

---

**Question Bank – Advanced C++ & Data Structures (Solved)**

**Fill in the Blanks:**

1. A \_\_\_\_\_ is a special member function used to initialize the data members of a class. (constructor)
2. The default access for members of a class is \_\_\_\_\_. (private)
3. Member functions of a class are normally made \_\_\_\_\_ and data members of a class are normally made \_\_\_\_\_. (public, private)
4. Inheritance enables \_\_\_\_\_ which saves time in development , and encourages using previously proven and high quality software. (reusability)
5. The three member access specifiers are \_\_\_\_\_, \_\_\_\_\_ and \_\_\_\_\_. (public, private, protected)
6. A “has a” relationship between classes represents \_\_\_\_\_ and an “is a” relationship between classes represent \_\_\_\_\_. (containment, inheritance)
7. A pure virtual function is specified by placing \_\_\_\_\_ at the end of its prototype in the class definition. (=0;)
8. A \_\_\_\_\_ operator is called as de-referencing operator. (\*)
9. The size of a class with no data members and member functions is \_\_\_\_\_ bytes. (1)
10. A class is called as abstract base class if it has a \_\_\_\_\_ function. (pure virtual)
11. A queue is \_\_\_\_\_ list. (linear)
12. A \_\_\_\_\_ is a function that invokes itself. (Recursion)
13. Run-time or dynamic allocation of memory may be done using the C++ operator \_\_\_\_\_. (new)
14. Operator overloading permits to extend the applicability of existing C++ operators so that they work with new \_\_\_\_\_ or \_\_\_\_\_. (data types, objects)
15. A \_\_\_\_\_ is a set of instance or values. (object)
16. A \_\_\_\_\_ is a linear list in which additions and deletions take place at the same end. (stack)
17. A \_\_\_\_\_ is a linear list in which additions and deletions take place at different ends. (queue)
18. C++ programming language was designed and developed by \_\_\_\_\_ at \_\_\_\_\_. (Bjarne Stroustrup, AT&T Bell Labs)
19. C++ is a \_\_\_\_\_ programming language with \_\_\_\_\_ extensions. (procedural, object oriented)
20. A variable definition defines a \_\_\_\_\_ of the variable and reserves \_\_\_\_\_ for it. (instance, memory)
21. In C++ a keyword cannot be used as \_\_\_\_\_. (identifier)
22. Literals are always \_\_\_\_\_. (constants)
23. In C++, \_\_\_\_\_ operator has the lowest precedence. (,)
24. Name mangling is done by the \_\_\_\_\_. (C++ compiler)
25. Pointers are \_\_\_\_\_ that contain the addresses of other variables and \_\_\_\_\_. (variables, functions)
26. A program can use the address of any variable in an expression, except variables declared with the \_\_\_\_\_ storage class. (register)
27. One cannot perform pointer arithmetic on a \_\_\_\_\_ pointer without \_\_\_\_\_. (void , cast)
28. new operator allocates memory blocks from the \_\_\_\_\_. (Heap)
29. The new operator throws a \_\_\_\_\_ when heap is exhausted. (runtime exception)
30. The <cassert> header is used for \_\_\_\_\_. (debugging)
31. The constructor and destructor of a class are called \_\_\_\_\_. (automatically)
32. Two or more functions may have the same name, as long as their \_\_\_\_\_ are different. (parameter lists).
33. A constructor with default arguments for all its parameters is called a \_\_\_\_\_ constructor. (default)

**Question Bank – Advanced C++ & Data Structures (Solved)**

34. Static member functions can access only the \_\_\_\_\_ data members of a class. (Static)
35. The two types of polymorphism is : \_\_\_\_\_ & \_\_\_\_\_. (Run time and compile time.)
36. A file stream is an extension of a \_\_\_\_\_ stream. (console)
37. The Standard Template Library(STL) is a library of \_\_\_\_\_ templates.(container class)
38. Run time polymorphism is \_\_\_\_\_ than the compile time polymorphism.
39. With private inheritance, public and protected members of the base class become \_\_\_\_\_ members of the derived class. (private)
40. A tree node with no children is called a \_\_\_\_\_ node. (leaf)
41. The shape of a binary tree is determined by the order in which values are inserted. (True)
42. In inorder traversal, the node's data is processed first, then the left and right nodes are visited. (True)
43. The STL stack container is an adapter for the \_\_\_\_\_, \_\_\_\_\_, and \_\_\_\_\_ STL containers. (vectors, lists, deque)
44. A data structure that points to an object of the same type, as itself is known as a \_\_\_\_\_ data structure. (self-referential)
45. After creating a linked list's head pointer, one should make sure it points to \_\_\_\_\_ before using it in any operations.(NULL)
46. \_\_\_\_\_ a node means adding it to a list, but not necessarily to the end. (inserting)
47. In a \_\_\_\_\_ list, the last node has a pointer to the first node. (circular).
48. The \_\_\_\_\_ block contains code that directly or indirectly might cause an exception to be thrown. (try)
49. When writing function or class template, one uses a \_\_\_\_\_ to specify a generic data type. (type-parameter)
50. A \_\_\_\_\_ template works with a specific data type. (specialized)
51. A \_\_\_\_\_ container uses keys to rapidly access elements. (associative)
52. \_\_\_\_\_ are pointer-like objects used to access information stored in a container.(Iterators)
53. In some cases the class must be declared(not defined ) before a reference is made to it. This is known as \_\_\_\_\_. (forward declaration).
54. \_\_\_\_\_ are blue prints of a function that can be applied to different data types. (templates)
55. \_\_\_\_\_ and \_\_\_\_\_ are two error objects. (clog, cerr)
56. We can convert a class data type to basic data type using \_\_\_\_\_. (member conversion function)

**Select True or False:**

1. Data items in a class may be public. (True)
2. Class members are public by default. (False)
3. Friend functions have access only to public members of the class. (False)
4. A function designed as public can be accessible by non-members of a class. (True)
5. Friendship is commutative. (False)
6. new and delete are operators. (True)
7. 'new' operator can be overloaded ? (True)
8. We can implement generic classes using templates (True)
9. Using operator overloading we can invent new operators. (False)
10. Inheritance helps in making a general class into a more specific class. (True)
11. A base class is never used to create objects. (False)
12. Classes can be inherited privately. (True)
13. It is legal to return local variables from a function which returns by reference. (False)
14. Constructors can be virtual like virtual destructors (False)

*Question Bank – Advanced C++ & Data Structures (Solved)*

15. C++ is a pure object oriented programming language. (False)
16. In C++ one can define a function within another function. (False)
17. A class encapsulates the implementation and interface of a user-defined data type and constitutes an abstract data type. (True)
18. In c++ an identifier can begin with a \$ sign. (True)
19. “My Variable” is a valid identifier in C++. (False)
20. In C++ one can get the memory addresses of variables and functions. (True)
21. %= is not a operator in C++. (False)
22. std::cout is a standard input stream. (True)
23. Preprocessor #define macro and inline functions use the same mechanism. (False)
24. All C++ functions are recursive. (True)
25. The ‘break’ keyword is only used in the switch..case statement. (False)
26. The new operator returns the address and size of the memory block that it allocates. (False)
27. The heap storage is used for local objects. (False)
28. A call to a function that returns a reference can appear on either side of an assignment. (True).
29. It is not necessary to initialize a reference to real object when it is declared. (False)
30. There can be a null reference. (False)
31. One can change the value of a reference after it is initialized. (False)
32. It is nothing wrong that a function returning a reference to an automatic variable. (False)
33. One can apply pointer arithmetic with reference variables. (False)
34. The preprocessor processes source code before the compiler does. (True)
35. A class is a basic unit of object-oriented programming. (False)
36. A function template defines a parameterized nonmember function, which enables a program to call the same function with different types of arguments. (True)
37. Destructors can be overloaded. (False)
38. Static data members cannot be private. (False)
39. Static member functions can use this pointer. (False)
40. One cannot use enumerations in a class. (False)
41. One cannot create an object of a virtual class. (False)
42. The push operation inserts an element at the end of a stack. (False)
43. It is not necessary for each node in a linked list to have a self-referential pointer. (False)
44. In physical memory, the nodes in a linked list may be scattered around. (True)
45. When the head pointer points to NULL, it signifies an empty list. (True)
46. Linked list are not superior to STL vectors. (False)
47. Deleting a node in a linked list is a simple matter of using the delete operator to free the node’s memory. (False)
48. A class that builds a linked list should destroy the list in the class destructor. (True)
49. Once an exception has been throwe, it is not possible for the program to jump back to the throw point. (True)
50. In C++, only one catch block can handle all the exceptions. (True)
51. It is not possible to rethrow an exception. (False)
52. There can be only one catch block in a program.(False)
53. When an exception if throw, but not caught, the program ignorers the error. (False)
54. A class object passed to a function template must overload any operators used on the class object by the template. (True)
55. In the function template definition it is not necessary to use each type parameter declared in the template prefix. (False)

**Question Bank – Advanced C++ & Data Structures (Solved)**

- 56. It is possible to overload a function template and an ordinary (non-template) function. (True)
- 57. A class template may not be used as a base class. (False)
- 58. When declaring an iterator from the STL, the compiler automatically creates the right kind, depending upon the container it is used with. (True)
- 59. 'ios' stream is derived from iostream. (False)
- 60. 'eof()' function returns zero value if the eofbit is set. (False)

**Select the correct answer:**

1. What is the output of the following code?

```
#include<iostream.h>
#include<string.h>
void main()
{
    cout<<strlen("Hello, World.\n")<<"\n";
}
```

- (a) 14                      (b) 13                      (c) 12                      (d) None

2. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    /* this is /* an example */ of nested comment */
    cout<<123<<endl;
}
```

- (a) 123                      (b) Compile time error                      (c) None                      (d) Run time Error

3. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    cout << ;
}
```

- (a) 1                      (b) Compile time error                      (c) NIL                      (d) None

4. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a = 20;
    int &n = a;
    n=a++;
    a=n++;
    cout<<a <<" ,"<<n<<endl;
}
```

- (a) 20, 20                      (b) 20, 21                      (c) 21, 22                      (d) None

**Question Bank – Advanced C++ & Data Structures (Solved)**

5. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a = 20,b=100;
    int &n = a;
    n=a++;
    n = &b;
    cout<<a <<" "<<n<<endl;
}
```

- (a) 21, 21                      (b) 20, 21                      (c) 21, 22                      (d) Error

6. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    bool a=10;
    cout<<a<<endl;
}
```

- (a) 10                              (b) false                              (c) 1                              (d) Error

7. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int main;
    main = 100;
    cout<<main++<<endl;
}
```

- (a) 101                      (b) 100                      (c) None                      (d) Error: one cannot use main as identifier

8. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a=0,x;
    x = ++a * --a;
    cout<<++a<<" "<<" "<<a++<<" "<<x<<endl;
}
```

- (a) 0, 0, 0                      (b) 2, 0, 0                      (c) 2, 2, 2                      (d) 3, 2, 2

9. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    a=32;
    cout<<a<<endl;
}
```

**Question Bank – Advanced C++ & Data Structures (Solved)**

```

        int a;
    }

```

(a) 32            (b) 0            (c) Compile time error            (d) Run time error

10. Which of the following is not valid expression?

- (a) 33 / 9 / 3            (b) 23 % ( 5 % 2 )            (c) 34 ( 7 / 3 )            (d) None

11. Evaluate the m%n++ expression, assuming m=24 and n=7

- (a) 4            (b) 3            (c) 2            (d) None

12. Evaluate the m%++n expression, assuming m=24 and n=7

- (a) 4            (b) 3            (c) 2            (d) None

13. Which of the following statement is true?

- (a) ! ( p || q ) is the same as !p || !q  
(b) !!p is the same as !p  
 (c) p && q || r is the same as p && ( q || r )  
 (d) None

14. Elements in an array are identified by a unique\_\_\_\_\_.

- (a) symbol            (b) order            (c) subscript            (d) data type

15. An address is a\_\_\_\_\_, while a pointer is a\_\_\_\_\_.

- (a) variable, location            (b) variable, position            (c) constant, variable            (e) None

16. 6.5 is a\_\_\_\_\_constant.

- (a) string literal            (b) float literal            (c) double literal            (d) character literal

17. To execute a C++ program, one first need to translate the source code into object code. This process is called\_\_\_\_\_.

- (a) translating            (b) sourcing            (c) compiling            (d) coding

18. What is wrong with the following program?

```

#include<iostream.h>
void main()
{
    do
    {
        int b=0;
        cout<<b;
        b++;
    }while(b!=10);
}

```

- (a) There is nothing wrong in the program.  
 (b) Variable 'b' must not be initialized in the loop  
(c) Variable 'b' must not be declared in the loop  
 (d) The condition for while loop is not valid

**Question Bank – Advanced C++ & Data Structures (Solved)**

19. Sending a copy of data to a program module is called\_\_\_\_\_.  
 (a) recursion (b) passing a reference (c) passing a value (d) None

20. Each generic type in a template function definition is preceded by the keyword\_\_\_\_\_.  
 (a) class (b) type (c) function (d) template

21. Which of the followings is not a C++ ooperator?  
 (a) ^= (b) .\* (c) &= (d) ->>

22. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    char p[]="This is a test";
    cout<<sizeof(p)<<" "<<strlen(p);
}

```

(a) 14, 14 (b) 15, 14 (c) 14, 15 (d) 15,15

23. What is wrong with the following program?

```
#include<iostream.h>
void main()
{
    int a[5] = {0};
    for(int i=0;i<2;i++)
    a[i]=i;
    for(int i=0;i<5;i++)
    cout<<a[i]<<endl;
}

```

(a) Array ‘a’ is not initialized properly (b) There is no problem  
 (c) Redeclaration of variable ‘i’ (d) There is a run time error

24. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a[5] = {100,2,3,22,400};
    int b[5];
    b=a;
    for(int i=0;i<5;i++)
    cout<<b[i]<<endl;
}

```

(a) 100,2,3,22,400 (b) garbage values (c) error (d) None

25. What is the output of the following code?

```
#include<iostream.h>
void main()

```

**Question Bank – Advanced C++ & Data Structures (Solved)**

```
{
    int a[5] = {1,2,3};
    for(int i=0;i<5;i++)
        cout<<a[i]<<endl;
}
```

- (a) No output                      (b) 1 2 3 garbage garbage  
(c) 1 2 3 0 0                      (d) There is a run time error

26. To delete a dynamically allocated array named 'a', the correct statement is

- (a) delete a;                      (b) delete a[0];                      (c) delete []a;                      (d) delete [0]a;

27. Which of the followings is not a valid assignment statement?

- (a) total = 9;                      (b) name = "CDAC";  
(c) profit = 123.123;                      (d) A = 'A';

28. When do preprocessor directives execute?

- (a) Before the compiler compiles the program.  
(b) After the compiler compiles the program.  
(c) At the same time as the compiler compiles the program.  
(d) None

29. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int i=5,j=0;
    while(i-- || j++)
    {
        cout<<i<<" "<<j<<",";
    }
}
```

- (a) 5 1, 4 2, 3 3, 2 4, 1 5,                      (b) 4 0, 3 0, 2 0, 1 0, 0 0.  
(c) 4 1, 3 2, 2 3, 1 4, 0 5,                      (d) None

30. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a;
    bool b;
    a = 12 > 100;
    b = 12 >= 100;
    cout<<a<<" "<<b<<endl;
}
```

- (a) Error                      (b) 0 false                      (c) 0 1                      (d) 0 0



**Question Bank – Advanced C++ & Data Structures (Solved)**

31. What is the output of the following code?

```
#include<iostream.h>
int a = 1;
void main()
{
    int a = 100;
    {
        int a = 200;
        {
            int a = 300;
            cout<<a<<" ";
        }
        cout<<a<<" ";
    }
    cout<<a<<" ";
    cout<<::a<<" ";
}
```

- (a) Error (b) 100, 200, 300, 100,  
 (c) 300, 200, 100, garbage, (d) 300, 200, 100, 1,

32. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int x=10;
    (x<0)?(int a =100):(int a =1000);
    cout<<a;
}
```

- (a) Error (b) 1000 (c) 100 (d) None

33. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int a = 0;
    cout<<(a = 10/a);
}
```

- (a) 0 (b) 1 (c) Compile Time error (d) Runtime Error

34. What is the output of the following code?

```
#include<iostream.h>
void main()
{
    int x=0;
    while(x++<5)
    {
        static x;
    }
}
```

**Question Bank – Advanced C++ & Data Structures (Solved)**

```

        x+=2;
        cout<<x<<" ";
    }
}

```

- (a) 1 2 3 4 5                      (b) 2 4 6 8 10                      (c) Compile Time error                      (d) Runtime Error

35. What is the output of the following code?

```

#include<iostream.h>
void main()
{
    char str1[]="India", str2[]="India";
    if(str1==str2)
        cout<<"Both the string are same";
    else
        cout<<"Both the string are not same";
}

```

- (a) Both the string are same                      (b) Both the string are not same  
(c) Compile Time error                      (d) Runtime Error

36. What is the output of the following code if user enters "This is a test"?

```

#include<iostream.h>
#include<string.h>
void main()
{
    char str[8];
    cin>>str;
    cout<<str;
}

```

- (a) This is a test                      (b) This is a                      (c) This                      (d) Error

37. What is the output of the following code?

```

#include<iostream.h>
void main()
{
    int arr[] = {10,20,30,40,50};
    int *ptr = arr;
    cout<< *ptr++<<" "<<*ptr;
}

```

- (a) 10 20                      (b) 10 10                      (c) 20 20                      (d) 20 10

38. What is the output of the following code?

```

#include<iostream.h>
void main()
{
    int arr[] = {10,20,30,40,50};
    int x,*ptr1 = arr, *ptr2=&arr[3];
    x = ptr2 - ptr1;
}

```

**Question Bank – Advanced C++ & Data Structures (Solved)**

```

    cout<<x;
}

```

(a) 6                      (b) 3                      (c) Compile Time error                      (d) Runtime Error

39. Which of the following statement is false about pointers?  
 (a) The ++ and -- operators may be used with pointer variables  
 (b) An integer may be added and subtracted from a pointer variable  
(c) A pointer may be added to another pointer.  
 (d) A pointer may be subtracted from another pointer.

40. A null pointer is a pointer that contains  
(a) the address 0                      (b) the address that points to 0  
 (c) the address that points to '\0'                      (d) the address that points to -1

41. What is the output of the following code?  

```

#include<iostream.h>
void main()
{
    int arr[][3]={0,11,22,33,44,55};
    int *a = &arr[0][0];
    cout<<arr[1][2]<<" "<<*(a+3);
}

```

 (a) 55 33                      (b) 33 55                      (c) Compile Time error                      (d) Runtime Error

42. What is the output of the following code?  

```

#include<iostream.h>
void main()
{
    int arr[2][3][2]={{ {2,4},{7,8},{3,4},}, { {2,2},{2,3},{3,4}, }};
    cout<<(*(*arr+1)+2)+0)+7;
}

```

 (a) 10                      (b) 3                      (c) garbage value                      (d) Error

43. What is the output of the following code?  

```

#include<iostream.h>
void main()
{
    int arr[2][3][2]={{ {2,4},{7,8},{3,4},}, { {2,2},{2,3},{3,4}, }};
    cout<<**(*arr+1)+2+7;
}

```

 (a) 16                      (b) 7  
 (c) 11                      (d) Error

44. The design of classes in a way that hides the details of implementation from the user is known as:  
 (a) Encapsulation                      (b) Information Hiding  
 (c) Data abstraction                      (d) All of the above

*Question Bank – Advanced C++ & Data Structures (Solved)*

45. Which of the following keywords do you think can be used when declaring static members in a class?

- (i) Public
- (ii) Private
- (iii) Protected

- (a) i, ii and iii.
- (b) i and ii.
- (c) Only i.
- (d) i and iii.

46. I want a nonmember function to have access to the private members of a class. The class must declare that function:

- (a) friend
- (b) inline
- (c) static
- (d) virtual

47. The ability to reuse objects already defined, perhaps for a different purpose, with modification appropriate to the new purpose, is referred to as

- (a) Information hiding.
- (b) Inheritance.
- (c) Redefinition.
- (d) Overloading.

48. What do you think is the outcome of calling a redefined non-virtual function using a base-class pointer?

- (a) The appropriate redefined version of the function will be used.
- (b) The base-class version of the function will always be used.
- (c) The outcome is unpredictable.
- (d) A run-time error will occur.

49. A class member that is to be shared among all objects of a class is called

- (a) A const member
- (b) A reference parameter
- (c) A static member
- (d) A function member

50. What is a base class?

- (a) An abstract class that is at the top of the inheritance hierarchy.
- (b) A class with a pure virtual function in it.
- (c) A class that inherits from another class
- (d) A class that is inherited by another class, and thus is included in that class.

51. A variable that is declared protected:

- (a) Is visible only in the subclasses (and not in the class it is declared in).
- (b) Is visible only in the class it is declared in.
- (c) Is visible to all classes, but modifiable only in the class where it is declared.
- (d) Is visible in the class it is declared in, and all of its subclasses.

52. What is a destructor?

- (a) A function called when an instance of a class is initialized.
- (b) A function that is called when an instance of a class is deleted.
- (c) A special function to change the value of dynamically allocated memory.
- (d) A function that is called in order to change the value of a variable.

*Question Bank – Advanced C++ & Data Structures (Solved)*

53. In protected inheritance:

- (a) The public members of the base class become public.
- (b) The public members of the base class become protected.
- (c) The protected members of the base class become private.
- (d) The public members of the base class become inaccessible.

54. If a class declares a variable static, this means:

- (a) Each instance of a class will have its own copy of the variable.
- (b) Changing the variable in one instance will have no effect on other instances of the class.
- (c) There will be only one instance of the variable initialized for all classes.
- (d) Every instance of the class must consider the value of the static variable before initializing.

55. In case of a copy constructor, which of the following is true?

- (a) Used to instantiate an object from another existing object
- (b) To copy one object to another existing object.
- (c) Can be a substitute for a '=' operator.
- (d) All of the above.

56. A class declaring another class as a friend will:

- (a) Have wine and cheese with that other friend.
- (b) Allow that class to declare an instance of it in its list of private variables.
- (c) Allow the other class (the one declared as friend) to access to the declaring class's private variables
- (d) Allow the class declaring the other as a friend to access the declared class's private variables.

57. Which of the following can be virtual?

- (a) constructors
- (b) destructors
- (c) static functions
- (d) None of the above

58. Where is an exception generated?

- (a) In the catch block
- (b) In the throw clause
- (c) In the constructor of a class
- (d) Only when memory allocation fails.

59. Static member functions \_\_\_\_\_

- (a) can be used without an instantiation of an object.
- (b) can only access static data.
- (c) Both 1 and 2 are correct.
- (d) Neither 1 nor 2 are correct.

60. Which one is the simplest data structure:

- (a) Array
- (b) Linked List
- (c) Tree
- (d) Struct

*Question Bank – Advanced C++ & Data Structures (Solved)*

61. Which is not a data structure:
- (a) Array (b) Linked List  
(c) Binary (d) Struct
62. Which is not a sorting technique:
- (a) Radix sort (b) Merge sort  
(c) Poll sort (d) Quick sort
63. There are \_\_\_\_\_ types of searching technique:
- (a) 2 (b) 3  
(c) 1 (d) 4
64. For balancing a tree we use:
- (a) Left rotation (b) Right rotation  
(c) Both (d) None
65. In recursion which data structure is used:
- (a) Array (b) Linked List  
(c) Tree (d) Stack
66. If a tree has only one node than the tree may be a:
- (a) Binary tree (b) Tertiary tree  
(c) Not a tree (d) (a) & (b)
67. Stack is not used in
- (a) Recursion (b) Quick Sort  
(c) Postfix Notation (d) Simulation
68. Which one is not a type of a queue:
- (a) Deque (b) Circular Queue  
(c) Priority Queue (d) **Non-linear Queue**
69. Which one is not a linear data structure:
- (a) Array (b) Stack  
(c) Queue (d) Tree
70. Linked lists are not used in:
- (a) OS (b) Linker  
(c) Compiler (d) None
71. In double order traversal:
- (a) Every node is visited once  
(b) Every node is visited twice  
(c) Some nodes are visited twice  
(d) Only root node is visited twice
72. A Graph is a :
- (a) Linear Data Structure

*Question Bank – Advanced C++ & Data Structures (Solved)*

- (b) Non- Linear Data Structure
- (c) Not a Data Structure
- (d) Circular Data Structure

73. Searching Techniques in Graph are :

- (a) Breadth-last Search & Depth- first Search
- (b) Depth- last Search & Breadth-first Search
- (c) Breadth-first Search & Depth- first Search
- (d) Breadth-last Search & Depth- last Search

74. The organization and management of data structures are take place in:

- (a) Primary Memory
- (b) Secondary Memory
- (c) External Memory
- (d) Primary & Secondary Memory

75. The node of the circular doubly linked list must have:

- (a) One data and two address fields
- (b) One data and one address fields
- (c) Two data and two address fields
- (d) Two data and one address fields

76. In a complete binary tree of 'n' levels, there are:

- (a)  $2^n$  leaves and  $2^n-1$  non-leaf nodes
- (b)  $2n-1$  leaves and  $2n$  non-leaf nodes
- (c)  $n^2$  leaves and  $n^2-1$  non-leaf nodes
- (d)  $2^n -1$  leaves and  $2^n$  non-leaf nodes

77. In an expression binary tree, to obtain the postfix form of the expression we traverse in:

- (a) Pre order
- (b) Post order
- (c) In order
- (d) Pre and Post order both

78. In a binary tree, to delete a node that has two children, we require:

- (a) Post order successor
- (b) Pre order successor
- (c) In order ancestor
- (d) In order successor

79. What is the max number of edges an undirected graph with N nodes can have?

- (a) N
- (b)  $N^2$
- (c)  $2N$
- (d) none of the above

80. What type of data structure is used in a depth-first search?

- (a) Stack
- (b) queue
- (c) Arrays
- (d) All of the above

*Question Bank – Advanced C++ & Data Structures (Solved)*

81. One of the following algorithms is NOT an example of using the divide-and-conquer technique. Which one?

- (a) quicksort
- (b) mergesort
- (c) bubblesort
- (d) binary search

82. To apply the binary search algorithm, the data items should be represented as:

- (a) a binary tree
- (b) a list implemented as a linked-list
- (c) a list implemented as an array
- (d) an ordered list implemented as an array

83. Which of the following statements is true?

- (a) A graph can be drawn on paper in only one way.
- (b) Graph vertices may be linked in any manner.
- (c) A graph must have at least one vertex.
- (d) A graph must have at least one edge.

84. Suppose you have a directed graph representing all the flights that an airline flies. What algorithm might be used to find the best sequence of connections from one city to another?

- (a) Breadth first search.
- (b) Depth first search.
- (c) A cycle-finding algorithm.
- (d) A shortest-path algorithm.

85. What kind of list is best to answer questions such as "What is the item at position n?"

- (a) Lists implemented with an array.
- (b) Doubly-linked lists.
- (c) Singly-linked lists.
- (d) Doubly-linked or singly-linked lists are equally best

86. The operation for adding an entry to a stack is traditionally called:

- (a) add
- (b) append
- (c) insert
- (d) push

87. A \_\_\_\_\_ performs the copying for value returns as well as for value parameters.

- (a) Copy Constructor
- (b) Parameterize Constructor
- (c) Default Constructor
- (d) none

89. Consider A and B as two operands, and "+" as the operator, the presentation A + B is called:

- (a) prefix
- (b) postfix
- (c) infix
- (d) suffix

90. What makes a class abstract?

- (a) The class must not have method
- (b) The class must have a constructor that takes no arguments
- (c) The class must have a function definition equal to zero
- (d) The class may only exist during the planning phase







Question Bank – Advanced C++ & Data Structures (Solved)

```

        case 0:cout<<"zero ";
        case 1:cout<<"one ";continue;
        case 2:cout<<"two ";break;
    }
}
return 0;
}

```

(a) zero one one two (b) zero one two  
(c) zero two (d) none

96. What is the output of the following code?

```

#include<iostream.h>
class obj
{
public :
    obj(){cout<<"in ";}
    ~obj(){cout<<"out ";}
};
int main()
{
    obj A,B;
    {
        obj D;
    }
    obj E;
    return 0;
}

```

(a) in in in out in out out out (b) in in in in out out out out  
(c) in in in out in out out out (d) none

97. What is wrong in the following code?

```

#include<iostream.h>
class Base
{
public :
    Base(){};
    virtual ~Base(){};
};
class Derived : protected Base
{
public:
    virtual ~Derived(){};
};
int main()
{
    Base *pb = new Derived();
    return 0;
}

```

**Question Bank – Advanced C++ & Data Structures (Solved)**

- ```

}
(a) There is nothing wrong
(b) One cannot have a 'Base' pointer to 'Derived' since it is not derived publicly
(c) One need a derived class pointer to point to a derived class.
(d) One required to code a constructor for Derived.

```

98. What is the output of the following code?

```
#include<iostream.h>
```

```

class professor{public:professor(){cout<<"professor ";}};
class researcher{public: researcher(){cout<<"researcher ";}};
class teacher: public professor{public: teacher(){cout<<"teacher ";}};

```

```

class myprofessor: public teacher, public virtual researcher
{public:myprofessor(){cout<<"myprofessor ";}};

```

```

int main()
{
    myprofessor obj;
    return 0;
}

```

- (a) professor researcher teacher myprofessor  
**(b) researcher professor teacher myprofessor**  
(c) myprofessor teacher researcher professor  
(d) myprofessor researcher professor teacher

99. What is the output of the following code?

```
#include<iostream.h>
```

```

class Parent
{
public:
    Parent(){Status();}
    virtual ~Parent() { Status();}
    virtual void Status(){cout<<"Parent ";}
};
class Child: public Parent
{
public:
    Child(){Status();}
    virtual ~Child() { Status();}
    virtual void Status(){cout<<"Child ";}
};
void main()
{
    Child c;
}

```

**Question Bank – Advanced C++ & Data Structures (Solved)**

- (a) Parent Parent
- (b) Parent Child Child Parent
- (c) Child Parent Parent Child
- (d) Error

100. What is wrong in the following code?

```
#include<iostream.h>
class Base
{
public:
    virtual void Method()=0{n=1;}

private:
    int n;
};
class D1:Base {};
class D2:public D1
{
    int i;
    void Method(){i=2;}
};
int main()
{
    D2 test;
    return 0;
}
```

- (a) There is no error.
- (b) There is a syntax error in the declaration of “Method”.
- (c) Class D2 does not have access to “Method”.
- (d) Class D1 must define “Method”.

**Answers the followings:**

1. Write five main differences between C and C++.
2. Write three main differences between inline functions and macros with parameters.
3. What is name mangling in C++?
4. What are the differences between pointers and references?
5. What is there is no null reference?
6. Why we use operator overloading?
7. What is an automatic default constructor, and what does it do?
8. When is it appropriate to use a const reference parameter? Give a small example as part of your answer?
9. What are the differences between variable declaration and variable definition?
10. What is the significance of wchar\_t keyword?
11. What are the manipulators in C++?
12. Differences between new operator and malloc function.

**Question Bank – Advanced C++ & Data Structures (Solved)**

13. What are constant member functions?
14. In what situations a copy constructor is invoked.
15. What is initializer list?
16. What are template classes?
17. What are the operators that cannot be overloaded?
18. What are constant member functions?
19. When do you need deep copying?
20. Are templates memory efficient and why?
21. What is a conversion constructor?
22. What is the significant of this keyword?
23. When you use static data members. Given an example.
24. What is the use of mutable data members?
25. What are the smart pointers?
26. What is the difference between multiple inheritance and multilevel inheritance?
27. What is the difference between overloading and overriding?
28. What is the significant of 'vptr' in virtual functions.
29. How is the runtime polymorphism implemented in C++?
30. Write difference types of polymorphisms in C++.
31. What is the significant of pure virtual functions in C++.
32. Write different types of exceptions in C++.
33. What is the use of dynamic\_cast in C++.
34. Write five examples for STL sequence containers.
35. What do you mean by dangling pointer.
36. Write the steps to delete a node with two children in binary search tree.
37. What are iterators?
38. What are namespaces? What is the advantage of having a namespace? Give suitable examples
39. Why static member functions do not receive 'this' pointer?
40. What do you mean by abstraction?
41. What do you mean by encapsulation?
42. What are template classes?
43. Write the syntax for defining a function outside the class.
44. Write a operator + of the string class which append two strings. The string is stored within the class as data, which is char[50].
45. What is binding? Describe static and run time binding.
46. Write a function template that takes five parameters and returns the maximum of them. Also it is to be initialized using int and float.
47. Why do we need virtual destructors?
48. It is legal to return local variables from a function which returns by reference. State true/ false with justification.
49. Write a code to initialize the pointer to a data member of a class.
50. Static member functions do not receive 'this' pointer. State true/ false with justification.