UTHM

Universiti Tun Hussein Onn Malaysia

UNIVERSITI TUN HUSSEIN ONN MALAYSIA

FINAL EXAMINATION
SEMESTER II
SESSION 2013/2014

COURSE NAME	:	OBJECT-ORIENTED PROGRAMMING
COURSE CODE	:	BIT 20603
PROGRAMME	:	2 BIT
EXAMINATION DATE	:	JUNE 2014
DURATION	:	2 HOURS AND 30 MINUTES
INSTRUCTION	:	A) ANSWER **ALL** QUESTIONS
		B) PLEASE WRITE YOUR ANSWERS IN THIS QUESTION BOOKLET.

THIS QUESTION PAPER CONSISTS OF **TEN (10)** PAGES

Q1   Figures **Q1(a)-Q1(d)** apply object-oriented programming to find areas of various shapes. Refer to figures **Q1(a)-Q1(d)** to answer **Q1(a)-Q1(d)**.

(a)   Describe the relationship between the classes. Support your answer with appropriate diagram(s).

(6 marks)

Answer:

(b)   Use an example from the figure(s) to explain polymorphism concept.

(3 marks)

Answer:

(c)   Declare method `showAnswer()` as virtual function.

(2 marks)

Answer:

(d)   Declare an array of pointer, named `shape_Ptr` that can store three elements. The array can be assigned address(es) of objects instantiated for any class in Figures **Q1(a)-Q1(c)**.

(2 marks)

Answer:

(e)   Write programming statements to assign the address of the objects instantiated in Figure **Q1(d)**. In your answer, use the objects instantiated in Figure **Q1(d)** to assign the:

•   first element of `shape_Ptr` as referencing pointer to the class in Figure **Q1(a)**

•   second element of `shape_Ptr` as referencing pointer to the class in Figure **Q1(b)**

•   third element of `shape_Ptr` as referencing pointer to the class in Figure **Q1(c)**.

(3 marks)

Answer:

(f)   Write a looping statement to invoke method `showAnswer()` using the elements of `shape_Ptr`.

(9 marks)

Answer:

Q2   Figure **Q2** shows employee information of a company in a file called `Employee.dat`. Each row of the data presents four information of an employee. The information are `IC number, name, department,` and `basic salary`.

(a)   Declare fields in a structure form to store each employee's information.

(5 marks)

Answer:

(b)   Declare a class named `File`. The class should contain the answer in **Q2(a)** as private data and a public method called `Get_Data()`. Note that `Get_Data()` is neither receiving nor returning any value.

(5 marks)

Answer:

(c)   Write the method called `Get_Data()` for the class named `File`. In the method, implement program segment to read the content of `Employee.dat` and display the information on output screen. Note that the answer in **Q2(a)** is applicable.

(15 marks)

Answer:

Q3 (a) Figures **Q3(a)** and **Q3(b)** contain programming statements with errors. Find and fix at least **FIVE (5)** errors in the statements. Write your answers in **Table 1**.

Answer:

Table 1: List of Errors

Error	Correction

(5 marks)

(b) Design a test plan that consists of any **FIVE (5)** test cases and their expected output for Figures **Q3(a)** and **Q3(b)**.

(10 marks)

Answer:

Q4   Questions **4(a)-4(d)** are based on Figures **Q4(a)** and **Q4(b)**.

(a)   Declare the class for Figure **Q4(a)**. Apply `struct` definition to store customer's information shown in Figure **Q4(a)**.

(5 marks)

Answer:

(b)   Implement method `Add(Customer newCust)` to add new object of a customer's information at the beginning of a linked list.

(15 marks)

Answer:

(c)   Implement method `DisplayList()` to display all of the information from the linked list.

(15 marks)

- END OF QUESTION –

FINAL EXAMINATION

SEMESTER/SESSION: SEM II/2013/2014
COURSE NAME: OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 2 BIT
COURSE CODE: BIT 20603

```cpp
//Filename : Shape.cpp
class Shape{
    private:int width, height;
    public: void setValue(int w, int h);
            int getWidth();
            int getHeight();
            void showAnswer(); };

void Shape::setValue(int w, int h){width = w;height = h; };
int Shape::getWidth() {return width;};
int Shape::getHeight() {return height;};
void Shape::showAnswer(){
    cout<<"\nThe values are: width="<<getWidth();
    cout<<" height="<<getHeight() };
```

FIGURE Q1(a)

```cpp
//Filename : Triangle.cpp
#include "Shape.cpp"

class Triangle: public Shape{
    public: void showAnswer();};

void Triangle::showAnswer{
cout<<"\nThe area is:"; cout<<((getWidth()*getHeight())/2); };
```

FIGURE Q1(b)

```cpp
//Filename : Rectangle.cpp
#include "Triangle.cpp"

class Rectangle: public Shape {
    public: void showAnswer() (); };

void Rectangle:: showAnswer(){
cout<<"The area is"<<getWidth()*getHeight();};
```

FIGURE Q1(c)

FINAL EXAMINATION

SEMESTER/SESSION: SEM II/2013/2014
COURSE NAME: OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 2 BIT
COURSE CODE: BIT 20603

```
//Filename: mainDr.cpp
#include "Rectangle.cpp"

int main() {
int x;
Shape shape;
shape.showAnswer();

Triangle triangle;
triangle.setValue(2,9);
triangle.showAnswer();

Rectangle rectangle;
rectangle.setValue(3,3);
rectangle. showAnswer() ();

return 0;};
```

FIGURE Q1(d)

ID Number	Name	Department	Basic Salary (RM)
810221115678	Upin	HR	1200
800705107896	Ipin	Store	900
820926048973	Bujal	Admin	2000
801011016751	Kuntum	Registrar	1500
801109012345	Mail	Maintenance	1000

FIGURE Q2

FINAL EXAMINATION

SEMESTER/SESSION: SEM II/2013/2014   PROGRAMME : 2 BIT

COURSE NAME: OBJECT-ORIENTED PROGRAMMING   COURSE CODE: BIT 20603

```
1    Filename : Multiply.cpp
2    This class is meant to find result of multiplication for  two
3    numbers
4    #include<iostream.h>
5    class Multiply {
6         private:
7               int x,y;
8         public:
9         int MultiplyNo();
10        void Result();};
11
12   void Multiply::MultiplyNo() {
13        cout<<"Please enter a number.";
14        cin>>x;
15        cout<<"Please enter a number.";
16        cin>>y;
17        test = xy;
18        return test;
19   }; //method MultiplyNo
20
21   int Multiply::Result() {
22   result = MultiplyNo();
23   cout<<"\n The result for the multiplication of x:" >> x;
24   cout<< " and y: " >> y;
25   cout<< " is " >>result;};//method Result
```

FIGURE Q3(a)

```
1    //Filename: MutiplyDr.cpp
2    //This program serves as driver (main)
3    #include <iostream.h>
4    int main(){
5         Multiply Multiply2No;
6         Multiplyof2No.Result();
7         return 0; };
```

FIGURE Q3(b)

FINAL EXAMINATION

SEMESTER/SESSION: SEM I/2013/2014
COURSE NAME.: OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 2 BIT
COURSE CODE: BIT 20603

Customer		
-	Name	
-	Id	
-	TotalPoints	
+	SetCust()	: void
+	GetCust()	: void

FIGURE Q4(a)

```
class CustList  {
     private:
          struct ListNode {
               Customer acust;
               ListNode *next;
          };
          ListNode *head;
     public:
          CustList();
          int IsEmpty();
          void Add(Customer newCust);
          void Remove();
          void DisplayList();
};

CustList::CustList() {
          head = NULL;   };

int CustList::IsEmpty() {
          if (head == NULL)
               return 0;
          else
               return 1;
}; // method IsEmpty
```

FIGURE Q4(b)