

1. En este ejercicio tienes que hacer dos cosas. Primero implementar una clase `ArrayEnMovimiento` y después los tests JUnit necesarios para chequear que tu código hace lo que se supone debe hacer. `ArrayEnMovimiento` tendrá un solo método testeable con la siguiente signatura

`int[] moverUno(int[] inArray)`

Este método recibe un array de enteros y devuelve un array de la misma longitud, cuyo primer elemento es -1 y que a partir de la segunda posición incluye los valores del array de entrada, excepto el último, y en el mismo orden.

2. Este es muy parecido al anterior. Hay que implementar la clase `ArrayMultiplica` y después los tests JUnit necesarios para chequear que tu código hace lo que se supone debe hacer. `ArrayMultiplica` tendrá un solo método testeable con la siguiente signatura

`int[] multiplica(int[] array1, int[] array2)`

La longitud de los arrays puede ser diferente. Si `minlong` y `maxlong` representan la longitud mínima y máxima de los dos arrays, el array de salida debe tener longitud `maxlong` y

- a. For all  $i < \text{minlong}$ , `outArray[i] = array1[i] * array2[i]`.
- b. For all  $j \geq \text{minlong}$ , `outArray[j] = longArray[j]`.

3. En este ejercicio tendrás que crear la Plantilla de Empleados de la Universidad que implemente los siguientes métodos con las siguientes signaturas
  - a. `ArrayList<Empleados> getEmpleados1825 ()` que filtra los empleados cuya edad es  $>18$  y  $<25$ . La clave corresponde al DNI del empleado.
  - b. `Empleado getEmpleadoPorDNI(String DNI)` que devuelve el empleado cuyo DNI se recibe como argumento.
  - c. `String getEmpleadoDepartamento(String DNI)` devuelve el nombre del departamento en el que trabaja el empleado.
  - d. `ArrayList<Empleado> getEmpleadosPorEdadSexo(int minEdad, String sexo)`. Que devuelve los empleados con edad mayor que `minEdad` y sexo el indicado.

Cada empleado está asociado a un departamento que viene dado por el dominio del correo electrónico.

Esos serán los datos de los empleados con los que trabajaremos.

DNI	Nombre	Correo	Edad	Sexo
s0189034	Pedro	pedro@mate	17	Hombre
s0289125	Miguel	migul@geo	21	Hombre
s0378435	Elena	elena@fisic	28	Mujer
s0412375	Maria	maria@inf	18	Mujer
s0456782	Juan	juan@inf	22	Hombre
s0355689	Diana	diana@leng	33	Mujer
s0768633	Luis	luis@quim	36	Hombre

Dominio	Departamento
quim	Química
geo	Geografía
inf	Informática
leng	Lengua
mate	Matemáticas
fisic	Física

Implementa los tests necesarios para chequear que los métodos se comportan como deberían.