KASNEB

CICT PART II SECTION 4

OBJECT ORIENTED PROGRAMMING

THURSDAY: 26 November 2015. **Time Allowed: 3 hours.**

Answer ALL questions. Marks allocated to each question are shown at the end of the question. Show ALL your workings.

ALL programs written should be in C++ object oriented programming language.

QUESTION ONE

(a) Identify four circumstances in which objects with unique names are unsuitable for use in an object oriented program. (4 marks)

(b) An extract of object-oriented code is given below:

```
# include <iostream.h>

    Class emp
    {
        int empno;
        char name [10]
        float salary;
     } el ;

Void main ( )
{
cout<< sizeof (emp) << endl << sizeof (e
}
```

Required:
Generate the output to the code above. (3 marks)

(c) A priority queue is an abstract data type that captures the idea of a container whose elements have "priorities" attached to them.

Required:
Describe four operations used in object oriented class template priority queue. (8 marks)

(d) Citing two examples, explain the use of arithmetic operators with pointers in object oriented programming. (5 marks)
(Total: 20 marks)

QUESTION TWO

(a) Summarise three problems associated with object oriented programs. (6 marks)

(b) (i) Explain the fundamental difference between passing an argument by value and passing an argument by reference. (4 marks)

(ii) A function named "swap" interchanges two integer values; x and y.

Required:
Write two versions of the above function to illustrate passing an argument by value and passing an argument by reference.

Ensure that in each case you include a main function that prompts for entry of the two integer values and calls the above functions. (10 marks)
(Total: 20 marks)

QUESTION THREE

(a) Rewrite the C++ code below correctly by debugging the error:

```
Int main ( )
{
   Const double  Pi;
     inst n;
     Pi  =  3.14;
     n   =  22;
}
```

(2 marks)

(b) Describe each of the following methods of handling exceptions:

(i) Try. (2 marks)

(ii) Catch. (2 marks)

(iii) Throw. (2 marks)

(c) Using illustrations, describe the following types of inheritance in the context of object oriented programming:

(i) Multiple inheritance. (2 marks)

(ii) Hierarchical inheritance. (2 marks)

(iii) Multilevel inheritance. (2 marks)

(iv) Hybrid inheritance. (2 marks)

(d) Use a suitable C++ code to explain the use of constant member functions. (4 marks)

(Total: 20 marks)

QUESTION FOUR

(a) Explain the effect of the following functions used during file operations:

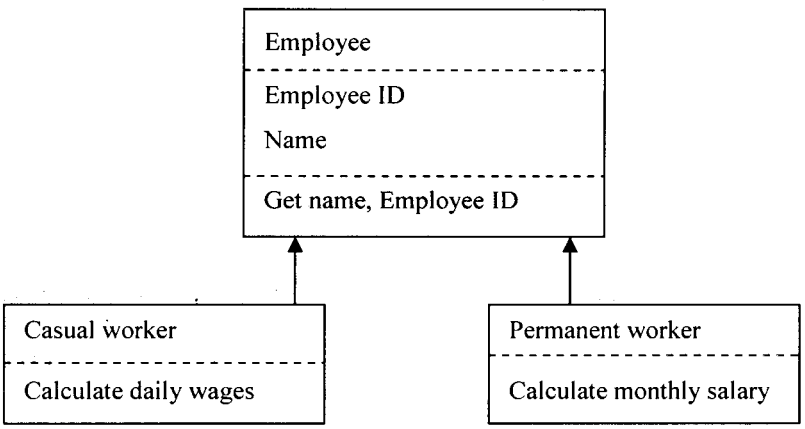
(i) Good ( ). (2 marks)

(ii) Eof ( ). (2 marks)

(iii) Fail ( ). (2 marks)

(iv) Bad ( ). (2 marks)

(b) The figure below shows a type of generalisation in object oriented programming:



Required:
Write an object-oriented program to implement the above generalisation.

The program should consist of the following:

(i) Three class declarations containing data members and function members where necessary. (8 marks)

(ii) A main program which instantiates the classes created in (b) (i) above and returns a worker's daily wages or monthly salary. (4 marks)

Assume that casual workers are paid on daily basis depending on the number of hours worked and permanent workers are paid a basic salary less statutory deductions at the end of the month.

(Total: 20 marks)

QUESTION FIVE

(a) Citing a suitable example in each case, describe the following message passing relationships in object-oriented programming:

(i) Association. (3 marks)

(ii) Dependency. (3 marks)

(b) An extract of object oriented code is given below:

```
# include <iostream.h>
  class ABC
      {
        private int i;
                float f;
        public  tp( ) ;
            {
              i  =   0;
              f  =   0.0;
            }
   }

 void main ( )
     {
         tp  t1;
      }
```

Required:

(i) Identify three errors in the code segment. (3 marks)

(ii) Rewrite the debugged code. (3 marks)

(c) Using C++ code, create a class 'Rider' with one attribute "rider_name". The class should consist of a parameterised constructor to set the name when an instance of the class is created and a method to return the rider's name. (8 marks)

(Total: 20 marks)

.....