King Saud University

College of Computer and Information Sciences

Computer Science Department

Course Code:	CSC 111
Course Title:	Introduction to Programming
Semester:	Spring 2016
Exercises Cover Sheet:	**Mid 2 Exam - A**
Duration: 90 min	

Student Name:	
Student ID:	
Student Section No.	

Tick the Relevant	Computer Science B.Sc. Program ABET Student Outcomes	Question No. Relevant Is Hyperlinked	Covering %
√	a) Apply knowledge of computing and mathematics appropriate to the discipline;	1,2	25%
	b) Analyze a problem, and identify and define the computing requirements appropriate to its solution;		
	c) Design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;		
	d) Function effectively on teams to accomplish a common goal;		
	e) Understanding of professional, ethical, legal, security, and social issues and responsibilities;		
	f) Communicate effectively with a range of audiences;		
	g) Analyze the local and global impact of computing on individuals, organizations and society;		
	h) Recognition of the need for, and an ability to engage in, continuing professional development;		
√	i) Use current techniques, skills, and tools necessary for computing practices.	1,2	25%
	j) Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;		
	k) Apply design and development principles in the construction of software systems of varying complexity;		

Important Notes:

- Cheating is prohibited! Looking at your colleague's paper will get you an F in the course immediately!

- Turn OFF your Phone/s. If you take out your phone for ANY reason, you will get an F in the course immediately!

ملاحظات هامة

- الغش ممنوع! عند النظر إلى ورقة زميلك، سترسب في المادة مباشرة!

سببِ لأي للجوال إخراجك عند ***!***جوالاتك **/** جوالك أغلق
!مباشرة المادة في سترسب كان،

Question 1.A (1 Mark)

Put your answer of the question 1 (**multiple choice questions**) in the following table. ***Note: Student is allowed 2 wrong choices with no penalty. After that, for each wrong choice he loses half a point***:

Question	Answer
1	
2	

1- What is the output of the following program, if any?

```
public class StaticTest {
     private static int a;

// constructors, setter, getters ,…

     public int m1(int x){
          return (a + x);
     }
}
public class Test {
     public static void main(String[] args) {
          StaticTest st = new StaticTest();
          st.setA(10);
          System.out.print(st.m1(5));
     }
}
```

A. **10**
B. **15**
C. **The method m1 is correct, because the static method is using a static variable.**
D. **The method m1 is incorrect, because the non-static method is using a static variable.**
E. **The method m1 is incorrect, because the static method is using a local variable.**

2- What is the output of the following code segment if any?

```
int j;
for(int i = 1; i <2; i++)
{
     for (j = 0; j < (j+1); j++)
     {
          System.out.print(j);
     }
}
```

A. 0 1 2 **B. 0, 1 , 2, 3, 4** **C.0 2 4 6** **D. Infinit loop**

Question 1.B (3 Mark)

Suppose you have the following class:

```
class Q2 {
     private int a;
     private int b;
     public Q2() { a = 10; b = 20;}
     public Q2(int a, int b) { this.a = a; this.b = b;}
     public void setA(int a){this.a = a;}
     public void setB(int b){this.b = b;}
     public int getA(){return a;}
     public int getB(){return b;}

     public boolean m(int a, int b){
          a += 10;
          this.b = a * 2;
          return this.a > b ;
     }
}
```

What is the output of the main method in the following class `TestQ2`?

```
public class TestQ2 {
     public static void main(String args[]){
          Q2 obj1 = new Q2();
          Q2 obj2 = new Q2(1, 2);
          System.out.println("obj1.a = " + obj1.getA());
          System.out.println("obj2.b = " + obj2.getB());
          obj1.setA(5); obj1.setB(7);
          System.out.println("obj1.a = " + obj1.getA());
          System.out.println("obj1.b = " + obj1.getB());
          obj2 = obj1;
          obj2.setA(4); obj2.setB(6);
          System.out.println("obj1.a = " + obj1.getA());
          System.out.println("obj2.b = " + obj2.getB());
          if(!obj1.m(obj1.getB(), obj2.getA()))
               System.out.println("True");
          else
               System.out.println("False");
          System.out.println("obj2.a = " + obj2.getA());
          System.out.println("obj2.b = " + obj2.getB());
     }
}
```

____

____

Question 2 (6 Marks)

Given the following class:

Rocket
- x: int - y: int - z:int - size: String
+ Rocket() + Rocket(a: integer, b: integer, c:integer) + setRocket(a: integer, b: integer, c:integer): void + getX() : integer + gety(): integer + getz(): integer + getsize() : integer + moveRocket(dx : integer, dy : integer, dz : integer):void - changeSize() : void + display() : void + *main(args: String) : void*

x	**y**	**z**	**size**
<0	<0	=0	small
>0	<0	=0	medium
<0	>0	>0	Large
>0	>0	>0	Very Large
othervalue	othervalue	othervalue	Notmatching

Consider the class **`Rocket`** that defines a position with its 3D coordinates(x,y,z) and its size. The class Rocket has the following attributes:

x : represents the x axis value of its first coordinate
y : represents the y axis value of its first coordinate
z : represents the z axis value of its first coordinate
size : size of rocket

The class **`Rocket`** has the following methods:

- A constructor Rocket() initializes x,y,z with the value 0(zero) by calling next constructor.
- A constructor Rocket(int a, int b, int c): initializes x, y, z with the value a, b and c by calling setRocket.
- setRocket: stores the values of the input parameters in the attributes x, y and z and calls changeSize to change the size of the rocket based values of x, y and z and the rules in table above.
- Getter methods for all attributes.
- moveRocket: move the rocket from its actual position to a new position where dx, dy and dz represent the shift from the actual position to the new one.
- changesize: a private method that changes the size of the rocket according to its coordinates x, y and z and the rules in the table above.
- display: a method that displays the information of the rocket (its coordinates and size).

Note: setRocket and moveRocket changes the coordinates of the rocket. Any changes of the coordinates of the Rocket changes the size of the Rocket according to the defined table.

Complete the blank space.

```
public class Rocket {
private int x;
private int y;
private int z;
private String size;
public Rocket()
{
     //initialize the object to 0, 0, 0 using next constructor

     ------------------------------
}
public Rocket(int a, int b, int c)
{
     // use setRocket method to set x, y, z to values a, b, c

     ---------------------------------------;
}
public void setRocket(int a, int b, int c)
{

     ---------------------;//set x to a

     ---------------------;//set y to b

     ---------------------;//set z to z

     ---------------------;//set size using changeSize method
}
public ----------- getx()
{
     ---------- x;
}
public ----------- gety()
{
     ---------- y;
}
public ----------  getz()
{
     ---------- z;
}
public ------------- getSize()
{
     ---------- size;
}
// moves the rocket according to passed distances dx, dy, and dz of
type int
public void moveRocket(---------------------------)
{
     x += dx;
     y += dy;
     z += dz;

     --------------------------;
}
------- ------- changeSize()
{
----------------------------------------;

----------------------------------------;

----------------------------------------;

----------------------------------------;

----------------------------------------;

----------------------------------------;
}
public void display()
{
     System.out.println("Rocket at ("+x+" ,"+y+" ,"+z+") is
"+size);
}
public static void main(String[] args) {

          //create a rocket object with values -5, -2, 0

          ------------------------------------;

          // move the rocket by 7, 9, 10

          ------------------------------------;

          // set the rocket position to -5, -10, -20

          ------------------------------------;

          // print the rocket info

          ------------------------------------;
}// main
}// class
```

Result

Question No.	Relevant Student Outcome	SO is Covered by %	Full Mark	Student Mark	Assessor's Feedback
1	a	25			
2	i	25			
Totals		50%			

I certify that the work contained within this assignment is all my own work and referenced where required. Student Signature: Date:			Feedback Received: Student Signature: Date:	