

Programación

EXAMEN TEMA 3 – ESTRUCTURAS DE DATOS

(27/11/2024)

1. (0,75p) ¿Qué devuelve la siguiente sentencia?

```
String[] nombres = new String[2];
nombres[2] = "Pepe";
for(int n : nombres) {
    System.out.println(n + " ");
}
```

2. (0,75p) ¿Qué sacará por pantalla el siguiente código Java?

```
int[] nums = {1, 2, 3, 4, 5};
int[] copiaArray;
copiaArray = nums;
copiaArray[3] = 0;
System.out.println(Arrays.toString(nums));
System.out.println(Arrays.toString(copiaArray));
```

3. (0,75p) ¿Qué sacará por pantalla el siguiente código Java? Identifica sobre el método `.arraycopy()` lo que significa cada uno de los elementos que contiene.

```
int[] primos = {2, 3, 5, 7, 11, 13, 17};
int[] copia = new int[primos.length];
System.arraycopy(primos, 2, copia, 4, 3);
System.out.println(Arrays.toString(primos));
System.out.println(Arrays.toString(copia));
```

4. (1,5p) ¿Qué 3 opciones tenemos si queremos buscar un elemento dentro de un vector o *array*? Explica el tipo de dato que devuelve cada una.
5. (0,5p) ¿Qué hace *Java* cada vez que usamos la palabra ***return***?
6. (1,5p) Explica paso a paso qué haría el algoritmo SELECCIÓN DESCENDENTE para ordenar el siguiente vector:

[34,23,10,45,4,6,0]

- ¿Es un tipo de algoritmo *in-place* o *no-in-place*? ¿Por qué?
- ¿Por qué este tipo de algoritmo es muy lento si lo comparamos con otros más eficientes, como por ejemplo el *QuickSort*?
- ¿Qué método podemos utilizar en *Java* para realizar una ordenación rápidamente?

7. (1,5p) Explica paso a paso qué haría el algoritmo de búsqueda binaria para encontrar los valores **8** y **24** dentro del vector dado.

[3,8,14,19,24,35,48]

- ¿Por qué es tan eficiente?
- ¿Qué devuelve el método `.binarySearch()` si decidimos aplicarla en *Java* para buscar el número 8?

8. (0,5p) ¿Qué sacará por pantalla el siguiente código Java?

```
int vector[] = {3,3,7,8,8,9,10,15,15};
int vector2[] = Arrays.stream(vector).distinct().toArray();
System.out.println(Arrays.toString(vector));
System.out.println(Arrays.toString(vector2));
```

9. (1,5p) Realiza una traza de los siguientes programas:

a)

```
for (int i = 1; i <= 5; i++) {
    for (int j = 1; j <= i; j++) {
        System.out.print(j + " ");
    }
    System.out.println();
}
```

b)

```
bucle1:
for (int i = 5; i >= 1; i--) {
    bucle2:
    for (int j = 1; j <= i; j++) {
        if (i == 3 && j == 3) {
            break bucle1;
        }
        System.out.print(j + " ");
    }
    System.out.println();
}
```

10. (0,75p) Dibuja la matriz resultante de ejecutar el siguiente código:

```
String vector[] = {"Pepe","Juan","Patri","Carlos","Bruna"};
String matriz[][] = new String[3][5];
for (int i = 0; i < matriz[0].length; i++) {
    matriz[2][i] = vector[i];
}
```