

# Fundamentos de Programación

## Hoja de Ejercicios 7a

### Arrays unidimensionales (Vectores)

1. Elaborar un método para normalizar un vector de números reales, con el criterio de que el nuevo vector resultante tenga módulo 1 (es decir, con la misma dirección pero tamaño 1).
2. Elaborar un método para normalizar un vector de números reales, con el criterio de que el nuevo vector resultante tenga como mayor componente una con módulo 1.
3. Sea A un vector de 10 números reales no nulos. Se desea dividir todos sus elementos por el elemento que tiene índice k. ¿Es correcto el siguiente método para conseguirlo?

```
final int TAM = 554;
int j, k;
double [ ] a = new double [TAM];
/* leer k, entre 0 y TAM-1 */
/* leer a */
for (int i = 0 ; i < TAM ; i++)
    a[i] = a[i]/a[k] ;
/ * escribir a */
```

4. Para operar con números enteros de tamaño grande se puede utilizar un vector cuyas componentes sean dígitos decimales (entre 0 y 9, ambos inclusive). Elaborar un subprograma para sumar números enteros de hasta cuarenta cifras mediante dicho procedimiento.
5. Supongamos que una determinada colección consta de 225 cromos que se venden en sobres conteniendo cada uno 6 cromos distintos entre sí. Elaborar un programa que simule la compra de sobres mostrando el estado de la colección en cada momento y finalmente el número de sobres que han sido necesarios para completar la colección una vez conseguida.
6. Mostrar de qué forma puede representarse un conjunto de números enteros de valor comprendido entre 1 y n mediante un vector de componentes booleanas. Escribir un subprograma que lea dos de esos conjuntos y calcule su unión y su intersección.
7. Se dispone de un vector, parcialmente ocupado, que almacena números enteros clasificados en orden creciente. Se desea conocer si un número introducido por teclado se encuentra en el vector, en caso afirmativo se mostrará su posición, y en caso negativo se insertará en la posición que le corresponda. Si se intenta insertar y el vector está lleno, se mostrará un mensaje de error.
8. Sean dos vectores a y b cuyas componentes son de un tipo enumerado, por ejemplo int. Diremos que  $a < b$  si:
  - Existe una posición i para la que la componente de a es menor que la componente correspondiente de b ( $a_i < b_i$ ), siendo las componentes anteriores iguales ( $a_j = b_j$  para  $j < i$ ).
  - Siendo todas las componentes de a iguales a las de b, el número de componentes de a es menor que el de b.

Elaborar un método que devuelva -1, 0 ó 1 dependiendo de si  $a < b$ ,  $a = b$  o  $a > b$ .

9. Escribir un método que reciba un array cuyos elementos son enteros positivos. El método dibuja tantos asteriscos como indique el número almacenado en cada elemento del array. Por ejemplo, el array:

2	3	5	7	4	6	1	3	2	9
---	---	---	---	---	---	---	---	---	---

Escribiría este resultado por la consola:

```
**
***
*****
*****
****
*****
*
***
**
*****
```

10. Crear una función que reciba un número entero positivo y a partir del número obtenga como resultado un array que almacena por separado cada cifra del número.  
Por ejemplo, para la entrada 5492 la función debe retornar el array:

5	4	9	2
---	---	---	---

11. Crear una función que reciba dos arrays de números enteros y devuelva verdadero si el segundo contiene todos los elementos del primero sin importar el orden de los elementos. Por ejemplo, para los arrays siguientes:

Array 1:

18	24	31	47	32	10	9
----	----	----	----	----	----	---

Array 2:

47	32	24	18	9	31	10
----	----	----	----	---	----	----

La función debería devolver verdadero, porque ambos contienen los mismos elementos.

12. Crear un programa que defina un array que almacene 20.000 letras mayúsculas aleatorias (caracteres de la A a la Z sin contar la Ñ). Crear un programa que muestre cada letra y las veces que aparece en el array.
13. Crear un programa que permita simular la generación de un mazo de cartas españolas. En dicho programa se colocarán las 40 cartas de la baraja española (las cartas van del 1 al 7 y luego siguen la sota, el caballo y el rey; de cada uno de los 4 palos). El programa desordenará el array de forma aleatoria (lo cual significará “barajear” las cartas) y luego mostrará en pantalla la colocación actual de la baraja (por supuesto no se puede repetir la misma carta).

Ejemplo de posible salida por consola:

```
Carta 1: 7 de espadas
Carta 2: As de bastos
Carta 3: Caballo de oros
Carta 4: 5 de copas
Carta 5: 2 de copas
...
```

14. Para almacenar enteros grandes, Java nos proporciona el tipo básico **long**. Pero incluso este tipo admite números de hasta 18 cifras como mucho. Queremos trabajar con números enteros mucho más largos. Para ello vamos a hacer un programa que realice estas acciones:
  - a. Crear un array de 1600 números aleatorios del 0 al 9.
  - b. Tras crear el array, aprovecharle para mostrar por pantalla 20 números aleatorios de 80 cifras.
  - c. Finalmente mostrar por pantalla cuál es el número (de 80 cifras) más grande de todos los que han salido.