

Problema 5: Practicar con arrays, recorridos y búsquedas

Datos personales	
Apellidos:	
Nombre:	

1 Arrays de números reales

Objetivos

Practicar el recorrido de un array de números.

Descripción

Escribir un método estático Java que reciba como parámetro un array de números reales conteniendo ángulos en grados, y que retorne otro array del mismo tamaño con todos los ángulos del original convertidos a radianes y en el orden inverso (la primera casilla contendrá el último ángulo, la segunda el penúltimo, etc.)

Respuesta:

<poner aquí el código del método>

2 Recorrido parcial de un array

Objetivos

Practicar el recorrido parcial de un array de números y el paso de pseudocódigo a código.

Descripción

Escribir un método en Java que codifique el siguiente algoritmo descrito mediante pseudocódigo, que calcula la máxima diferencia entre una casilla del array x y la anterior:

```
método estático maximaDiferencia(double[] x) retorna real
    real maxDif=menor número real posible
    // bucle para todas las casillas de x excepto la primera
    para i desde 1 hasta el último índice de x
        real dif=valor absoluto(x[i]-x[i-1])
        si dif>maxDif entonces
            maxDif=dif
        fin si
    fin para
    retorna maxDif
fin del método
```

Introducción al Software, Curso 2017-2018

Observar que es un recorrido parcial porque no se empieza en la primera casilla sino en la segunda (que es la de índice 1).

Respuesta:

<Poner aquí el método en Java, sin olvidar el comentario de documentación>

3 Recorrido de un array de Strings

Objetivos

Practicar con el uso de un array de Strings

Descripción

Escribir un método Java al que se le pasa como parámetro un String llamado texto que contiene un texto largo con muchas palabras separadas por espacios en blanco. El método deberá contar y retornar el número de veces que aparece en el texto la palabra indicada en su segundo parámetro, llamado palabra. La cabecera del método será:

```
public static int contarApariciones(String texto, String palabra)
```

Para ello el método debe obtener en primer lugar un array de Strings conteniendo en cada casilla una palabra. Se puede usar para ello el método de la clase String llamado split(), definido así:

```
String[] split(String separador)
```

Si se usase como separador "-", las palabras deberían ir separadas por caracteres '-'. En cambio, si se usa "\\s" esto es un carácter especial que indica cualquier tipo de espacio en blanco (espacios, tabuladores o saltos de línea). Por último, si se usa como separador "\\s+", esto indica que las palabras se separan por cualquier secuencia de uno o más espacios en blanco de cualquier tipo, que es lo más apropiado en este caso.

Posteriormente, el método deberá hacer un recorrido de todas las casillas del array obtenido y contar aquellas iguales al parámetro palabra. Finalmente retornará el valor de esta cuenta.

Respuesta:

<poner aquí el código Java del método>

4 Búsqueda de un objeto en un array

Objetivos

Practicar con el algoritmo de búsqueda en arrays

Descripción

Introducción al Software, Curso 2017-2018

Se dispone de la siguiente clase que contiene los datos de un parte de trabajo:

```
public class ParteTrabajo {
    private int horas; // Horas trabajadas en el día
    private String trabajador; // Nombre del trabajador
    private String dia; // "Lunes", "Martes", "Miércoles", ... , "Domingo"

    public ParteTrabajo(int horas, String trabajador, String dia) {...}
    public String getDiaSemana() { return dia;}
    public String getTrabajador() { return trabajador;}
    ...
}
```

Se pide escribir un método perteneciente a otra clase que, dado un array de objetos de la clase ParteTrabajo que se pasa como parámetro, nos devuelva un String que indique el día en que trabajó "Andrés", o null si ese trabajador no se encuentra en el array.

Respuesta:

<poner aquí el código Java del método>