



Navegación por el cuestionario



Finalizar revisión

Comenzado el	lunes, 31 de diciembre de 2018, 17:33
Estado	Finalizado
Finalizado en	miércoles, 2 de enero de 2019, 12:25
Tiempo empleado	1 día 18 horas
Calificación	7,73 de 10,00 (77%)

Pregunta 1

Correcta

Puntuación 0,67 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
public class Programa {  
    public static void main(String[] args) {  
        int[] contarOpciones = new int[3];  
        int[] opciones = {0, 1, 2, 2, 1, 0, 0, 1, 0, 2, 0, 1, 1, 2};  
  
        for(int i = 0; i < opciones.length; i++) {  
            int opcion = opciones[i];  
            contarOpciones[opcion]++;  
        }  
        System.out.println(Arrays.toString(contarOpciones));  
    }  
}
```

¿Qué se mostrará por consola?

Respuesta: [5, 5, 4] ✓

La respuesta correcta es: [5, 5, 4]

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,67/1,00.

Pregunta 2

Correcta

Puntuación 1,00 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
int[] nums = {5, 15, 25, 35, 45, 55, 65, 75, 85, 95};  
  
int num = nums[0];  
for (int i = 0; i < nums.length - 1; i++) {  
    nums[i] = nums[i + 1];  
}  
nums[nums.length - 1] = num;  
  
System.out.println(Arrays.toString(nums));
```

¿Qué se mostrará por consola?

Respuesta: [15, 25, 35, 45, 55, 65, 75, 85, 95, 5] ✓

La respuesta correcta es: [15, 25, 35, 45, 55, 65, 75, 85, 95, 5]

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 3

Correcta

Puntuación 1,00 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
for (int i = 0; i < nums.length - 1; i++) {  
    if (nums[i] > nums[i + 1]) {  
        nums[i + 1]++;  
    }  
}  
  
System.out.println(Arrays.toString(nums));
```

¿Qué se mostrará por consola con cada uno de los arrays siguientes?

Array	Consola
int[] nums = {5};	[5] ✓
int[] nums = {15, 5};	[15, 6] ✓
int[] nums = {15, 8, 9, 5, 5};	[15, 9, 9, 6, 6] ✓
int[] nums = {5, 1, 3, 2, 0, 5};	[5, 2, 3, 3, 1, 5] ✓

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 4

Correcta

Puntuación 1,00 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
int[] nums1 = {1, 2, 3, 4, 5};  
int[] nums2 = {1, 5, 10, 15, 25};  
  
for (int i = 0; i < nums1.length; i++) {  
    nums1[i] += nums2[nums2.length - 1 - i];  
}  
  
System.out.println(Arrays.toString(nums1));
```

¿Qué se mostrará por consola?

Respuesta: [26, 17, 13, 9, 6] ✓

La respuesta correcta es: [26, 17, 13, 9, 6]

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 5

Incorrecta

Puntuación 0,00 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
public class Programa {  
    public static final int PIEDRA = 0;  
    public static final int PAPEL = 1;  
    public static final int TIJERA = 2;  
    public static final String[] OPCIONES = {"papel", "tijera", "piedra"};  
  
}
```

¿Qué valores tendrán las siguientes expresiones?

Expresión	Valor
OPCIONES[PAPEL]	tijera ❌
OPCIONES[TIJERA]	piedra ❌
OPCIONES[PIEDRA]	papel ❌

**Incorrecta**

Puntos para este envío: 0,00/1,00.

Pregunta **6**

Correcta

Puntuá 0,67 sobre 1,00

🚩 Marcar pregunta

Tenemos el siguiente código:

```
Scanner leer = new Scanner(new File("datos.txt"));

while (leer.hasNextLine()) {
    String linea = leer.nextLine();
    while (linea.length() > 50) {
        System.out.println(linea.substring(0, 50));
        linea = linea.substring(50);
    }
    System.out.println(linea);
}

leer.close();Indica qué afirmaciones son válidas.
```

Seleccione una o más de una:

- ☒ a. Fuera del while se muestran las líneas menores de 50 caracteres ✓
- ☒ b. Dentro del while (linea.length() > 50) la línea se divide en trozos de 50 caracteres ✓
- ☒ c. linea.substring(50) se queda con los primeros 50 caracteres del String
- ☒ d. Si el fichero está vacío se producirá un error

Respuesta correcta

La respuesta correcta es: Dentro del while (linea.length() > 50) la línea se divide en trozos de 50 caracteres, Fuera del while se muestran las líneas menores de 50 caracteres

**Correcta**

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,67/1,00.

Pregunta **7**

Correcta

Puntuá 0,80 sobre 1,00

🚩 Marcar pregunta

Corrige el siguiente programa para que cuente cuántos alumnos y alumnas no han superado el curso(< 5), cuántos han tenido un rendimiento excelente (>= 9) y cuántos un rendimiento bueno (>= 5 y < 9).

Para ello, se utilizará el array seguimiento.

**Respuesta:** (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
1 public class Programa {
2     public static void main(String[] args) {
3         double[] notas = {9.5, 7.3, 5.9, 9.6, 6.5,7};
4         int[] seguimiento = new int [3];
5
6         for (int i = 0; i < notas.length; i++) {
7             if (notas[i] <= 5) {
8                 seguimiento[0]++;
9             } else if (notas[i] > 9) {
10                seguimiento[1]++;
11            } else {
12                seguimiento[2]++;
13            }
14        }
15        System.out.println("Rendimiento excelente " + seguimiento[1]);
16        System.out.println("Rendimiento bueno " + seguimiento[2]);
17    }
18 }
```

	Expected	Got	
✓	Rendimiento excelente 2	Rendimiento excelente 2	✓
	Rendimiento bueno 4	Rendimiento bueno 4	

Todas las pruebas superadas. ✓

Question author's solution:

```
public class Programa {
    public static void main(String[] args) {
        double[] notas = {4.7, 5.0, 7.3, 5.9, 9.6, 6.5, 9.0};
        int[] seguimiento = new int [3];

        for (int i = 0; i < notas.length; i++) {
            if (notas[i] < 5) {
                seguimiento[0]++;
            } else if (notas[i] >= 9) {
                seguimiento[2]++;
            } else {
                seguimiento[1]++;
            }
        }
        System.out.println("Rendimiento excelente " + seguimiento[2]);
        System.out.println("Rendimiento bueno " + seguimiento[1]);
    }
}
```

**Correcta**

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,80/1,00.

Pregunta **8**

Correcta

Puntuá 0,70 sobre 1,00

🚩 Marcar pregunta

Corrige el siguiente método para que diga si en un array hay tres números consecutivos como se muestra a continuación.

```
int[] nums = {1, 3, 5, 7, 9};
boolean result = metodo(nums);
System.out.println("Resultado: " + result);
```

Resultado: false

```
int[] nums = {1, 3, 4, 5, 9};
boolean result = metodo(nums);
System.out.println("Resultado: " + result);
```

Resultado: true

**Respuesta:** (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
1 public static boolean metodo(int[] nums) {
2     for (int i = 0; i < nums.length - 1; i++) {
3         if (nums[i + 1] == nums[i] + 1 && nums[i + 2] == nums[i] + 2) {
4             return true;
5         }
6     }
7     return false;
8 }
```

```
5     }
6     }
7     return false;
8 }
```

	Test	Expected	Got	
✓	int[] nums = {1, 3, 5, 7, 9}; boolean result = metodo(nums); System.out.println("Resultado: " + result);	Resultado: false	Resultado: false	✓
✓	int[] nums = {1, 3, 5, 6, 7}; boolean result = metodo(nums); System.out.println("Resultado: " + result);	Resultado: true	Resultado: true	✓
✓	int[] nums = {1, 2, 3, 7, 9}; boolean result = metodo(nums); System.out.println("Resultado: " + result);	Resultado: true	Resultado: true	✓
✓	int[] nums = {1}; boolean result = metodo(nums); System.out.println("Resultado: " + result);	Resultado: false	Resultado: false	✓

Todas las pruebas superadas. ✓

Question author's solution:

```
public boolean metodo(int[] nums) {
    for (int i = 0; i < nums.length - 2; i++) {
        if (nums[i + 1] == nums[i] + 1 && nums[i + 2] == nums[i] + 2) {
            return true;
        }
    }
    return false;
}
```

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,70/1,00.

Pregunta 9

Correcta

Puntúa 1,00 sobre 1,00

⚑ Marcar pregunta

Escribe el código para los métodos palabraBuscada y mostrarResultados.

palabraBuscada recibirá como parámetro una palabra y devolverá el índice de esa palabra en el array PALABRAS. Si la palabra no está en el array devolverá -1. mostrarResultados recibirá como parámetro una array de enteros y mostrará lo siguiente:

```
public static final String[] PALABRAS = {"de", "y", "el", "la"};
```

```
int[] contador = {4, 2, 1, 3};
mostrarResultados(contador);
```

Palabra de: 4  
Palabra y: 2  
Palabra el: 1  
Palabra la: 3

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
21
22
23     mostrarResultados(contador);
24
25     public static int palabraBuscada(String palabra) {
26         int result = -1;
27         for(int i = 0; i < PALABRAS.length ; i++){
28             if(PALABRAS[i].equals(palabra)){
29                 return i;
30             }
31         }
32         return result;
33     }
34
35     public static void mostrarResultados(int[] contador) {
36         for(int i = 0; i < PALABRAS.length ; i++){
37             System.out.println("Palabra " + PALABRAS[i] + ": " + contador[i]);
38         }
39     }
40 }
```

	Expected	Got	
✓	Palabra de: 4 Palabra y: 2 Palabra el: 1 Palabra la: 3	Palabra de: 4 Palabra y: 2 Palabra el: 1 Palabra la: 3	✓

Todas las pruebas superadas. ✓

Question author's solution:

```
import java.io.*;
import java.util.*;

public class Programa {
    public static final String[] PALABRAS = {"de", "y", "el", "la"};
    public static final int NUM = 4;
    public static void main(String[] args) throws FileNotFoundException {
        Scanner leer = new Scanner(new File("datos.txt"));
        int[] contador = new int[NUM];

        while (leer.hasNext()) {
            String palabra = leer.next();
            int indice = palabraBuscada(palabra);
            if (indice != -1) {
                contador[indice]++;
            }
        }

        mostrarResultados(contador);
    }

    public static void mostrarResultados(int[] contador) {
        for (int i = 0; i < NUM; i++) {
            System.out.println("Palabra " + PALABRAS[i] + ": " + contador[i]);
        }
    }

    public static int palabraBuscada(String palabra) {
        int result = -1;

        for (int i = 0; i < NUM; i++) {
            if (PALABRAS[i].equals(palabra)) {
```

```
        return i;
    }
}
return result;
}
```

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 10

Correcta

Puntuo 0,90 sobre 1,00

Marcar pregunta

Escribe el código para el método capicua. Recibirá como parámetro un array de números y devolverá true si el array es capicúa y false en caso contrario.:

```
int[] nums1 = {12, 4, 7, 25};
boolean resultado = capicua(nums);
System.out.println("Resultado: " + resultado);
```

Resultado: false

```
int[] nums1 = {2, 4, 7, 4 2};
boolean resultado = capicua(nums);
System.out.println("Resultado: " + resultado);
```

Resultado: true

Respuesta: (penalty regime: 10, 20, ... %)

```
1 public static boolean capicua(int[] numeros){
2     for(int i = 0; i < numeros.length/2; i++){
3         if(numeros[i] != numeros[numeros.length - 1 -i]){
4             return false;
5         }
6     }
7     return true;
8 }
```

Test	Expected	Got	
✓ int[] nums = {12, 4, 7, 25}; boolean resultado = capicua(nums); System.out.println("Resultado: " + resultado);	Resultado: false	Resultado: false	✓
✓ int[] nums = {2, 4, 7, 4, 2}; boolean resultado = capicua(nums); System.out.println("Resultado: " + resultado);	Resultado: true	Resultado: true	✓
✓ int[] nums = {12, 4, 4, 12}; boolean resultado = capicua(nums); System.out.println("Resultado: " + resultado);	Resultado: true	Resultado: true	✓

Todas las pruebas superadas. ✓

Question author's solution:

```
public static boolean capicua(int[] nums) {
    for (int i = 0; i < nums.length / 2; i++) {
        if (nums[i] != nums[nums.length - 1 - i] ) {
            return false;
        }
    }
    return true;
}
```

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,90/1,00.

Finalizar revisión



Urrutiko Lanbide Heziketako Institutua  
Instituto de Formación Profesional a Distancia

Contacta con nosotros

📍 Dirección: Calle Álava 41, interior - Vitoria-Gasteiz  
☎ Teléfono : 945 667 963  
✉ E-mail: ulhi@ulhi.net  
🐦 Twitter: @UrrutikoLH

