SRM INSTITUTE OF SCIENCE &TECHNOLOGY, RAMAPURAM CAMPUS, DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CONTINUOUS LEARNING ASSESSMENT-1

18CSC202J– Object Oriented Design and Programming

ANSWER KEY SET-C

Part-A (5*1=5 Marks)

1.Which Feature of OOP illustrates code reusability?

a) Polymorphism

b) *Abstraction*

c) *Encapsulation*

d) *Inheritance*

Answer: D

2. Which among the following code is correct?

```
class student
{
   private: student()
   {
   }
   public : student( int x)
   {
           marks =x;
   }
};
```

a) The object can never be created

b) The object can be created without parameters

c) Only the object with only 1 parameter can be created

d) Only the object with some parameters can be created

Answer: C

3. Which of the following two features match with each other?

a) Inheritance and Encapsulation

b) Encapsulation and Polymorphism

c) Encapsulation and Abstraction

d) Abstraction and Polymorphism

Answer: C

4. Among the following UML diagrams, which one has Static structure?

a) Object

b) Use case

c) State chart

d) Activity

Answer: A or B

5. Which among the following is correct?

a) Private specifier must be used before public specifier

b) Private specifier must be used before protected specifier

c) Private specifier must be used first

d) Private specifier can be used anywhere in class

Answer: D

Part – B (2 X 4 = 8 Marks)

Answer any 2 Questions

6. **What is an Inline function? Explain with example program.**

C++ provides an inline functions to reduce the function call overhead. Inline function is a function that is expanded in line when it is called. When the inline function is called whole code of the inline function gets inserted or substituted at the point of inline function call. This substitution is performed by the C++ compiler at compile time. Inline function may increase efficiency if it is small.

```
Ex: using namespace std:
inline int cube(int s)
{
   return s*s*s:
}
int main()
{
   cout << "The cube of 3 is: " << cube(3) << "\n";
   return 0:
} //Output: The cube of 3 is: 27
```

7. **Explain the access specifies used in C++ programming with example?**

- public - members are accessible from outside the class
- private - members cannot be accessed (or viewed) from outside the class
- protected - members cannot be accessed from outside the class. however. they can be accessed in inherited classes.

8. **What are the components and the relationship used in UML class diagram?**

An association, dependency, generalization, and realization relationships are defined by UML. Composition relationship can also be used to represent that object can be a part of only one composite at a time

Part – C (1 X 12 = 12 marks)

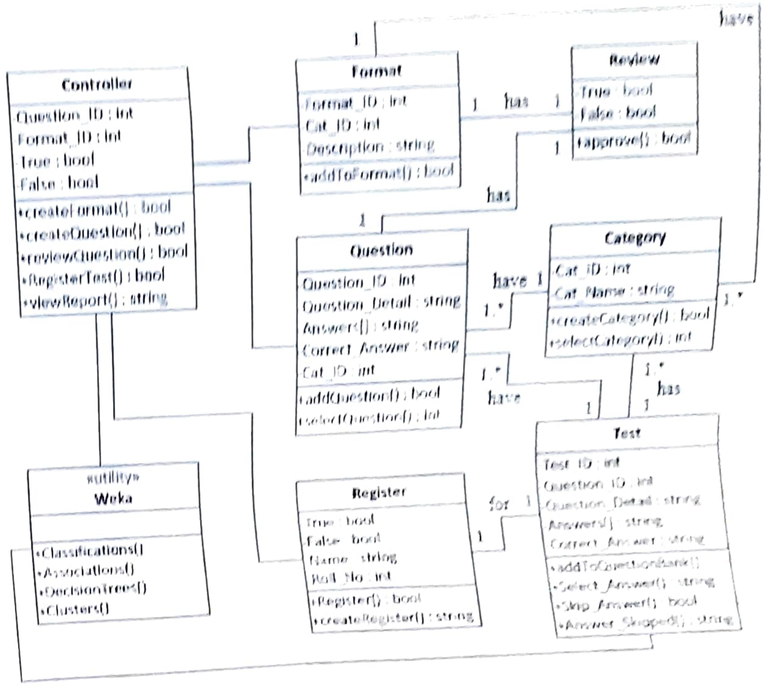
9. **a. Create the class Student and make use of the functions read(), sum(), and print(). To find the sum and average of 5 subjects.**

```
#include <iostream>
using namespace std;
class Student
{
public:
          int Read(int a, int b, int c, int d, int e);
          int Sum(void);
    void Print(int S);
private:
          int a,b,c,d,e;
          int Sum;
          float Average;
};
int Student::Read(int aa, int bb, int cc, int dd, int ee)
{
   a=aa;
   b=bb;
   c=cc;
   d=dd;
   e=ee;
}
int Student::Sum(void)
{
          Sum=a+b+c+d+e;
          return Sum;
}
```

```
void Student::Print(int S)
{
    Sum=S;
     Average=S/5;
     cout<<"The average is"<<Average;
  }
 int main()
  {
          Student a;
           a.Read(98,98,98,98,58);
            cout << "Value of a is: " << a.Sum();
             a.Print(S);
              return 0;
     }
```

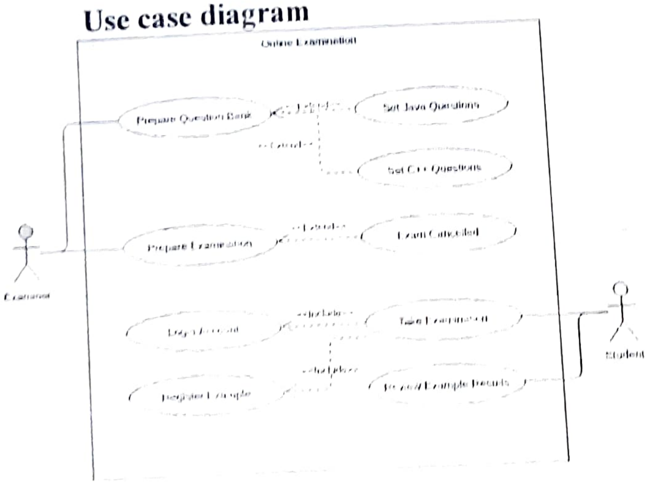
b.What is UML and the importance of UML? Draw the Use case and Class diagrams for the Online examination System.

UML:



Class diagram

Use case diagram



R. Sath 27/1/21
Course Coordinator
(R.SATHYA)

Hod/CSE