8   D. 64

28. What will be the output of following program?
```
int a=2;
switch ( a ) {
case 1:
    printf("b");
    break;
case 2:
    printf("c");
default:
    break;
    printf(" d");
}
```
A. b   B. c   C. c d   D. none of the above

29. What will be the output of following program?
```
#include <stdio.h>
void swap(int x, int y);
int main()
{
int x = 4;
int y = 2;
printf("%d, %d,",x,y);
   swap(x,y);
printf(" %d, %d\n",y,x);
```

```
}
void swap(int first, int second)
{
int temp;
   temp = second;
   second = first;
   first = temp;
}
```
Output of above program is going to be

A. 4, 2, 4, 2   B. 4, 2, 2, 2   C. 4, 2, 4, 4   D. 4,2,2,4

30. What is an accessor?

A. An accessor is a class operation that allow to modify the state of an object.
B. An accessor is a class operation that does not modify the state of an object and the accessor functions need to be declared as const operations
C. An accessor is a class operation that does not modify the state of an object and the accessor functions should not be declared as const operations
D. None of Above

31. Which of the following suffers data loss in a 32 bit binary if we perform cross compilation?
i. type cast from int to long.
ii. type cast from long to int.
iii. type cast from double to int.

A. All of them   B. i & ii   C. only iii   D. none

32. What is the expected result of the below program?
```
class A
{
        public:
        A() { }
        A(int a)  {}
        ~A() {}
        ~A(int a) {}
}
```
A. compiles and executes.   B. compile time error.
C. run-time error.   D. None of above

33. A static function can access
   i. static functions
   ii. static variables
   iii. only static members

A. i & ii   B. iii   C. i   D. ii

34. What is a pure virtual function?

   A. A pure virtual function is a function declared in a base class that has no definition relative to the base.

   B. A pure virtual function is a function declared in a child class that has no definition relative to the base.

   C. A pure virtual function is a function declared in a base class that has no definition relative to the child.

   D. None Of Above

35. void *(*foo)(int *); is

   A. pointer to a function returning void pointer.
   B. pointer to a pointer of function returning void
   C. depends on compiler
   D. is not a valid declaration

36. What is the expected result of the below program?

```
# include <iostream>
using namespace std;
class Sample
{
public:
        int *ptr;
        Sample(int i)
           {
              ptr = new int(i);
           }
           ~Sample()
           {
              delete ptr;
           }
           void PrintVal()
           {
              cout<< "The value is " << *ptr;
           }
};
void SomeFunc(Sample x)
{
        cout<< "String" <<endl;
}
int main()
{
        Sample s1= 10;
        SomeFunc(s1);
        s1.PrintVal();
}
```

A. Compile time error.   B. Run time error.
C. Prints "String" and Run-time error.   D. Prints "String" Only

37. Given two sorted lists of size m and n respectively.The number of comparisons needed in the worst case by the merge sort algorithm will be?

A. m   B. (max,n)   C. min(m,n)   D. m+n-1

38. The complexity of Bubble sort algorithm is

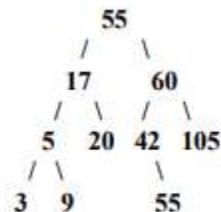A. O(n)   B. O(log n)   C. O(n2)   D. O(n log n)

39. Convert the following Infix expression to Postfix form using a stack :
     x + y * z + (p * q + r) * s
Follow usual precedence rule and assume that the expression is legal.

A. xyz+*pq*r+s*+   B. xyz*+pq*r+s+*   C. xyz*+pq*r+s*+   D.None of Above

40. Is this a binary search tree?

```
            55
           /   \
         17      60
        / \     / \
       5   20  42  105
      / \          \
     3   9          55
```

A. Yes
B. No, because of value "55" wrongly placed
C. No, because of value "60" wrongly placed
D. No, because of value "9" wrongly placed

41. ………….. refers to a situation in which a process is ready to execute but is continuously denied access to a processor in deference to other processes.

A. Synchronization   B. Mutual Exclusion   C. Dead lock   D. Starvation

42. Algorithm to be used for reducing External Fragmentation

A. MFT   B. MVT   C. Compaction   D. Deadlock avoidance

43. What is the functionality of the following piece of code?

```
public int function()
{
    Node temp = tail.getPrev();
    tail.setPrev(temp.getPrev());
    temp.getPrev().setNext(tail);
    size--;
    return temp.getItem();
}
```

A. Return the element at the tail of the list but do not remove it
B. Return the element at the tail of the list and remove it from the list
C. Return the last but one element from the list but do not remove it
D. Return the last but one element at the tail of the list and remove it from the list

44. What is a dequeue?

A. A queue with insert/delete defined for both front and rear ends of the queue
B. A queue implemented with a doubly linked list
C. A queue implemented with both singly and doubly linked lists
D. None of the mentioned

45. Consider the following code snippet to search an element in a linked list:

```
struct Node
{
intval;
        struct Node* next;
}*head;
intlinear_search(int value)
{
    struct Node *temp = head->next;
    while(temp != 0)
    {
        if(temp->val == value)
          return 1;
        _____;
    }
    return 0;
}
```
Which of the following lines should be inserted to complete the above code?

A. temp=next   B. temp->next=temp   C. temp=temp->next   D. return 0

46. Which of the following is non-linear data structure

A. Stacks   B. List   C. String   D.Tree

47. In priority queue insertion and deletion takes place at..

    A. Front, rear end   B. Only at rear end   C. Any position   D. Only at front

48. In ………………. only one process at a time is allowed into its critical section, among all processes that have critical sections for the same resource.

    A. Mutual Exclusion
    B. Synchronization
    C. Deadlock
    D. Starvation

49. A semaphore :

    A. is a binary mutex
    B. must be accessed from only one process
    C. can be accessed from multiple processes
    D. none of the mentioned

50. The strategy of making processes that are logically runnable to be temporarily suspended is called :

    A. Non preemptive scheduling
    B. Preemptive scheduling
    C. Shortest job first
    D. First come First served