Universidad de las Fuerzas Armadas-ESPE, DCCO- TC - OOP 7490 - Final Exam

Puntos totales 22/51 ?

This is it! The semester is over
I hope you enjoyed the ride!
Good Luck, future Telecommunications Engineers!

Se ha registrado el correo del encuestado (**mmgomez5@espe.edu.ec**) al enviar este formulario.

8 de 11 puntos

✗ Full Name (LastName FirstName) for example. Lascano Jorge * .../1

Gómez Melissa ✗

Select the right answer

✗ 1. Mongo DB is? * 0/1

○ NO-SQL Database that uses JSON by default

○ SQL Database

● Cloud Database ✗

Respuesta correcta

● NO-SQL Database that uses JSON by default

✓ 2. Modularity for OOP is expressed by? * 1/1

○ Classes and methods

○ Packages and methods

● Packages, classes and methods ✓

✓ 3. The Essence of Encapsulation is? * 1/1

● Ensure that the private implementation details of a component are insulated so they cannot be accessed or modified by other components ✓

○ Ensure that the public implementation details of a component are not insulated so they can be accessed or modified by other components

○ Ensure that the private implementation details of a component are public so they can be accessed or modified by other components

✓ 4. First Object Oriented Programming Language 1/1

● Simula ✓

○ Ada

○ SmallTalk

✓ 5. Diference between an abstract class and an interface? * 1/1

○ An interface has method implemetations, an abstraction doesn’t

○ An abstract class only has method protoypes

● An abstract class has method implemetations, an interface doesn’t ✓

✓ 6. S in SOLID principles tells that? * 1/1

○ A class must not depend on other class

○ An application must be based in interfaces

● A class only does one thing related to its target ✓

✓ 7. O in SOLID principles tells that? * 1/1

○ Software entities must be open for extension and open for modification

○ Software entities must be closed for extension but open for modification

● Software entities must be open for extension but closed for modification ✓

✓ 8. L in SOLID principles tells that? * 1/1

○ We must program towards interfaces and layers

○ Concrete classes don’t depend on abstract classes

● Specialization Objects can replace its generalization objects ✓

✗ 9. The state of an object is given by? * 0/1

○ its attributes

● its methods ✗

○ its name

Respuesta correcta

● its attributes

✓ 10. What Programming gParadigm did you study this semester? * 1/1

● Object Oriented Programming Paradigm ✓

○ Functional Programming

○ Procedural Programming

Which of the following sentences are NOT true? (mark with an X) **5 de 10 puntos**

	NOT TRUE	Puntuación	
We should not use humor when naming variables	☐	.../1	✗
We should use problem domain names, instead of using generic identifier names	☐	.../1	✗
The first rule of functions is that they should be as long as need, as far as they solve the problem	☐	0/1	✗
It is OK that one function performs several and different not related actions	☑	1/1	✓
It is a good practice of programming to have more than three arguments for a method	☑	1/1	✓
We should use verbs for naming classes	☑	1/1	✓
We should use nouns to name methods	☑	1/1	✓
We should always comment every method, every class, and every attribute	☑	1/1	✓
Vertical Formatting means indent every line of code	☐	0/1	✗
Team rules must be adjusted to the rules agreed by every programmer's rule	☐	0/1	✗

Answer with the name of the design pattern that matches its definition, please use Capital letters for all answers: SINGLETON, STRATEGY, TEMPLATE METHOD

9 de 10 puntos

✓ 21.22 Defines a family of algorithms, encapsulates each one, and make them interchangeable. This pattern lets the algorithm vary independently from the clients that use it * 2/2

Strategy ✗

Respuestas correctas

STRATEGY

Strategy

✓ 23.24 Defines the skeleton of an algorithm in an operation, deferring some steps to subclasses. This pattern lets subclasses redefine certain steps of an algorithm without changing the algorithm's structure * 2/2

Template Method ✗

Respuestas correctas

TEMPLATE METHOD

Template Method

✓ 25. 26. Ensure a class only has one instance, and provide a global point of access to it * 2/2

Singleton ✓

Answer the code snippet questions according to your understanding of what is happening in the Java programing language

✗ 27. int a; (what is a?) * 0/1

Variable ✗

Respuesta correcta

VARIABLE

✓ 28. Calculator.add(3,4); what is add? * 1/1

Method ✓

✓ 29. Person person = new Person(); (What is Person()?) * 1/1

Constructor ✓

✓ 30. JDK stands for * 1/1

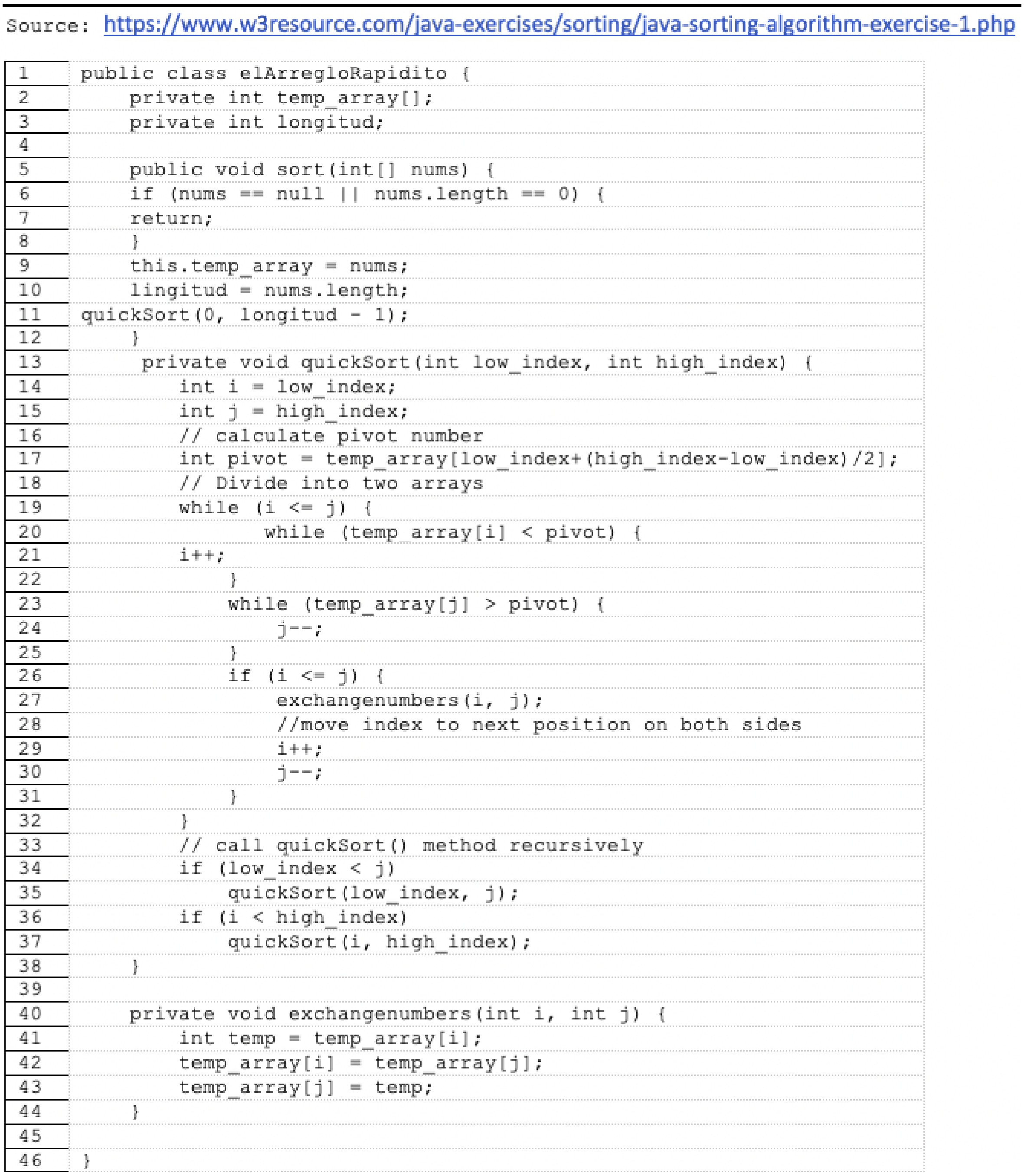
Java Development Kit ✓

Clean Code

0 de 20 puntos

Find the 20 lines that make the following snippet less understandable. Only select the 20 ugly lines of code. If you select more than 20 lines of code. Your grade will be deducted based on those extra selected lines

Ugly QuickSort implementation in Java



Source: https://www.w3resource.com/java-exercises/sorting/java-sorting-algorithm-exercise-1.php

```
public class elArregloRapidito {
    private int temp_array[];
    private int longitud;

    public void sort(int[] nums) {
    if (nums == null || nums.length == 0) {
    return;
    }
    this.temp_array = nums;
    lingitud = nums.length;
quickSort(0, longitud - 1);
    }
     private void quickSort(int low_index, int high_index) {
        int i = low_index;
        int j = high_index;
        // calculate pivot number
        int pivot = temp_array[low_index+(high_index-low_index)/2];
        // Divide into two arrays
        while (i <= j) {
               while (temp_array[i] < pivot) {
        i++;
            }
            while (temp_array[j] > pivot) {
                j--;
            }
            if (i <= j) {
                exchangenumbers(i, j);
                //move index to next position on both sides
                i++;
                j--;
            }
        }
        // call quickSort() method recursively
        if (low_index < j)
            quickSort(low_index, j);
        if (i < high_index)
            quickSort(i, high_index);
    }

    private void exchangenumbers(int i, int j) {
        int temp = temp_array[i];
        temp_array[i] = temp_array[j];
        temp_array[j] = temp;
    }

}
```

only click the 20 ugly lines of code

	Column 1	Puntuación	
1	○	0/1	✗
2	○	0/1	✗
3	○	0/1	✗
4	●	.../0	✗
5	○	0/1	✗
6	○	0/1	✗
7	○	0/1	✗
8	●	.../0	✗
9	○	.../0	✗
10	○	0/1	✗
11	○	0/1	✗
12	●	.../0	✗
13	○	0/1	✗
14	○	.../0	✗
15	○	.../0	✗
16	●	.../0	✗
17	○	0/1	✗

18	●	.../0	✗
19	○	.../0	✗
20	○	0/1	✗
21	○	.../0	✗
22	○	.../0	✗
23	○	0/1	✗
24	○	.../0	✗
25	○	.../0	✗
26	○	.../0	✗
27	○	0/1	✗
28	●	.../0	✗
29	○	.../0	✗
30	○	.../0	✗
31	○	.../0	✗
32	○	.../0	✗
33	●	.../0	✗
34	○	0/1	✗
35	○	0/1	✗
36	○	0/1	✗

37	○	0/1	✗
38	○	.../0	✗
39	○	.../0	✗
40	○	0/1	✗
41	○	0/1	✗
42	○	0/1	✗
43	○	.../0	✗
44	○	.../0	✗
45	○	.../0	✗
46	○	.../0	✗

Respuestas correctas

	Column 1
4	○
8	○
12	○
16	○
18	○
28	○
33	○