



## Final 2 January 2018, questions

Object-Oriented Programming I (Concordia University)



Scan to open on Studocu

**COMP 248 EE – Final Exam**

**Instructions:**

Date: Thursday June 14<sup>th</sup>, 2012  
Time: 14:00 – 17:00 (3:00 hours)

Only ENCS-approved calculators (with the ENCS sticker) are allowed.  
No other electronic devices (including cell phones) are allowed.  
No books, papers or extra notes are allowed.  
Answer directly on this questionnaire.

**Identification:**

Last Name:	_____
First Name :	_____
Student ID :	_____
Signature:	_____

**Marking scheme (for the instructor's use only):**

QUESTION 1	QUESTION 2	QUESTION 3	QUESTION 4	QUESTION 5	QUESTION 6	TOTAL
_____	_____	_____	_____	_____	_____	_____
12	18	8	13	15	34	100

**Question 1 (8×1.5pts=12pts)** Assume the following declarations:

```
int i = 1;
char a = 'd';
boolean isEasy = false;
String s = "Question1";
int [] a1 = {1, 0, 2};
int [] a2 = {1, 0, 2};
```

For each of the following expressions, indicate its value and its type.

	Expression	Type of the expression	Value of the expression
<b>1.1</b>	'f' >= 'a' && a == 'a'		
<b>1.2</b>	a1[i++]		
<b>1.3</b>	(double)7 / 2		
<b>1.4</b>	a1.length > 2 ? a1[1]+1 : a1[1+1]		
<b>1.5</b>	a1 == a2		
<b>1.6</b>	(s.charAt(4) != 'e')    isEasy		
<b>1.7</b>	(char)('f'+4)		
<b>1.8</b>	a2[a1[2]]		

**Question 2 (9×2pts=18pts)** Multiple choice questions – Circle only one answer.

**1.1** What is the value of `result` after executing the following Java statements?

```
int a, b, c, d, result;  
a = 4;  
b = 12;  
c = 3;  
d = 5;  
result = d % a * c + b % a + a;
```

- a) 13
- b) 11
- c) 9
- d) 7
- e) None of the above

**1.2** What does the following statement print?

```
System.out.print("ABC".compareToIgnoreCase("abA"));
```

- a) 0
- b) a positive integer
- c) a negative integer
- d) ABA
- e) ABC

**1.3** What is the last line printed by the following program segment?

```
int count = 0;  
do  
{  
    System.out.println("Java");  
}  
while (count++ < 3);  
System.out.println(count);
```

- a) 0
- b) 1
- c) 3
- d) 4
- e) None of the above

**1.4** What will be the value of `x` after the following code is executed?

```
int x = 70;  
while (x < 100)  
{  
    x += 10;  
}
```

- a) 90
- b) 100
- c) 110
- d) This is an infinite loop

**1.5** What does this fragment display on the screen?

```
int[] array = {1, 4, 3, 6, 8, 2, 5};
int what = array[0];
for (int index=0; index < array.length; index++)
{
    if (array[index] > what)
        what = array[index];
}
System.out.println(what);
```

- a) 1
- b) 5
- c) 1 4 3 6 8 2 5
- d) 8

**1.6** What integer values could you enter to terminate the following while loop?

```
System.out.print("Enter an integer: ");
int number = myKeyboard.nextInt();
while (number < 100 && number > 500)
{
    System.out.print("Enter another integer: ");
    number = myKeyboard.nextInt();
}
```

- a) Integers less than 100 or greater than 500
- b) Integers in the range 100–499
- c) Integers in the range 100–500
- d) The boolean condition can never be true

**1.7** Assume the following class:

```
public class Circle
{
    private double radius;

    public Circle(double radius)
    {
        this.radius = radius;
    }
}
```

The following statement in the `main()` means:

```
Circle[] x = new Circle[10];
```

- a) `x` is an array of ten int values.
- b) `x` is a reference to an array and each element in the array can hold a `Circle` object.
- c) `x` is an array of ten `Circle` classes.
- d) `x` contains a reference to an array and each element in the array refers to ten `Circle` objects.
- e) None of the above

**1.8** What is the output of the following code?

```
public class Test
{
    public static void main(String[] args)
    {
        int[] a = {1, 2};
        swap(a[0], a[1]);
        System.out.println(a[0] + "--" + a[1]);
    }

    public static void swap(int n1, int n2)
    {
        int temp = n1;
        n1 = n2;
        n2 = temp;
    }
}
```

- a) 1--1
- b) 2--1
- c) 2--2
- d) None of the above

**1.9** Assume the following code. Which statement is not true?

```
public class Sphere
{
    private double radius;
    public double x;
    private double y;
    private double z;
}
```

- a) `x` is available to code that is written outside the `Sphere` class.
- b) `radius` is not available to code written outside the `Sphere` class.
- c) `radius`, `x`, `y`, and `z` are called members of the `Sphere` class.
- d) `z` is available to code that is written outside the `Sphere` class.

**Question 3 (8pts)** What is the output of the following program segment?

```
int[] myArray = {4, 0, 9, 2, 6, 3, 5, 7, 8, 2};
int left, right;
int last = myArray.length-1;

for (int i = 0; i < myArray.length; i++)
    System.out.print(myArray[i]);

for (left = 0, right = last-1; left < right; )
{
    while (myArray[left] > myArray[last])
        left++;
    while (myArray[right] < myArray[last])
        right--;

    System.out.print("\n" + left + " " + right + "\n");

    int temp = myArray[left];
    myArray[left] = myArray[right];
    myArray[right] = temp;

    for (int i = 0; i < myArray.length; i++)
        System.out.print(myArray[i]);
}
```

**Answer:**

#### Question 4 (13pts)

**A- (10pts)** Write a public static method called `product_n` that takes an integer `n` as parameter and returns the product of the `n` first integers. For example, if `n` is 4, your method should compute and return 24 (because  $1 \times 2 \times 3 \times 4 = 24$ ).

Note: If the value of `n` is zero or is negative, your method should return zero.

**Answer:**

**B-(3pts)** Assume that you are in the same class as the method declared in part **A** above, and you can call and use the method `product_n` that you declared above. Write the appropriate instructions to call your method and display “YES” if the product of 1 to 50 is greater than 1000; and display “NO” otherwise.

**Answer:**



**Question 5 (15pts)** The inner product (or dot product) of 2 vectors (or 2 arrays) is the sum of the product of the corresponding elements of the 2 arrays. For example, the inner product of the 2 arrays:

1.0	2.0	3.0	-4.0	-5.0	-6.0
-----	-----	-----	------	------	------

and

5.0	1.5	2.0	2.5	3.0	3.5
-----	-----	-----	-----	-----	-----

is -32, because  $(1.0 \times 5.0) + (2.0 \times 1.5) + (3.0 \times 2.0) + (-4.0 \times 2.5) + (-5.0 \times 3.0) + (-6.0 \times 3.5) = -32$

Complete the following program to determine and display on the screen the inner product of 2 arrays. The program should declare 2 arrays of doubles, ask the user to enter the values of the arrays, then call your method `innerProduct` and display the result on the screen. Your method `innerProduct` should be `static`.

**Answer:**

```
import java.util.Scanner;

public class Question5
{
    public static void main(String[] args)
    {
        // declare 2 arrays called array1 and array2 of 100 doubles each

        // read from the keyboard the values of each element of the arrays
        Scanner key = new Scanner(System.in);
        System.out.print("Enter all 200 doubles?");

        // call the static method innerProduct (that you will define below)
        // and display the result

        } // end of main

    // on the next page, define the method innerProduct
```

```
// define the method innerProduct here  
// you can assume that the 2 arrays will have the same length and  
// will have at least 1 element each
```

```
} // end of Question 5
```

**Question 6 (34 pts)** Assume that we want to write a program to manipulate musical notes. 🎵

**A- (22pts)** Making sure that encapsulation is not violated; write the definition of the class `MusicalNote`. Your class should contain:

The following attributes:

- a) one of 7 solfège syllables (DO, RE, MI, FA, SOL, LA, TI).  
A syllable can be represented by an integer value between 1 and 7 (e.g. 1 for DO, 2 for RE,...).
- b) a duration (a positive integer value)

The following methods:

- c) A default constructor with no argument that creates a note to DO-1 (i.e. syllable=DO and duration=1).
- d) A second constructor that accepts a syllable and a duration. If the syllable is not valid (it must be between 1 and 7), it is assigned the value DO; if the duration is not valid (it must be >0), then it is assigned the value 1. This constructor should call the mutators defined below.
- e) Appropriate mutator methods to set the values stored in an object's attributes. If the syllable is not valid (it must be between 1 and 7), it is assigned the value DO; if the duration is not valid (it must be >0), then it is assigned the value 1.
- f) Appropriate accessor methods to get the values stored in an object's attributes.
- g) A method called `shorten`. Each time this method is called, it shortens the duration of a note by 1 unit. The method should first check to make sure the note's duration does not go below 1; in that case, the duration remains 1. No value is returned.
- h) A method called `toString`. This method formats the instance variables as you wish (e.g. syllable-duration, like 1-4 for DO-4 or 3-4 for MI-4). Make sure that your method follows the standard use of the `toString` method.
- i) A method called `sameSyllable` that returns `true` if the syllables of the calling object and the passed object are the same; otherwise, it should return `false`.
- j) A method called `equals` that returns the `true` if the content of all of the data members of the calling object and the passed object are the same and the boolean `false` otherwise.

**Answer:**

For questions **B** to **G**, write the statement(s), which would appear in the driver program, to perform the requested tasks.

**Note:** You can only use the member methods listed in part **A** of this question.

**B. (1pt)** Declare 2 `MusicalNote` objects: `note1` whose initial value is FA-6 (1-6) and `note2` for which you don't have the details yet.

**C. (2pts)** Write the statement(s) to shorten `note1` by 15 units. Once it has been shortened, display its duration (only its duration).

**D. (2pts)** Write the statement(s) which will display on the screen the content of all instance variables of the object `note1`.

**E. (2pts)** Assuming that an object `note3` has been declared, write the statement(s) to determine which object `note1` and `note3` has the longest duration and display an appropriate message. It is possible that both notes have the same duration.

**F. (2pts)** Assuming that an object `note4` has been declared, write the statement(s) to determine whether the data members of `note1` and `note4` contain the same information and display an appropriate message.

**G. (3pts)** Create an array called `myTune` of 600 `MusicalNote` objects. For each note, ask the user for the syllable and the duration and create the object with this data.

*End of the exam*