Question asked in Interview

KPIT

1st round

1.

```
class shape
{
private:
   int x;
public:
   A(int v):x(v){}//paramertize constructor
   A(A &obj)       //copy constructor
   {
      x = obj.x;
   }
   virtual void cal()=0;
   void print(){
   std::cout<<"In the shape";
   }
};

template<class T>
class retangle: public shape
{
private:
   T lenght,breadth,result;
public:
   retangle():length(0),breadth(0){}
   retangle(T length_parameter, T breadth_parameter):
      length(length_parameter),breadth(breadth_parameter){}
   void cal(){
      result = lenght * breadth;
   }
   void print()
   {
      std::cout<<"Inside the rectangle";
      shape::print();
   }
};
```

```
int main()
{
//    A a(5);
//    A b(a);
//    recatngle *object_rectabgle = new recatngle(3,4);
 //   std::shared_prt<rectagle> sahre_rectangle_pointer( new recatngle(3,4);)

//    auto shared_rectangle = std::make_shared<rectangle>(23,45);
    retangle <shared_rectangle> object_rectangle(5,7);
    shape *shape_pointer = &object_rectangle;
    shape_pointer->cal();
    shape_pointer->print();
}
```

Put shared_pointer types into templates of an object in main().

2. Difference between L value reference and R value reference?

Harman

1. Argument of function are pointer to float and pointer to pointer to char and return type is pointer to pointer to int.
    a. Int** fun(float *, char**)
    b. Int * fun(float** , char*)
2. When inline function is explaned?
    a. Runtime
    b. Complietime
3. What is output

```
Int main()
{
Int a=5;
Auto fun= [=]()
{
A = 10;
}
Cout<<"a: "<<fun();
Return 0;
}
```

A. A: 5

B. A: 10
C. ERROR

4. What is output

```
Int main()
{
Bool a=true;
Bool b = false;
Int x=10;
Int y=5;
Int result = ((x|y) + (a+b));
Cout<<result;
Retrun 0;
}
```

   a. 12
   b. 0
   c. 2
   d. 16

5. Tell error in code

```
#include<iostream>       // 1
Using namespace std;
Int main()
{
Cout<<"Hello world";
Retrun 0;
}
```

```
#include <iostream>      //2

Int main()
{
Std::cout<<"Hello World";
Retrun 0;
}
```

   a. Error in 1
   b. Error in 2
   c. Neither 1 nor 2

6. What is output

```
Int main()
{
Int cin;
Cout<<"Enter value: ";
Cin>>cin;

Cout<<"Value: "<<cin;
Return 0;
}
```

7. What is output

```
Int main()
{
Char *ch =”hello\0world” ;
Cout<<ch;
Retrun 0;
}
```

   a. Hello
   b. World
   c. Hello world
   d. None of these

8. What is its output

```
Int main()
{
Std::string str = “Hello.”;
Str.back(!);
Cout<<str;
Return 0;
}
```

   a. Hello.!
   b. Hello!.
   c. Hello
   d. None of these

9. Which concept allow to reusability of function?
   a. Inheritance
   b. Polymorphism
   c. Encapsulation

10. Which is correct multiple inheritance
   a. X,y->z
   b. x->y->z
   c. x->y,x->z
   d. none of these
11. What is the use of virtual inheritance

    To solve the problem of getting properties of parent 2 times.

12. Class something

```
{
        Public:
        Something(){ cout<<”Constructor: “;}
        ~something(){ cout<<”Destructor: “;}
};
Int main()
{
```

```
Something *obj = new something[5];
Delete [] obj;
Retrun 0;
}
```

a. Constructor call 1 times & destructor call 5 times.
b. Constructor call 5 times & destructor call 5 times.
c. Constructor never calls.

Program:

1. Write a program for virtual function with the help of smart pointer.
2. Write a program for multithread with the help of lock.
3. Write a program for singletone or factory design pattern
4. Write a program for multiple memory management.
5. Write a program for thread using lambda function.