C++ & DS (60 Minutes)

1. Which of the following options are true about inheritance?
   a) When deriving from a protected base class, public members become protected members of the derived class.
   b) When deriving from a protected base class, protected members become public members of the derived class.
   c) When deriving from a private base class, protected and public members become private members of the base class.
   d) When deriving from a public base class, the public members become public members of the base class.
   1. All of the above.
   2. a, c and d
   3. b, c and d
   4. c and d
2. Identify the true statements about virtual functions.
   i. A call to a virtual function using an object name and the dot member operator is resolved at run time.
   ii. Virtual functions are recognized by the inclusion of the keyword virtual in the function prototype.
   iii. Redefined virtual functions need to have the same number of parameters and the same return type.
   iv. Redefined virtual functions can be selected polymorphically at run time.
   1. i, ii and iv
   2. ii, iii and iv
   3. i and iv
   4. i, ii, iii and iv.
3. Which of the following can be virtual?
   1. constructors
   2. destructors
   3. static functions
   4. None of the above
4. Why is the extraction operator (>>) generally declared as a friend?
   1. To allow the class to be read in a specific format.
   2. To allow the operator to have access to private variables of the class.
   3. Since declaring the extraction operator part of the class will result in a compilation error.
   4. To allow the class to modify the stream.
5. Use the following code to answer the question

```
class Z {
    public:
        void def(char a);
        int ghi();
    private:
        char j;
        int k;
};
```

Which of the following is legal in a program that uses this class, after the following declaration:

```
Z x;
```

   1. x.ghi();
   2. x.j = 'd';
   3. Z.ghi();
   4. None of the above is legal.
6. Given the class definition:

```
class A
{
public:
    A(){}
    A(int x, char y):xx(x), yy(y) {}
// other members
private:
    int xx;
    char yy;
};
```

Which of the following initialization of this class is **not** legal (cause a compiler error)?
   1. A x(2, 'y');
   2. A x = A(2, 'A');
   3. A x(1);
   4. A x();
7. Consider the class inheritance:

```
class B
{
public:
    B();
    B(int nn);
    void f();
    void g();
private:
int n;
};
class D: public B
{
public:
    D(int nn, float dd);
    void h();
private:
    double d;
};
```

Which of the following functions can be invoked by an object of class D?
   1. f()
   2. g()
   3. h()
   4. All of the above
8. Consider the following class inheritance:

```
class B
{
public:
    B();
    B(int nn);
    void f();
    void g();
    virtual void h();
private:
    int n;
};
class D: public B
```

```
{
public:
    D(int nn, float dd);
    virtual void h();
private:
    double d;
};
```

After initializing an instance of B in the main program, and calling function h(), which classes' h will be called?

1. There will be a compile time error of ambiguity.
2. Class B's function h().
3. There will be a run-time error, and the program will crash.
4. None of the above are correct answers.

9. Which of the following is not a valid initialization of a template class, assuming the class is declared as follows:

```
template <class T>
class Pair {
```

1. Pair <int>
2. Pair <char>
3. Pair <abc> (assuming abc is a user defined class)
4. All of the above are valid initializations of a template class.

10. Suppose we have the class definitions, where the exception specification is as listed below:

```
class B
 {
 public:
     virtual void f() throw(int, double);
 };
 class D : public B
 {
         public:
             virtual void f() /*The exception
 specification you choose from the list goes here*/
 ;
 };
```

Which of these exception specifications is correct?

1. No exception specification is necessary.
2. throw (int, double);
3. throw (int, double, string);
4. throw (string);

11. Given the class declaration:

```
class D : public class B {/*...*/};
```

which of the following is true?

1. Public members of B become public members of D.
2. Private members of D become public members of B.
3. Protected members of B become public members of D.
4. Private members of B become public members of D.

12. Consider the following class inheritance:

```
class B
{
public:
    B();
    B(int nn);
    void f();
    void g();
    virtual void h();
private:
    int n;
};
class D: public B
{
public:
    D(int nn, float dd);
    void g();
    virtual void h();
private:
    double d;
};
```

When initializing an instance of B in the main program, and calling function g(), which classes's g() will be called?

1. There will be a compile time error of ambiguity.
2. Class B's function g().
3. There will be a run-time error, and the program will crash.
4. Class D's function g().

13. How does a class refer to itself?

1. By passing itself to a constructor with itself as the parameter
2. There is no way for a class to refer to itself
3. By pointing to another class just like this one.
4. By using the this pointer

14. VTABLE contains

1. addresses of virtual functions
2. addresses of virtual pointers
3. address of virtual table
4. None of the above

15. What is upcasting

1. storing the address of VTABLE in VPTR
2. storing the address of virtual functions in VPTR
3. storing the address of base class object in base class pointer
4. storing the address of derived class object in the base class pointer

16. Which of the following is not required in a class that contains dynamic allocation?

1. The copy constructor
2. A constructor that copies variables into private variables.
3. Destructor
4. All of the above are required

17. It is legal to return local variables from a function, which returns by reference.

1. True
2. False

18. In C++ one can define a function within another function.

1. True
2. False

19. In c++ an identifier can begin with a $ sign.

1. True
2. False

20. There can be a null reference.

1. True
2. False

21. Linked list are not superior to STL vectors.

1. True
2. False

22. Deleting a node in a linked list is a simple matter of using the delete operator to free the node's memory.
    1. True
    2. False
23. 'ios' stream is derived from iostream.
    1. True
    2. False
24. 'eof()' function returns zero value if the eofbit is set.
    1. True
    2. False
25. 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;
}
```
    1. Error
    2. 0 false
    3. 0 1
    4. 0 0
26. 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<<",";
}
```
    1. Error
    2. 100, 200, 300, 100,
    3. 300, 200, 100, garbage,
    4. 300, 200, 100, 1,
27. 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;
}
```
    1. Error
    2. 1000
    3. 100
    4. None
28. What is the output of the following code?
```
#include<iostream.h>
void main()
{
  int a = 0;
  cout<<(a = 10/a);
}
```
    1. 0
    2. 1
    3. Compile Time error
    4. Runtime Error
29. What is the output of the following code?
```
#include<iostream.h>
void main()
{
  int x=0;
  while(x++<5)
  {
       static x;
       x+=2;
       cout<<x<<" ";
  }
}
```
    1. 1 2 3 4 5
    2. 2 4 6 8 10
    3. Compile Time error
    4. Runtime Error
30. 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";
}
```
    1. Both the string are same
    2. Both the string are not same
    3. Compile Time error
    4. Runtime Error
31. 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;
}
```
    1. This is a test
    2. This is a
    3. This
    4. Error
32. 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;
}
```
    1. 10 20
    2. 10 10
    3. 20 20
    4. 20 10

33. 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;
  cout<<x;
}
```

1. 6
2. 3
3. Compile Time error
4. Runtime Error

34. Which of the following statement is false about pointers?
1. The ++ and -- operators may be used with pointer variables
2. An integer may be added and subtracted from a pointer variable
3. A pointer may be added to another pointer.
4. A pointer may be subtracted from another pointer.

35. A null pointer is a pointer that contains
1. the address 0
2. the address that points to 0
3. the address that points to '\0'
4. the address that points to –1

36. Namespace definition can only appear at
1. global scope
2. local scope
3. both local scope and global scope
4. None of the above

37. RTTI is used to find out
1. The address of class
2. The address of static member function
3. The exact type of object using a pointer or reference to the base class
4. The address of virtual function

38. The advantage of link list over array is
1. Link list can grow and shrink in size during life time
2. Less space is required for storing elements
3. Both 1 and 2 are correct
4. None of the above

39. Which one of the following algorithm is NOT an example of divide and conquer technique
1. Quick Sort
2. Merge Sort
3. Bubble Sort
4. Binary Search

40. Which one supports unknown data types in a single framework?
1. Inheritance
2. Virtual functions
3. Templates
4. Abstract Base Class

41. Which of the following is false about struct and class in C++?
1. The members of a struct are public by default, while in class, they are private by default
2. Struct and class are otherwise functionally equivalent
3. A class supports all the access specifiers like private, protected and public
4. A struct cannot have protected access specifier

42. Protected keyword is frequently used
1. For function overloading
2. For protecting data
3. For inheritance
4. For security purpose

43. The inorder traversal of some binary tree produces the sequence DBEAFC, and the postorder traversal of the same tree produced the sequence DEBFCA. Which of the following is a correct preorder traversal sequence?
1. DBAECF
2. ABEDFC
3. ABDECF
4. None of the above

44. How many cycles should be contained in a tree?
1. 0
2. at least 1
3. any number
4. None of the above

45. If graph G has no edges then corresponding adjacency matrix is
1. unit matrix
2. zero matrix
3. matrix with all 1's
4. None of the above

46. What is not true for linear collision processing?
1. It is easier to program
2. It may include more collision
3. It requires space for links
4. All are true

47. In an adjacency matrix parallel edges are given by
1. Similar columns
2. Similar rows
3. Not representable
4. None of the above

48. The element at the root of heap is
1. largest
2. smallest
3. depending on type of heap it may be smallest or largest
4. None of the above

49. Which keyword is used to decide on the choice of function or method at runtime?
1. abstract
2. virtual
3. protected
4. static

50. Which of the following is a correct statement?
1. Abstract class object can be created
2. Pointer to abstract class can be created
3. Reference to abstract class can be created
4. Both 2 and 3