

CONFIDENTIAL



UNIVERSITI TUN HUSSEIN ONN MALAYSIA

**FINAL EXAMINATION
SEMESTER I
SESSION 2015/2016**

COURSE NAME : OBJECT-ORIENTED PROGRAMMING
COURSE CODE : BEC20702
**PROGRAMME : BACHELOR OF ELECTRONIC
ENGINEERING WITH HONOURS**
EXAMINATION DATE : DECEMBER 2015 / JANUARY 2016
DURATION : 2 HOURS
**INSTRUCTION : ANSWER ALL QUESTIONS.
WRITE ALL ANSWERS USING
BLUE/BLACK INK PEN. ANY
ANSWERS WRITTEN IN PENCIL WILL
NOT BE GRADED.**

THIS QUESTION PAPER CONSISTS OF TEN (10) PAGES

CONFIDENTIAL

INSTRUCTION: Write **all** answers **using** blue/black ink pen in the answer booklet. Any answers written using pencil will **not** be graded.

- Q1** (a) Refer to the C++ fragment code in **Figure Q1(a)**. Answer (i) and (ii).
- (i) Explain the flow control for exception handling of the fragment code. (8 marks)
 - (ii) Assume that the `IsValidError` function is called using the following statement.
`IsValidError (175);`
Therefore, analyse the fragment code to determine the output produced. (5 marks)
- (b) Answer (i) and (ii).
- (i) Write C++ statements based on the program description in **Figure Q1(b)**. (12 marks)
 - (ii) Rewrite the function definition of `SmallestObject` function using overloaded operator `<`. The task performed by overloaded operator `<` is same as the `SmallestObject` function. (5 marks)

INSTRUCTION: Write **all** answers **using** blue/black ink pen in the answer booklet. Any answers written using pencil will **not** be graded.

- Q2** (a) In answering this question, you are required to consider the fragment code in **Figure Q2(a)** to access the non-public member of the class. A non-member function of the *DateofYear* class named `equal` function performs the following task.

The `equal` function takes two arguments of the type of *DateofYear*. The function returns a `bool` value. The function returns `true` if the first argument and the second argument represent a date has same value. Otherwise, the function returns `false`.

Write its function definition using C++ programming language.

(10 marks)

- (b) A friend function is another method to allow the `equal` function to access all non-public members of the class. (Please keep in mind that the `equal` function is a non-member of the *DateofYear* class).

- (i) Explain how to implement this method in the declaration section of the class, the function definition of `equal` function and the function call, to call the friend function.

(5 marks)

- (ii) Without altering the fragment code in **Figure Q2(a)**, write the C++ statement to declare the `equal` function as friend function of the *DateofYear* class.

(2 marks)

- (iii) State which line of code to put the statement(s) of Q2(b)(ii). Provide your reason.

(3 marks)

- (c) Write C++ statements for Q2(c)(i) and Q2(c)(ii) based on the program description in **Figure Q2(c)**. Please take note that in answering this question, you are required to consider the fragment code in **Figure Q2(a)**, and implementation of friend function in Q2(b)(i) to Q2(b)(iii).

(10 marks)

- Q3** (a) Fill in the following blanks for (i) to (vi) with the correct answer. Write all answers in the answer booklet.

CONFIDENTIAL

BEC20702

INSTRUCTION: Write **all** answers **using** blue/black ink pen. Any answers written using pencil will **not** be graded.

- (i) Inheritance is the process that allows a creation of a new class from existing class. The existing class is called ____ class (derived/base) and the new class is referred to as the ____ class (derived/base).
(2 marks)
- (ii) The base class may be inherited through _____ (public/derived class's members), _____ (protected/base class's members) and private inheritance. The type of inheritance is specified by the _____ (member of class/access specifier).
(3 marks)
- (iii) Protected member may be _____ (accessed / inaccessible) by derived class. Private members are _____ (accessed / inaccessible) to derived class.
(2 marks)
- (iv) _____ (Member/Base class) access specification specifies how members of the base class are inherited by the derived class. _____ (Member/Base class) access specification specifies how class members may be accessed by code outside the class.
(2 marks)
- (v) The following C++ statement is given.

```
class Circuit : parallelCircuit
```

What is the name of base class? _____ (Circuit/parallelCircuit)
What is the name of derived class? _____ (Circuit/parallelCircuit)
What is the class access specification of the base class?
_____ (unknown/ private)
(3 marks)

- (vi) The meaning of the following C++ statement is as follows.

```
class FinalExam : public GradedActivity
```

The _____ (public/private) members of the *GradeActivity* class will become public members of the *FinalExam*. The private members of the *GradedActivity* class _____ (accessed/inaccessible)

CONFIDENTIAL

BEC20702

INSTRUCTION: Write **all** answers **using** blue/black ink pen. Any answers written using pencil will **not** be graded.

by code in *FinalExam* class. They can only be accessed by the member functions of the _____ (*GradedActivity/FinalExam*) class.
(3 marks)

- (b) Draw the UML diagram for the classes involved in **Figure Q3(b)**. In the diagram, show the relationship .
(15 marks)
- (c) Analyse the program in **Figure Q3(b)**. Then, list the Rectangle's class members.
(5 marks)
- (d) Without modifying the fragment code in **Figure Q3(b)**, write the C++ statement based on the program description for Q3(d)(i) and Q3(d)(ii) of in order to complete the program.
(5 marks)

- END OF QUESTIONS -

FINAL EXAMINATION

SEMESTER / SESSION : SEM I/2015/2016
COURSE : OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 3BEJ/4BEJ
COURSE CODE : BEC20702

/*Q1(a) fragment code.

Please note that the left column is representing line numbers that have been added for you to identify certain parts of the program.*/

```
1. void IsValError(int val){
2.     try
3.     {
4.         cout <<"Try block entered. \n";
5.         if (val>100)
6.             throw val;
7.         cout <<"This is the text after leaving try block. \n";
8.     }
9.     catch (int thrown_value)
10.    {
11.        cout <<"Error at val " << thrown_value <<".\n";
12.    }
13.    cout <<"This is the text after leaving catch block. \n";
14. }
```

FIGURE Q1(a)

FINAL EXAMINATION

SEMESTER / SESSION : SEM I/2015/2016
COURSE : OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 3BEJ/4BEJ
COURSE CODE : BEC20702

```
//Q1(b) program
1. | #include <iostream>
2. | using namespace std;
3. | class Circle
4. | {
5. | private:
6. |     float radius;
7. |     float diameter;
8. | public:
9. |     Circle(float, float);
10. |     void setRadius(float);
11. |     void setDiameter(float);
12. |     double SmallestObject(Circle, Circle);
13. | };
14. | Circle::Circle(float r=1.0, float d=1.0)
15. | {
16. |     radius=r;
17. |     diameter=d;
18. | }
19. | void Circle::setRadius(float r)
20. | {
21. |     radius=r;
22. | }
23. | void Circle::setDiameter(float d)
24. | {
25. |     diameter=d;
26. | }
```

/*Q1(b)(i)Function definition for function SmallestObject. The function takes two arguments of type Circle. The function compares between two objects of type Circle and returns a value of smallest diameter.

(12 marks)*/

/*Your C++ code for Q1(b)(i) is placed here.*/

FIGURE Q1(b)

FINAL EXAMINATION

SEMESTER / SESSION : SEM I/2015/2016
COURSE : OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 3BEJ/4BEJ
COURSE CODE : BEC20702

Q2(b) fragment code of class declaration and implementation

```
1. | class DateofYear
2. | {
3. |     public:
4. |         DateofYear ();
5. |         DateofYear (int, int);
6. |         void setMonth (int); //set the month of the year
7. |         void setDay (int); // set the day of the month
8. |
9. |     private:
10 |         int month;
11 |         int day;
12 | };
13 |
14 |
15 | DateofYear::DateofYear(){
16 |     day=1;
17 |     month=1;
18 | }
19 |
20 | DateofYear::DateofYear(int t_month, int t_day){
21 |     day=t_day;
22 |     month=t_month;
23 | }
24 |
25 | void DateofYear::setMonth(int t_month){
26 |     month=t_month;
27 | }
28 |
29 | void DateofYear::setDay (int t_day){
30 |     day=t_day;
31 | }
```

FIGURE Q2(a)

FINAL EXAMINATION

SEMESTER / SESSION : SEM I/2015/2016
COURSE : OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 3BEJ/4BEJ
COURSE CODE : BEC20702

//Q2(c) fragment code in client program

```
DateofYear today(12,3);  
DateofYear birthday(12,3);
```

/*Q2(c)(i) Call the mutator functions to modify the instance variable day and month in birthday to become 7 and 9, respectively.

(4 marks)*/

/*Your C++ code for Q2(c)(i) is placed here.*/

/*Q2(c)(ii) Call the equal function to compare and determine either the date is equal or not. If the date is equal, display "Today is your birthday." Otherwise, display "Today is someone else birthday.

(6 marks)*/

/*Your C++ code for Q2(c)(ii) is placed here.*/

FIGURE Q2(c)

FINAL EXAMINATION

SEMESTER / SESSION : SEM I/2015/2016
 COURSE : OBJECT-ORIENTED PROGRAMMING

PROGRAMME : 3BEJ/4BEJ
 COURSE CODE : BEC20702

```

1. | //Q3(b) program
2. | #include <iostream>
3. |
4. | class Shape{
5. | public:
6. |     void setWidth(int w){
7. |         width=w;
8. |     }
9. |     void setHeight(int h){
10. |         height=h;
11. |     }
12. | protected:
13. |     int width;
14. |     int height;
15. | };
16. |
17. | class Rectangle:public Shape{
18. |     _____: /*Q3(d)(i): Suggest a suitable access specifier
19. |                 and give a reason. (2 marks) */
20. |     int getArea(){
21. |         return(width*height);
22. |     }
23. | };
24. | void main(){
25. |     Rectangle Rect;
26. |
27. |     Rect.setWidth(5);
28. |     Rect.setHeight(7);
29. |
30. | /*Q3(d)(ii) Fill in the blanks with suitable C++ statements
31. | to display the area of the object.(3 marks)*/
32. | _____ <<"Area of rectangle is "<<_____<<endl;
33. | }

```

FIGURE Q3(b)