THE MOMBASA POLYTECHNIC UNIVERSITY COLLEGE

(A Constituent College of JKUAT)

Faculty of Engineering & Technology

DEPARTMENT COMPUTER SCIENCE & INFORMATION TECHNOLOGY

DIPLOMA IN INFORMATION COMMUNICATION TECHNOLOGY – DICT 10M

EIT 2208: OBJECT ORIENTED PROGRAMMING II

END OF SEMESTER EXAMINATIONS

SERIES: DECEMBER 2011

TIME: 2 HOURS

Instructions to Candidates:
You should have the following for this examination
- *Answer Booklet*

This paper consist of **FIVE** questions in **TWO** sections **A** & **B**
Answer question **ONE (COMPULSORY)** and any other **TWO** questions
Maximum marks for each part of a question are as shown
This paper consists of **SIX** printed pages

a) Explain the following Object Orient Programming concepts:

       i)   Class
       ii)   Object
       iii)   Encapsulation
       iv)   Inheritance
       v)   Destructor
       vi)   Constructor
       vii)   Friend function
       viii) access specifier   **[16 Marks]**

b) Explain the use of scope operator in OOP   **[2 Marks]**

c)   Explain why the following code will not compile. Rewrite the code with corrections

```
# include <iostream.h>

class A

 {

  int  X1;

  int GetA();

  void SetA(int Num);

 }; // end of class definition

 int GetA()

 {

   return X1;

 }

void SetA(int Num)

 {

  X1 = Num;

 }

void main()

 {

  A a;

  a.SetA(5);

  int b = a.GetA();

  cout << b;

 } // end of main
```
                                                      [6 Marks]

d)
```
include <iostream.h>
class Example
  {
   public:
     Example();
  };
Example::Example()
  {
   cout << "An example has been made!";
  }

void main()
  {
   Example e;
  }
```

For the above code:

      i)   identify the constructor function

      ii)   state the output of the code if it's executed

      iii)   Write a destructor function for the above class   **[4 Marks]**

e) Explain the concept of overloading   **[2 Marks]**

SECTION B (ANSWER ANY TWO QUESTIONS)

QUESTION TWO (15 Marks)

a)   Given the code:

```
include <iostream.h>
class A
  {
   public:
     int a;
     int b;
     void Update(int a);
  };
void A::Update(int X)
  {
   a = X;
   b = 0;
  };

void main()
  {
   A a;
   a.Update(3);
   a.b = a.a;
   a.b++;
   int c = a.a;
  }
```

State the value of the following:

i)   Class member a at the end of the program

ii)   Class member b at the end of the program
iii)   main() variable c at the end of the program   **[6 Marks]**

b)   Explain 'access functions' as used in C++. Describe the purpose of the two access functions
                                                                            [5 Marks]
c)   Briefly explain the concept of Polymorphism as applied to OOP and state the three types of Polymorphism provided by C++   **[4 Marks]**

QUESTION THREE (15 Marks)

a)   Class Date
```
{
public:
    int m_nMonth; // public
    int m_nDay; // public
    int m_nYear; // public
};
void main()
{
    Date cDate;
    cDate.m_nMonth = 10; // okay because m_nMonth is public
    cDate.m_nDay = 14;  // okay because m_nDay is public
    cDate.m_nYear = 2020;  // okay because m_nYear is public
}
```
The above class definition will work well but it does not take care of encapsulation. Rewrite the class definition so that encapsulation is implemented   **[6 Marks**]

b)   Explain how encapsulation helps in coding   **[9 Marks]**

QUESTION FOUR (15 Marks)

a)   State the rules for naming destructors   **[3 Marks**]

b)   Describe the three types of access specifiers.   **[3 Marks]**

c)   The Describe the three different ways in which a child class can inherit from the parent class, stating access specifiers the child class inherits of the members of the parent   **[6 Marks]**

d)   Both the constructor and destructors have the same name as the class name itself. Yet it is possible for a class to have more than one constructor but only one destructor. Explain this.

                                                                            [3 Marks)

QUESTION FIVE (15 Marks)

a)   #include <iostream>

class Calculate

{

private:

   int a, b;

public:

```cpp
    void SetValue(int X1, int X2)

    {

        a = X1;

        b = X2;

    }

};
```

void main()

```cpp
{

    Calculate evaluate ;

    evaluate.SetValue();

     cout << "The product of the numbers is: "

     <<  compute(evaluate);
```
// Calling of Friend Function compute with object as argument.

```cpp
}
```

The above C++ code is supposed to multiply two integers, but has errors. The code is supposed to use a friend function called 'compute' which will add the numbers. However the code has errors. Rewrite the code so by including declaration of 'compute' function in the class definition, defining the compute function, The main() also contains errors. Rewrite the main() so that it can use the class Calculate and the friend function compute to accept two integers and out put their product.   **[6 Marks]**

b) Given the class definition below:

```cpp
class Person

{

private:

    int m_nID ;

    int m_nAge;

    bool m_bIsMale;

protected:

    char m_strName[25];

public:


    int GetID () { return m_nID; }
```

```cpp
      int GetAge() { return m_nAge; }

      bool IsMale() { return m_bIsMale; }

    void SetInfo(char *strName, int nID, int nAge, bool bisMale )

     {

        strncpy(m_strName, strName, 25);

        m_nID = nID;

        m_nAge = nAge;

        m_bIsMale = bisMale;

     }

     Person(int nID = 0,  int nAge = 0, bool bIsMale = false)

     {

        m_nID = nID;

        m_nAge = nAge;

        m_bIsMale = bIsMale;

     }

};
```

Write definition for class 'Employee' which is derived from the 'Person' class. The 'Employee' class has the following additional properties: basic salary, house allowance, hourly overtime, overtime hours worked and two additional members functions one TotalPay to calculate the total pay. Where TotalPay is given as follwows:

TotalPay = (basic salary + house allowance) + (hourly overtime * overtime hours)

and Disp member function to display employee name, employee ID, age and total pay.

[9 Marks]