جامعة بيرزيت

BIRZEIT UNIVERSITY

Computer Science Department

Advanced Programming (COMP231)

Midterm Exam 7/11/2017 *Time: 75 minutes*

Student Name_____KEY Key

Student Number_____

Please circle the name of your instructor *as well as your lab section*:

- Dr. Majdi Mafarja: (sec 1) (sec 2)
- Mr. Hafeth Barghouthi: (sec 3) (sec 4)
- Dr. Mamoun Nawahdah: (sec 5) (sec 6)
- Mr. Abdallah Karakra: (sec 7)
- Mr. Nael Qaraeen: (sec 8) (sec 9)

Question	Mark
I	/ 20
II	/ 20
III	/ 10
IV	/ 20
V	/ 30
Total	/ 100

Question I (20%): Select the best answer for each of the following questions (1-10):

1) Which of the following is not a java reserved word ?

A) null **B) import** **C) args** **D) package**

2) Which of the following is generated when the source code is successfully compiled ?

A) Output **B) Bytecode** **C) Error** **D) None of the above**

3) In java , if you do not give a value to an ***int*** local variable before using it ,_____

A) It will contain a garbage value **B) It will be initialized with zero**
C) Compiler will give an error **D) None of the above**

4) The JDK command to compile a class in the file **Test.java** is

A) java Test **B) javac Test.java** **C) javac Test** **D) java Test.class**

5) Which of the following java statements causes an error:

A) int [ ] A; **B) float x = 5.6;**
C) int A[ ]; **D) None of the above**

6) What is the output of the following code fragment?

```
String str1 = "abcde";
System.out.println(str1.substring(1, 3));
```

A) abc **B) bcd** **C) bc** **D) abcd**

7) The following are all java primitive data types except:

A) integer **B) boolean** **C) char** **D) byte**

8) The access modifier that may be used for class members but not for classes themselves is:

A) final **B) public** **C) private** **D) default**

9) Suppose **b1** and **b2** are two ***BigInteger*** objects. The following statement prints the sum of b1 and b2:

```
System.out.println( b1.add(b2));
```

A) true **B) false**

10) To declare a constant in java, we must use the _____ reserved word.

A) static **B) final** **C) void** **D) double**

Answer Sheet for Question I:

Q#	1	2	3	4	5	6	7	8	9	10
Ans	C	B	C	B	B	C	A	C	A	B

2 pts. each

Question II (20%): What is the output of the following java code fragments:

I)

```
public class Test {
      public static void main ( String[] args ) {
        int[ ][ ] x = {{1, 2, 3, 4}, {5, 6, 7}};
        System.out.println( func(x[0]));
      }

      public static int func(int[] arr) {
        int result = 0;
        for (int i = 1; i < arr.length; i++)
          result += arr[i];
        return result;
      }
    }
```

ans → grade
9 → 10
10 → 7
13 → 5
18 → 3
else → #
empty → 0

Output (I)

10

9

II)

```
String  sent = "easy nice and easy";
String x = "easy", y = "hard";
String a="nice"  , b = "bad";
String newSent = "" ;
for ( int i=0;  i < 2; i++) {
     newSent = sent.replaceAll( x , y );
     x = a;
     y =  b;
  }
System.out.println( newSent );
```

Output (II)

3 2

easy bad and easy

3

5 5

Question III (10%)

Circle and describe the error in each of the following java code fragments. Each code may have ***at most one error or no errors***. If the code contains an error then you must ***circle it*** and ***describe*** it clearly in the box. ***Otherwise, you must write "No Errors" in the box.***

I)

```
class A {
        int  id;
        public static void main( String[ ]  args ) {
            A   a  =  new  A( );
            System.out.println( " id = "  +  a.id );
         }
        }
```

5

No Errors

II)

```
class B {
   static String   name;
   public static void main ( String[ ]  args ) {
       B   b  =  new  B( );
     System.out.println ( "The first char in name is " +
                          b.name.charAt(0) );
         }
        }
```

2

3

Null Pointer Exception

The following String methods ***may*** be useful:

length(): int	trim(): String	concat(String): String
split(delimiter: String):String[ ]	equals (String):boolean	compareTo(String):int

charAt(index:int): char	replace(oldChar: char, newChar: char): String

replaceFirst(oldString: String, newString: String): String
replaceAll (oldString: String, newString: String): String
substring (begin:int, end:int):String

Question IV (20%)

Complete the following method called *reverseString* in a class called *MyString* that receives a sentence of words and returns the sentence with its words reversed except the first and last words.

Example: After executing the statement:

```
String newStr = MyString.reverseString("start strong and finish to the end");
```

newStr will point to the string "start the to finish and strong end".

```
public static String reverseString ( String  sent ) {
    String [] words = sent.split(" ");
    String result = words[0] + " ";
    for (int i= words.length - 2; i > 0; i--)
        result = result + " " + words[i];
    result = result + " " + words[words.length - 1];
    return result;
}
```

4
3
8
3
2

Question 4 Solution:

```
package FormOne;
import java.util.Scanner;
public class QuestionFour {

	private static Scanner scanner = new Scanner(System.in);

	public static void main(String[] args) {

		System.out.println("enter a string : ");
		System.out.println(reverseString(scanner.nextLine()));
		scanner.close();

	}

	public static String reverseString(String str) {
		String[] temp = str.trim().split(" ");
		for (int i = 1, j = temp.length - 2; i <= (temp.length - 2) ; i++, j--) {
			str = temp[i];
			temp[i] = temp[j];
			temp[j] = str;
		}
		str = "";
		for (String x : temp)
			str += x + " ";

		return str.trim();
	}
}
```

Question V (30%)

A) (12%) Write an ***immutable*** class called ***Student*** which at least has the following members:

- attributes ***name (String), id (int), dateOfAdmission (Date)***
- non-default (argument) constructor with arguments called **name** and **id** to initialize both attributes
- A default (no-arg) constructor that calls the argument constructor to initialize name to "Sana" and id to 12.
- Any appropriate (مناسبة) setter/getter methods for attributes
  Note: ***dateOfAdmission*** should be initialized to current date inside both constructors

```
import java.util.*;

public class Student {
    private String name;
    private int id;
    private Date dateOfAdmission;

    public Student () {
        this ("Sana", 12);
        dateOfAdmission = new Date();
    }

    public Student(String name, int id) {
        this.name = name;
        this.id = id;
        dateOfAdmission = new Date();
    }

    public String getName() {
        return name;
    }
```

3

3

3

2

Question 5 Solution:

```
package FormOne;

import java.util.Date;
public class QuestionFive {

	private String name;
	private int id;
	Date dateOfAdmission;
	QuestionFive(){
		name = new String("Sana");
		id = 12;
		dateOfAdmission = new Date();
	}
	QuestionFive(String name,int id){
		this.name = name;
		this.id = id;
		dateOfAdmission = new Date();
	}
	public String getName() {
		return name;
	}
	public int getID() {
		return id;
	}
```

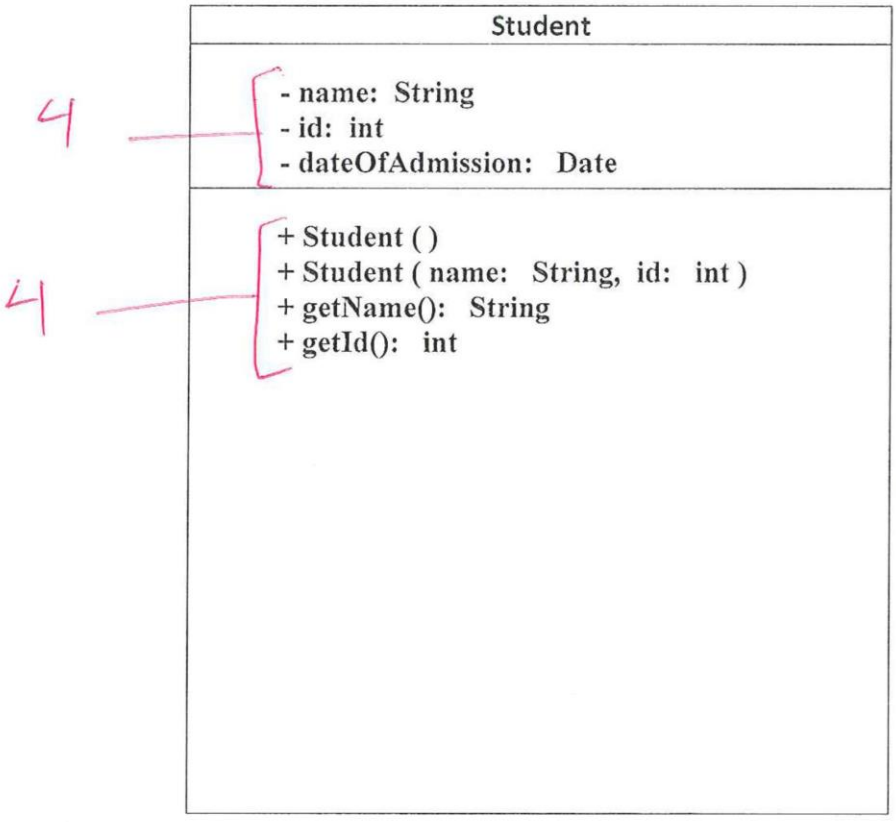
```
	public static void main(String[] args) {
		QuestionFive[] students = new QuestionFive[10];
		for(int i = 0 ; i < 10 ; i++) {
			students[i] = new QuestionFive("Ahmad_"+(i+1),i+1);
		}
		for(QuestionFive x : students) {
			System.out.println(x.getName()+" \t"+x.getID());
		}


	}

}
```

```
    public int getId() {
        return id;
    }
}
```

1

- subtract one point for each setter
- subtract two points for getDateOfAdmission

B) (8%) Complete the following *UML* diagram for class *Student*



4

4

C) (10%) Complete the *main* method in the following **Driver** class to do the following:

- Create an array of ten **Student** objects called *students* and initialize them all to name **Ahmad_x** and id number **x** in sequence where x is the sequence of numbers 1-10. (i.*e. the first Student will have the name Ahmad_1 and id 1 and the 10th student will have name Ahmad_10 and id 10*)
- Print the names and id numbers of all students in the array *students*

```
class Driver {
  public   static   void   main  ( String[ ]   args )
   {

    Student [] students = new Student[10];

    for (int i=0; i < students.length; i++){
     students[i] = new Student ( "Ahmad_" + (i+1), i+1);
    }


    for (int i=0; i< students.length; i++)
     System.out.println ( students[i].getName() + "  " +
         students[i].getId());

   }

}
```

2

4

4