



Programación

Act05_Arrays

Francisco José García Cutillas | 1FPGS_DAM



Índice

| | |
|----------------------------|----|
| Funciones auxiliares | 3 |
| Ejercicio A01..... | 3 |
| Ejercicio A02..... | 5 |
| Ejercicio A03..... | 7 |
| Ejercicio A04..... | 10 |
| Ejercicio A05..... | 12 |
| Ejercicio A06..... | 14 |
| Ejercicio A07..... | 15 |
| Ejercicio A08..... | 17 |
| Ejercicio A09..... | 19 |
| Ejercicio A10..... | 21 |
| Ejercicio A11..... | 23 |

Funciones auxiliares

Para la realización de esta tarea nos hemos creado una función auxiliar en la cual introducimos el array que deseemos y ésta nos lo imprime por pantalla.

```
//Función creada para usarla cada vez que queramos imprimir un array por pantalla
public void imprimeArray(int arrayEntrada[]) {

    for (int i = 0; i < arrayEntrada.length; i++) {
        System.out.println(arrayEntrada[i]);
    }

}
```

Ejercicio A01

Implementa y utiliza un método que reciba un array de 5 números enteros y devuelva un entero indicando cuántos de ellos son superiores a la media de los valores que contiene dicho array.

```
package com.mycompany.garciacutillasfranciscojoseactut05arrays;

/**
 *
 * @author Fran
 */
public class GarciaCutillasFranciscoJoseActUt05Arrays {

    public static void main(String[] args) {

        //Creación de las variables necesarias para llamar a los ejercicios
        EjerciciosUt05 ej05 = new EjerciciosUt05();
        ej05.ejercicio01();

    }

}
```

```

package com.mycompany.garciacutillasfranciscojoseactut05arrays;

public class EjerciciosUt05 {

    //Creación de arrays que se pueden utilizar a lo largo de la tarea
    int array5[] = {2, 4, 6, 8, 10};

    public void ejercicio01() {

        System.out.println(x: "****RESOLUCIÓN EJERCICIO 1****");
        System.out.println(x: "");

        //Creación de la variable para llamar al método de este ejercicio
        FuncionesUt05 fun05 = new FuncionesUt05();

        //Llamada a la función
        int mayoresMedia = fun05.mayorMedia(arrayEntrada: array5);

        //Salida por pantalla del resultado
        System.out.println("Hay " + mayoresMedia + " números mayores que la media");

    }

}

```

```

package com.mycompany.garciacutillasfranciscojoseactut05arrays;

public class FuncionesUt05 {

    public int mayorMedia(int arrayEntrada[]) {

        int sumaMedia = 0;
        int mayorMedia = 0;

        //Recorrido del array para hacer la suma para calcular la media
        for (int i = 0; i < arrayEntrada.length; i++) {
            sumaMedia += arrayEntrada[i];
        }

        //Cálculo de la media
        double media = sumaMedia / arrayEntrada.length;

        //Recorrido del array para ver qué valores del mismo son mayores que la media obtenida
        for (int i = 0; i < arrayEntrada.length; i++) {

            if (arrayEntrada[i] > media) {
                mayorMedia++;
            }

        }

        return mayorMedia;

    }

}

```

```

****RESOLUCIÓN EJERCICIO 1****

Hay 2 números mayores que la media
-----
BUILD SUCCESS
-----
Total time: 1.020 s
Finished at: 2022-11-24T19:42:25+01:00
-----

```

Ejercicio A02

Implementa y utiliza un método que reciba un array de 10 enteros y muestre por pantalla cuáles de ellos son el mayor, el menor y cuántas veces se repite el mayor y el menor en dicho array.

```
package com.mycompany.garciacutillasfranciscojoseactut05arrays;

/**
 *
 * @author Fran
 */
public class GarciaCutillasFranciscoJoseActUt05Arrays {

    public static void main(String[] args) {

        //Creación de las variables necesarias para llamar a los métodos de los ejercicios
        /*
        EjerciciosUt05 ej01 = new EjerciciosUt05();
        ej01.ejercicio01();
        */
        EjerciciosUt05 ej02 = new EjerciciosUt05();
        ej02.ejercicio02();
    }
}
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
```

```
public void ejercicio02() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 2****");
    System.out.println(x: "");

    //Creación de la variable para llamar al método de este ejercicio
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Llamada a la función
    fun05.mayorMenorRepite(arrayEntrada: array10);

}
```

```

public void mayorMenorRepite(int arrayEntrada[]) {

    int mayor = 0;
    int menor = 0;
    int repMayor = 0;
    int repMenor = 0;

    //Recorrido del array
    for (int i = 0; i < arrayEntrada.length; i++) {

        //Guardado del mayor y menor en la primera vuelta
        if (i == 0) {

            mayor = arrayEntrada[i];
            menor = arrayEntrada[i];

        }

        //Condición para calcular el mayor y veces que se repite
        if (arrayEntrada[i] >= mayor) {

            if (arrayEntrada[i] == mayor) {
                repMayor++;
            }
            mayor = arrayEntrada[i];
        }

        //Condición para calcular el menor y veces que se repite
        if (arrayEntrada[i] <= menor) {

            if (arrayEntrada[i] == menor) {
                repMenor++;
            }
            menor = arrayEntrada[i];
        }
    }

    //Salida por pantalla del resultado
    System.out.println("El número mayor del array es el " + mayor +
        " y se repite " + repMayor + " veces");
    System.out.println("El número menor del array es el " + menor + " "
        + "y se repite " + repMenor + " veces");

}

```

****RESOLUCIÓN EJERCICIO 2****

El número mayor del array es el 128 y se repite 2 veces
 El número menor del array es el -2 y se repite 2 veces

 BUILD SUCCESS

Total time: 0.925 s
 Finished at: 2022-11-26T12:44:45+01:00

Ejercicio A03

Implementa y utiliza un método que reciba un array de enteros (podemos hacer las pruebas con array de 5 y de 10 enteros) y devuelva un array con los mismos enteros en orden inverso al de entrada. Por ejemplo, si el array de entrada tiene los valores 2, 4, 3, 1 y 5, el de salida debe tener 5, 1, 3, 4 y 2.

```
package com.mycompany.garciacutillasfranciscojoseactut05arrays;

/**
 *
 * @author Fran
 */
public class GarciaCutillasFranciscoJoseActUt05Arrays {

    public static void main(String[] args) {

        //Creación de las variables necesarias para llamar a los métodos de los ejercicios
        /*
        EjerciciosUt05 ej01 = new EjerciciosUt05();
        ej01.ejercicio01();

        EjerciciosUt05 ej02 = new EjerciciosUt05();
        ej02.ejercicio02();
        */
        EjerciciosUt05 ej03 = new EjerciciosUt05();
        ej03.Ejercicio3();
    }
}
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
```

```
public void Ejercicio3() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 3****");
    System.out.println(x: "");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Llamada a la función para que me devuelva el array inverso
    int arrayResultado[] = fun05.arrayInverso(arrayEntrada: array10);

    //Salida por pantalla del resultado
    System.out.println(x: "El array inverso de:");
    fun05.imprimeArray(arrayEntrada: array10);

    System.out.println(x: "");
    System.out.println(x: "Es: ");
    fun05.imprimeArray(arrayEntrada: arrayResultado);

}
```

```
public int[] arrayInverso(int arrayEntrada[]) {  
    int arraySalida[] = new int[arrayEntrada.length];  
    int j = arrayEntrada.length - 1;  
    for (int i = 0; i < arrayEntrada.length; i++) {  
        arraySalida[i] = arrayEntrada[j];  
        j--;  
    }  
    return arraySalida;  
}
```

****RESOLUCIÓN EJERCICIO 3****

El array inverso de:

2
4
6
8
10

Es:

10
8
6
4
2


```
****RESOLUCIÓN EJERCICIO 3****
```

```
El array inverso de:
```

```
-2
```

```
4
```

```
6
```

```
-2
```

```
110
```

```
0
```

```
14
```

```
128
```

```
128
```

```
20
```

```
Es:
```

```
20
```

```
128
```

```
128
```

```
14
```

```
0
```

```
110
```

```
-2
```

```
6
```

```
4
```

```
-2
```

Ejercicio A04

Implementa y utiliza un método que recibe dos parámetros: un array de 10 enteros y un número entero. El método debe devolver el número de veces que se encuentra dicho número en el array.

```
package com.mycompany.garciacutillasfranciscojoseactut05arrays;

/**
 *
 * @author Fran
 */
public class GarciaCutillasFranciscoJoseActUt05Arrays {

    public static void main(String[] args) {

        //Creación de las variables necesarias para llamar a los métodos de los ejercicios
        /*
        EjerciciosUt05 ej01 = new EjerciciosUt05();
        ej01.ejercicio01();

        EjerciciosUt05 ej02 = new EjerciciosUt05();
        ej02.ejercicio02();

        EjerciciosUt05 ej03 = new EjerciciosUt05();
        ej03.Ejercicio3();
        */
        EjerciciosUt05 ej04 = new EjerciciosUt05();
        ej04.Ejercicio4();

    }
}
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
```

```
public void Ejercicio4() {

    System.out.println("****RESOLUCIÓN EJERCICIO 4****");
    System.out.println("");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Petición del número y llamada a la función para contar las veces que se repite
    int numeroVeces = 0;
    int numeroIntroducido = 0;

    Scanner sc = new Scanner(System.in);

    System.out.println("Introduce un número para ver cuántas veces se repite:");
    numeroIntroducido = sc.nextInt();

    numeroVeces = fun05.cuentaNumeroArray(arrayEntrada: array10, numero: numeroIntroducido);

    //Salida por pantalla del resultado
    System.out.println("El número " + numeroIntroducido + " se encuentra " + numeroVeces + " veces.");

}
```

```

public int cuentaNumeroArray(int arrayEntrada[], int numero) {

    int vecesNumero = 0;

    for (int i = 0; i < arrayEntrada.length; i++) {

        if (arrayEntrada[i] == numero) {
            vecesNumero++;
        }

    }

    return vecesNumero;

}

```

****RESOLUCIÓN EJERCICIO 4****

Introduce un número para ver cuántas veces se repite:

110

El número 110 se encuentra 1 veces.

BUILD SUCCESS

Total time: 10.025 s

Finished at: 2022-11-26T13:29:54+01:00

****RESOLUCIÓN EJERCICIO 4****

Introduce un número para ver cuántas veces se repite:

128

El número 128 se encuentra 2 veces.

BUILD SUCCESS

Total time: 3.508 s

Finished at: 2022-11-26T13:30:41+01:00

****RESOLUCIÓN EJERCICIO 4****

Introduce un número para ver cuántas veces se repite:

-5

El número -5 se encuentra 0 veces.

BUILD SUCCESS

Total time: 3.918 s

Finished at: 2022-11-26T13:31:14+01:00

Ejercicio A05

Implementa y utiliza un método que recibe dos parámetros: un array de 10 enteros y un número entero. El método debe devolver la primera vez que se encuentra dicho número en el array (si no se encuentra devolverá cero). Por ejemplo, si el array es 2, 3, 1, 4, 5, 6, 7, 8, 5, 1 y 2 y nos indican el valor 5, el método deberá devolver el valor 5 (ojo, aunque en el array es la posición [4]).

```
package com.mycompany.garciacutillasfranciscojoseactut05arrays;

/**
 *
 * @author Fran
 */
public class GarciaCutillasFranciscoJoseActUt05Arrays {

    public static void main(String[] args) {

        //Creación de las variables necesarias para llamar a los métodos de los ejercicios
        /*
        EjerciciosUt05 ej01 = new EjerciciosUt05();
        ej01.ejercicio01();

        EjerciciosUt05 ej02 = new EjerciciosUt05();
        ej02.ejercicio02();

        EjerciciosUt05 ej03 = new EjerciciosUt05();
        ej03.Ejercicio3();

        EjerciciosUt05 ej04 = new EjerciciosUt05();
        ej04.Ejercicio4();
        */
        EjerciciosUt05 ej05 = new EjerciciosUt05();
        ej05.Ejercicio5();

    }
}
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
```

```
public void Ejercicio5() {

    System.out.println("****RESOLUCIÓN EJERCICIO 5****");
    System.out.println("");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Petición del número y llamada a la función para buscar en qué posición se encuentra
    int posicion = 0;
    int numeroIntroducido = 0;

    Scanner sc = new Scanner(System.in);

    System.out.println("Introduce un número para ver en qué posición está:");
    numeroIntroducido = sc.nextInt();

    posicion = fun05.buscaNumeroArray(arrayEntrada: array10, numero: numeroIntroducido);
}
```

```

        //Salida por pantalla del resultado
        if (posicion == 0) {

            System.out.println("El número introducido no se encuentra en el array");

        } else {

            System.out.println("El número " + numeroIntroducido + " se encuentra en la posición "
                               + posicion + " del array.");

        }

    }

}

```

```

public int buscaNumeroArray(int arrayEntrada[], int numero) {

    int posicion = 0;

    for (int i = 0; i < arrayEntrada.length; i++) {

        if (arrayEntrada[i] == numero) {
            posicion = i + 1;
            break;
        }

    }

    return posicion;

}

```

```

****RESOLUCIÓN EJERCICIO 5****

Introduce un número para ver en qué posición está:
3
El número introducido no se encuentra en el array
-----
BUILD SUCCESS
-----
Total time:  3.299 s
Finished at: 2022-11-26T13:59:51+01:00

```

```

****RESOLUCIÓN EJERCICIO 5****

Introduce un número para ver en qué posición está:
128
El número 128 se encuentra en la posición 8 del array.
-----
BUILD SUCCESS
-----
Total time:  6.532 s
Finished at: 2022-11-26T14:00:34+01:00

```

Ejercicio A06

Implementa y utiliza un método que reciba un array de 10 valores de tipo decimal que pueda tener valores positivos y negativos. Debe mostrar por pantalla el total de la suma de los valores que son positivos y la suma de los valores que son negativos (que será un valor negativo)

```
EjerciciosUt05 ej06 = new EjerciciosUt05();
ej06.Ejercicio6();
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
double arrayD10[] = {-5.5, 56.2, -15.3, -5.1, 125.8, 200.2, -68.9, 87.1, -3.2, 0.5};
```

```
public void Ejercicio6() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 6****");
    System.out.println(x: "");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Llamada al método
    fun05.sumaPositivosNegativos(arrayEntrada: arrayD10);

}
```

```
public void sumaPositivosNegativos(double arrayEntrada[]) {

    double sumaPos = 0.0;
    double sumaNeg = 0.0;

    for (int i = 0; i < arrayEntrada.length; i++) {

        if (arrayEntrada[i] >= 0) {

            sumaPos += arrayEntrada[i];

        } else {

            sumaNeg += arrayEntrada[i];

        }

    }

    //Salida de los resultados por pantalla
    System.out.println("La suma del total de positivos es: " + sumaPos);
    System.out.println("La suma del total de negativos es: " + sumaNeg);

}
```

```

****RESOLUCIÓN EJERCICIO 6****

La suma del total de positivos es: 469.79999999999995
La suma del total de negativos es: -98.00000000000001

-----
BUILD SUCCESS
-----
Total time: 1.829 s
Finished at: 2022-11-26T16:31:19+01:00
-----

```

Ejercicio A07

Implementa y utiliza un método que recibe tres enteros. El primer entero va a indicar el número de elementos (enteros) que va a devolver el array. Los valores de estos elementos se deben calcular aleatoriamente entre un mínimo y un máximo que serán el segundo y tercer parámetro que reciba el método. El array conseguido debe ser devuelto por el método.

```

EjerciciosUt05 ej07 = new EjerciciosUt05();
ej07.Ejercicio7();

```

```

public void Ejercicio7() {

    System.out.println("****RESOLUCIÓN EJERCICIO 7****");
    System.out.println("");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    //Petición del número y llamada a la función para buscar en qué posición se encuentra
    int numElementos = 0;
    int valorMax = 0;
    int valorMin = 0;

    Scanner sc = new Scanner(System.in);

    System.out.println("Introduce el número de elementos del array:");
    numElementos = sc.nextInt();

    System.out.println("Introduce el valor máximo:");
    valorMax = sc.nextInt();

    System.out.println("Introduce el valor mínimo:");
    valorMin = sc.nextInt();
}

```

```

//Salida por pantalla del resultado
if (numElementos <= 0) {

    System.out.println("El número de elementos debe ser mayor de 0");

} else {

    int arraySalida[] = fun05.creaArray(numElementos, valorMax, valorMin);

    //Muestra el array creado por pantalla
    System.out.println("El array creado es el siguiente:");
    fun05.imprimeArray(arrayEntrada: arraySalida);

}

}

```

```

public int[] creaArray(int numElementos, int valorMax, int valorMin) {

    //Creación array de longitud variable dependiendo del valor "numElementos"
    int arraySalida[] = new int[numElementos];

    /*Formación del array con valores aleatorios máximos y mínimos preestablecidos
    en valorMax y valorMin*/
    for (int i = 0; i < arraySalida.length; i++) {

        arraySalida[i] = (int) (Math.random() * ((valorMax - valorMin) + 1)) + valorMin;

    }

    return arraySalida;

}

```

****RESOLUCIÓN EJERCICIO 7****

Introduce el número de elementos del array:

8

Introduce el valor máximo:

56

Introduce el valor mínimo:

-8

El array creado es el siguiente:

-7

45

39

54

18

52

-4

12

BUILD SUCCESS

Total time: 13.314 s

Finished at: 2022-11-26T17:04:33+01:00


```

****RESOLUCIÓN EJERCICIO 7****

Introduce el número de elementos del array:
0
Introduce el valor máximo:
6
Introduce el valor mínimo:
-5
El número de elementos debe ser mayor de 0
-----
BUILD SUCCESS
-----
Total time: 9.134 s
Finished at: 2022-11-26T17:05:14+01:00
-----

```

Ejercicio A08

Implementa un método que reciba dos arrays de enteros de la misma longitud y devuelva un array que será el resultante de sumar que se encuentran en el mismo índice de cada array. Por ejemplo, si el primer array tiene los valores 3, 4, 1, 2 y 4 y el segundo array tiene los valores 2, 5, 1, 1 y 3, devolverá el array 5, 9, 2, 3 y 7.

```

EjerciciosUt05 ej08 = new EjerciciosUt05();
ej08.Ejercicio8();

```

```

public void Ejercicio8() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 8****");
    System.out.println(x: "");

    int primerArray[] = array10;
    int segundoArray[] = array210;
    int resultado[] = new int[primerArray.length];

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    if (primerArray.length != segundoArray.length) {

        System.out.println(x: "Los arrays deben de tener el mismo número de elementos");

    } else {

        //Llamada al método
        resultado = fun05.sumaArray(primerArray, segundoArray);

        //Salida por pantalla del array resultado
        System.out.println(x: "El resultado de la suma es:");
        fun05.imprimeArray( arrayEntrada: resultado);

    }

}

```

```

public int[] sumaArray(int[] primerArray, int[] segundoArray){

    int sumaArray[] = new int[primerArray.length];

    for (int i = 0; i < primerArray.length; i++) {

        sumaArray[i] = primerArray[i] + segundoArray[i];

    }

    return sumaArray;

}

```

```

//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
int array210[] = {-5, -8, 10, 52, 100, -10, 18, -100, 2, 0};
double arrayD10[] = {-5.5, 56.2, -15.3, -5.1, 125.8, 200.2, -68.9, 87.1, -3.2, 0.5};

```

****RESOLUCIÓN EJERCICIO 8****

El resultado de la suma es:

```

-7
-4
16
50
210
-10
32
28
130
20

```

BUILD SUCCESS

Total time: 0.950 s

Finished at: 2022-11-26T17:29:17+01:00

```

//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {2, 4, 6, 8, 10};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
int array210[] = {-5, -8, 10, 52, 100, -10, 18, -100, 2}; //Le hemos quitado un elemento
double arrayD10[] = {-5.5, 56.2, -15.3, -5.1, 125.8, 200.2, -68.9, 87.1, -3.2, 0.5};

```

```

****RESOLUCIÓN EJERCICIO 8****

Los arrays deben de tener el mismo número de elementos

-----
BUILD SUCCESS
-----
Total time: 1.000 s
Finished at: 2022-11-26T17:31:25+01:00
-----

```

Ejercicio A09

Implementa un método que pida al usuario 10 números enteros (no es necesario hacerle llegar los datos, se piden en el método) y los devuelva en un array ordenados de mayor a menor.

```

EjerciciosUt05 ej09 = new EjerciciosUt05();
ej09.Ejercicio9();

```

```

public void Ejercicio9() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 9****");
    System.out.println(x: "");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    int arrayOrdenado[] = fun05.ordenaMayorMenor();

    //Salida por pantalla del resultado
    System.out.println(x: "El array introducido ordenado de mayor a menor sería el siguiente:");
    fun05.imprimeArray( arrayEntrada: arrayOrdenado);

}

```

```

public int[] ordenaMayorMenor() {

    //Creación de la variable Scanner para introducir datos por pantalla
    Scanner sc = new Scanner( source: System.in);

    int arrayEntrada[] = new int[10];
    int arraySalida[] = new int[10];

    //Petición de los datos para rellenar el array
    for (int i = 0; i < arrayEntrada.length; i++) {

        System.out.println("Introduce el elemento número " + (i + 1) + " del array");
        arrayEntrada[i] = sc.nextInt();

    }

    Arrays.sort( a: arrayEntrada);

    int j = (arrayEntrada.length) - 1;

    for (int i = 0; i < arrayEntrada.length; i++) {

        arraySalida[i] = arrayEntrada[j];
        j--;

    }

    return arraySalida;
}

```

****RESOLUCIÓN EJERCICIO 9****

```

Introduce el elemento número 1 del array
0
Introduce el elemento número 2 del array
56
Introduce el elemento número 3 del array
-5
Introduce el elemento número 4 del array
68
Introduce el elemento número 5 del array
97
Introduce el elemento número 6 del array
-336
Introduce el elemento número 7 del array
5
Introduce el elemento número 8 del array
10
Introduce el elemento número 9 del array
-98
Introduce el elemento número 10 del array
254

```

```
El array introducido ordenado de mayor a menor sería el siguiente:
```

```
254
97
68
56
10
5
0
-5
-98
-336
```

```
-----
BUILD SUCCESS
-----
```

```
Total time: 32.948 s
```

```
Finished at: 2022-11-26T19:06:55+01:00
-----
```

Ejercicio A10

Implementa y utiliza un método que reciba un array de números enteros y devuelva un array que indique Par o Impar según el valor en el índice correspondiente sea un número par o impar. Por ejemplo, si recibe el array 3, 4, 4, 1, 5 devolverá el array Impar, Par, Par, Impar, Impar.

```
EjerciciosUt05 ej10 = new EjerciciosUt05();
ej10.Ejercicio10();
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {-2, 0, 7, 8, 11};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
int array210[] = {-5, -8, 10, 52, 100, -10, 18, -100, 2, 0};
double arrayD10[] = {-5.5, 56.2, -15.3, -5.1, 125.8, 200.2, -68.9, 87.1, -3.2, 0.5};
```

```
public void Ejercicio10() {
    System.out.println("****RESOLUCIÓN EJERCICIO 10****");
    System.out.println("");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    String arrayParImpar[] = fun05.parImpar(arrayEntrada: array5);

    //Salida por pantalla del resultado
    System.out.println("El contenido del array introducido como Par o Impar sería el siguiente");

    for (int i = 0; i < arrayParImpar.length; i++) {
        System.out.println(arrayParImpar[i]);
    }
}
```

```
public String[] parImpar(int arrayEntrada[]) {  
    String salida[] = new String[arrayEntrada.length];  
    for (int i = 0; i < arrayEntrada.length; i++) {  
        if (arrayEntrada[i] % 2 == 0) {  
            salida[i] = "Par";  
        } else {  
            salida[i] = "Impar";  
        }  
    }  
    return salida;  
}
```

****RESOLUCIÓN EJERCICIO 10****

El contenido del array introducido como Par o Impar sería el siguiente

Par

Par

Impar

Par

Impar

BUILD SUCCESS

Total time: 0.994 s

Finished at: 2022-11-28T16:16:48+01:00

Ejercicio A11

Implementa y utiliza un método que reciba dos arrays de 10 números enteros e indique el número de números en el que coincide (en el mismo índice). Por ejemplo, si recibe como primer array 1, 3, 5, 7, 9, 9, 1, 3, 2 y 5 y como segundo array 3, 3, 6, 8, 9, 2, 1, 4, 4 y 6 devolverá 3 ya que coincide en el valor del índice [1], [4] y [6].

```
EjerciciosUt05 ej11 = new EjerciciosUt05();
ej11.Ejercicioll();
```

```
//Creación de arrays que se pueden utilizar a lo largo de la tarea
int array5[] = {-2, 0, 7, 8, 11};
int array10[] = {-2, 4, 6, -2, 110, 0, 14, 128, 128, 20};
int array210[] = {-5, 4, 10, 52, 110, -10, 18, -100, 2, 0};
double arrayD10[] = {-5.5, 56.2, -15.3, -5.1, 125.8, 200.2, -68.9, 87.1, -3.2, 0.5};
```

```
public void Ejercicioll() {

    System.out.println(x: "****RESOLUCIÓN EJERCICIO 11****");
    System.out.println(x: "");

    //Creación de la variable necesaria para llamar a la función
    FuncionesUt05 fun05 = new FuncionesUt05();

    int array1[] = array5;
    int array2[] = array210;

    //Salida por pantalla del resultado
    if (fun05.comparaArray(array1, array2) < 0) {

        System.out.println(x: "Los arrays deben de ser de la misma dimensión");

    } else {

        System.out.println("Los arrays introducidos coinciden en "
            + fun05.comparaArray(array1, array2) + " índices");

    }

}
```

```

public int comparaArray(int array1[], int array2[]) {

    int vecesRepite = 0;

    if (array1.length != array2.length) {

        vecesRepite = -1;

    } else {

        for (int i = 0; i < array1.length; i++) {

            if (array1[i] == array2[i]) {

                vecesRepite++;

            }

        }

    }

    return vecesRepite;

}

```

****RESOLUCIÓN EJERCICIO 11****

Los arrays introducidos coinciden en 2 índices

BUILD SUCCESS

Total time: 0.966 s

Finished at: 2022-11-28T16:32:06+01:00

****RESOLUCIÓN EJERCICIO 11****

Los arrays deben de ser de la misma dimensión

BUILD SUCCESS

Total time: 0.949 s

Finished at: 2022-11-28T16:40:30+01:00
