Sample OOP Lab Exam Paper

The lab exam will contain four parts that are as follows:

1. MCQ's
2. Output
3. Missing code
4. Scenario based question

Part 1. MCQ's

1.Polymorphism is achieved through:

a. destructor

b. constructor

c. virtual function

d. overloading operator

2. Which of the following stream class is used to perform both input and output file operation:

a. ofstream

b. ifstream

c. fstream

d. iostream

3.a++ statement, is equivalent to:

a. a=a+a

b. a=a+1

c.a=++a+1

d. both b and c

4.Which logical operator is unary operator?

a.//

b.!

c.&&

d.both a and b

5.The keyword 'friend' is used to define a:

a.Friend class

b.Friend constructor

c.Friend function

d.Both a and

6. Exposing only necessary information to users is known as

a. Data hiding

b. Encapsulation

c. Hiding complexity

d. Abstraction

7. Which of the following concepts is used to implement late binding?

a. Static function

b. Virtual function

c. Const function

d. Operator function

8. Which of the following operators cannot be overloaded ?

a. Static function

b. Virtual function

c. Const function

d. Operator function

9. Which of the following access specifier is used as a default in a class definition?

a. Private

b. Public

c. Friend

d. Protected

10. IS A relationship in C++ is

a. Inheritance

b. Encapsulation

c. Composition

d. None of the above

Part 2. Output

1.What is the output of this program?

```
#include <iostream>
using namespace std;
class team
{
   public : int member;
   void LFC()
   {
            cout<<"Its base class";
   };
};
class course:public team
{
   public :
   void LFC()
   {
     cout<<"Its derived class";
   }
};

int main()
{
team t; course c;
t.LFC();
c.LFC();
}
```

2. What is the output of this program?

```
#include <iostream>
using namespace std;
class course
{
  char name[10];
  public : void LFC()
  {
    cout<<"Its course system";
  }
```

```
};
class team : public course
{
   public: void LFC()
   {
     cout<<"Its team course system";
   }
};
int main()
{
     team t;
     t.LFC();
}
```

3. What is the output of this program?

```
#include<iostream>
using namespace std;
class P {
public:
  void print()
  { cout <<" Inside P::"; }
};

class Q : public P {
public:
  void print()
  { cout <<" Inside Q"; }
};

class R: public Q {
};

int main(void)
{
 R r;

 r.print();
 return 0;
}
```

4. What is the output of this program?

```
#include<iostream>
using namespace std;
class Shape {
private:
   int x;
   int y;
public:
   Shape(int i, int j);  // Constructor
};

Shape::Shape(int i = 0, int j = 0)  {
   x = i;
   y = j;
   cout << "Constructor called";
}

int main()
{
  Shape s1, *s2;
  return 0;
}
```

Part 3. Missing Code

1.Fill in the missing code where needed.

```
#include<iostream>
using namespace std;

class NUM
{
   private:
      int n;

   public:
      //function to get number
      void getNum(int x)
      {
         n=x;
      }
      //function to display number
      void dispNum(void)
      {
         cout << "value of n is: " << n;
      }
      //unary ++ operator overloading
```

a. Create a ++ operator overloading function that increments the value of n and does not return anything.

```
    //unary -- operator overloading
```
b. Create a -- operator overloading function that decrements the value of n and does not return anything

```
int main()
{
   NUM num;
```
c. Call the function and initialize the variable to get the output as shown in the output.

```
   ++num;
   cout << "After increment - ";
   num.dispNum();
   cout << endl;

   --num;
   cout << "After decrement - ";
   num.dispNum();
   cout << endl;
   return 0;
}
```
Output

```
   After increment - value of n is: 11
   After decrement - value of n is: 10
```

2. Fill in the missing code where needed.

```
#include <iostream>
#include <fstream>

using namespace std;

//class student to read and write student details
class student
{
   private:
      char name[30];
      int age;
   public:
      void getData(void)
      { cout<<"Enter name:"; cin.getline(name,30);
        cout<<"Enter age:"; cin>>age;
      }

      void showData(void)
      {
      cout<<"Name:"<<name<<",Age:"<<age<<endl;
      }
```

```
};

int main()
{
   student s;

   a.  Create an object for ofstream named “file”.

   b.Open file in write mode

   if(!file)
   {
    cout<<"Error in creating file.."<<endl;
    return 0;
   }
   cout<<"\nFile created successfully."<<endl;

   c. Write into file

   file.close();   //close the file
   cout<<"\nFile saved and closed succesfully."<<endl;

   //re open file in input mode and read data
   //open file1
   d.  Create an object for ifstream named “file1”.

   e.  Open file in read mode
   if(!file1){
      cout<<"Error in opening file..";
      return 0;
   }
   //read data from file
   file1.read((char*)&s,sizeof(s));

   //display data on monitor
   s.showData();
   //close the file
   file1.close();

   return 0;
}
```

Output

```
File created successfully.
Enter name:Mike
Enter age:21

File saved and closed succesfully.
Name:Mike,Age:21
```

Part 4. Scenarios

1. A school is creating new records that will be stored in files.The school wants to maintain separate files for each student that takes admission for the new school session. Write a program that will create a file and then write student's data such as name, age, address, guardian's phone number into the file. After writing into file, the file will be closed and open it in read mode to read all written text.

2. Student's are learning the concept of hierarchical inheritance. As a task, the teacher has them to write a program that uses the concept of hierarchical inheritance to calculate the square and the cube of a number. The program should contain a base class Number and two derived classes Cube and Square.

Note:

- The derived class should have a private member "num". The class contains two functions get and return that takes the input and returns the value.
- The Square and Cube class both have a get Function that returns the area for both, repectively.
- In the main, display the results for both.