OBJECT ORIENTED PROGRAMMING

(CSEN 2103)

Time Allotted: 3 hrs   **Full Marks : 70**

Figures out of the right margin indicate full marks.

Candidates are required to answer Group A and any 5 (five) from Group B to E, taking at least one from each group.

Candidates are required to give answer in their own words as far as practicable.

Group – A

(Multiple Choice Type Questions)

1. Choose the correct alternative for the following: **10 × 1 = 10**

(i)   `int main(){ int x; void *ptr; int x=6; ptr=&x; }`
       Which of the following statements will print the value of x.
       (a) `cout << *ptr;`   (b) `cout << *((int *)ptr);`
       (c) `cout << *((int)ptr);`   (d) `cout << ((int *)ptr);`
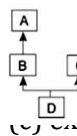
(ii)   ```
        #include<iostream>
        int& func(int a){return a*2;}
        int main(){func(2)=100;}
        ```
       The above code:
       (a) gives compiler error              (b) gives runtime error
       (c) compiles successfully             (d) none of the above.

(iii)   ```
        #include<iostream>
        int func(int a=5){return a*2;}
        int main(){std::cout<<func();}
        ```
       The above code:
       (a) gives compiler error   (b) gives runtime error

(iv)   Copy constructors are called when
       (a) we initialize one object from another of the same type
       (b) copy an object to return it from a function
       (c) we pass an object by value as an argument to a function
       (d) all of the above.

(vi)   Which of the following programs is correct?
       (a) ```
           class test {
           static int x=9;
           public: void print() { cout<<x<< endl; }
           }obj;
           int main() {
             obj.print();
           }
           ```
       (b) ```
           class test {
           static int x;
           public: void print() { cout<<x<< endl; }
           }obj;
           int test::x=10;
           int main() {     obj.print();     }
           ```
       (c) ```
           class test {
           static int x;
           public: void print() { cout<<x<< endl; }
           }obj;
           int main() {   obj.print();   }
           ```
       (d) ```
           class test {
           static int x;
           public: void print() { cout<<x<< endl; }
           }obj;
           int main()
             obj.print();
           ```

(vii)   A static member function can access only _____
       (a) private data members   (b) public data members
       (c) static data member   (d) other static member function.

(viii) Which of the following relationship is known as inheritance relation?
       (a) Is-a relationship   (b) Has-a relationship
       (c) Association relationship.   (d) Aggregation relationship.

(ix)   Which of the following cannot be instantiated?
       (a) Abstract class   (b) Concrete class
       (d) all of the above.   (d) Exception class.

(x)   What type of dispatch is important because this is how Java
       (a) Multiple   (b) Multipath
       (c) collection   (d) Multilevel polymorphism
       (c) exception handling   (d) encapsulation.

Group – B

2. (a) What is the difference between abstraction and encapsulation? Explain with real life example. Why is polymorphism useful?

   (b) When is the destructor of a class called? Explain with example in C++. State the advantage of call-by-reference over call-by-address.

   (3 + 2 + 2) + (1 + 2 + 2) = 12

3. (a) When is a copy constructor invoked? Why are objects passed by reference in a copy constructor?

   (b)
```cpp
#include <iostream>
using namespace std;
class B1 {
   public: B1(int a) {cout << "B1"<<endl;}
};
class B2 {
   public: B2(int b) {cout << "B2"<<endl;}
};
class C:public B2, public B1 {
public:
   C(int a):B1(a),B2(a){
   cout << "C"<<endl;
   }
};
int main() {
   C obj(5);
}
```
   What would be the output of the above program? Explain very briefly what decides the order in which the constructors are called.

   (c) Too frequent use of friend (function or class) defeats the purpose of encapsulation in object oriented programming. Why?

   (2 + 1) + (5 + 2) + 2 = 12

Group – C

4. (a) You have developed an Employee class with a public float pay (float hoursWorked) method and Manager class is derived from it and the method pay is overridden. You have created an employee object empl and a manager object mgr and access the pay method using following snippet:

   Employee *emplP;

```cpp
emplP = &empl;    emplP->pay(40.0);
   emplP = &mgr;   emplP->pay(40.0);
```

   In both the cases which pay method will be called and why?
   What is the rectification you suggest to call appropriate pay method in both the cases?
   What types of bindings are used in above snippet and your rectified solution?

   (b)
```cpp
try
   {
      if(x<100)   throw x;
      else     throw "Greater than 100";
   }
   catch (...) {cout << "Exception caught"; }
```
   What is catch(...) called? What is it used for?

   (c) What is rethrow? Give an example.

   (2 + 1 + 2) + 4 + 3 = 12

5. (a) Write a sample code in C++ to do matrix multiplication using operator overloading; do necessary error checking.
   Write another sample code in C++ to add two time objects using + operator overloading and also overload << and >> to get time input and show time output.

   (b) What are templates and why are they used?

   (c) What is the difference between class and function template?

   (3 + 3) + 3 + 3 = 12

Group – D

6. (a) State the difference between C++ and Java. Explain constructor overloading in Java with an example. How do protected and default access specifiers differ?

   (b) What are the static and non-static blocks in Java? What is dynamic method dispatch? Explain with example. Is inner class static?

   (1 + 3 + 1) + (2 + 4 + 1) = 12

7. (a) How do you instantiate static and non-static nested classes outside the outer class? How does garbage collection work in Java?

(b) Why do we use abstract class? Can abstract class be instantiated? Explain your answer with reason. When is subclass of an abstract class also an abstract class?

(3 + 3) + (2 + 2 + 2) = 12

Group – E

8. (a) Identify the problem in the following code pieces:-

(i) The following code is present in file testClass.java

```
public interface in1
{
  int a = 10;
  default void display() { System.out.println(a); }
}
class testClass implements in1
{
    public static void main (String[] args)
  {
    in1 t = new testClass();
    t.display();
  }
}
```

(ii)
```
interface in1
{
  int a = 10;
  default void display() { System.out.println(a); }
}
class testClass implements in1
{
    public static void main (String[] args)
  {
   a=20;
   in1 t = new testClass();
   t.display();
  }
}
```

(iii)
```
interface in1
{
  int a = 10;
  void display();
}
```

```
class testClass implements in1
{
    public static void main (String[] args)
  {
    in1 t = new testClass();
    t.display();
  }
}
```

(b) Differentiate between checked and unchecked exception in Java.

(c) What is the use of finally() block?

(2.5 + 2.5 +2.5) + 3 + 1.5 = 12

9. (a) What is the default name of the current executing thread?

(b) Write a program to change the name of the current executing thread to any name of your choice.

(c) What is the job of the repaint() method in applet and when should you invoke it?

(d) The method update() in applet should not be called until and unless it is absolutely necessary. Explain why.

2 + 4 + 3 + 3 = 12