

## FUNCIONES

1. Crea una biblioteca de funciones para arrays (de una dimensión) de números enteros que contenga las siguientes funciones:
  1. generaArrayInt: Genera un array de tamaño n con números aleatorios cuyo intervalo (mínimo y máximo) se indica como parámetro.
  2. minimoArrayInt: Devuelve el mínimo del array que se pasa como parámetro.
  3. maximoArrayInt: Devuelve el máximo del array que se pasa como parámetro.
  4. mediaArrayInt: Devuelve la media del array que se pasa como parámetro.
  5. estaEnArrayInt: Dice si un número está o no dentro de un array.
  6. posicionEnArray: Busca un número en un array y devuelve la posición (el índice) en la que se encuentra.
  7. volteaArrayInt: Le da la vuelta a un array.
  8. rotaDerechaArrayInt: Rota n posiciones a la derecha los números de un array.
  9. rotalZquierdaArrayInt: Rota n posiciones a la izquierda los números de un array.
2. Crea una biblioteca de funciones para arrays bidimensionales (de dos dimensiones) de números enteros que contenga las siguientes funciones:
  1. generaArrayBilnt: Genera un array de tamaño n x m con números aleatorios cuyo intervalo (mínimo y máximo) se indica como parámetro.
  2. filaDeArrayBilnt: Devuelve la fila i-ésima del array que se pasa como parámetro.
  3. columnaDeArrayBilnt: Devuelve la columna j-ésima del array que se pasa como parámetro.
  4. coordenadasEnArrayBilnt: Devuelve la fila y la columna (en un array con dos elementos) de la primera ocurrencia de un número dentro de un array bidimensional. Si el número no se encuentra en el array, la función devuelve el array {-1, -1}.
  5. esPuntoDeSilla: Dice si un número es o no punto de silla, es decir, mínimo en su fila y máximo en su columna.
  6. diagonal: Devuelve un array que contiene una de las diagonales del array bidimensional que se pasa como parámetro. Se pasan como parámetros fila, columna y dirección. La fila y la columna determinan el número que marcará las dos posibles diagonales dentro del array. La dirección es una cadena de caracteres que puede ser "nose" o "neso". La cadena "nose" indica que se elige la diagonal que va del noroeste hacia el sureste, mientras que la cadena "neso" indica que se elige la diagonal que va del noreste hacia el suroeste.
3. Crea la función de manejo de arrays que tenga la siguiente cabecera y que haga lo que se especifica en los comentarios (puedes incluirla en tu propia biblioteca de rutinas):
 

```
public static int[] filtraPrimos(int x[]).
```

 Devuelve un array con todos los números primos que se encuentren en otro array que se pasa como parámetro. Obviamente el tamaño del array que se devuelve será menor o igual al que se pasa como parámetro.  
 Utiliza esta función en un programa para comprobar que funcionan bien. Para que el ejercicio resulte más fácil, las repeticiones de primos se conservan; es decir, si en el array x el número 13 se repite 3 veces, en el array devuelto también estará repetido 3 veces. Si no existe ningún número primo en x, se devuelve un array con el número -1 como único elemento.
4. Crea una función con la siguiente cabecera: `public String convierteEnPalabras(int n)`

Esta función convierte los dígitos del número n en las correspondientes palabras y lo devuelve todo en una cadena de caracteres. Por ejemplo, el 470213 convertido a palabras sería: cuatro, siete, cero, dos, uno, tres

Utiliza esta función en un programa para comprobar que funciona bien. Desde la función no se debe mostrar nada por pantalla, solo se debe usar print desde el programa principal. Fíjate que hay una coma detrás de cada palabra salvo al final.

5. Crea la función de manejo de arrays que tenga la siguiente cabecera y que haga lo que se especifica en los comentarios (puedes incluirla en tu propia biblioteca de rutinas):

```
public int[] filtraCon7(int x[])
```

Devuelve un array con todos los números que contienen el 7 (por ej. 7, 27, 782) que se encuentren en otro array que se pasa como parámetro. El tamaño del array que se devuelve será menor o igual al que se pasa como parámetro. Utiliza esta función en un programa para comprobar que funcionan bien. Para que el ejercicio resulte más fácil, las repeticiones de números que contienen 7 se conservan; es decir, si en el array x el número 875 se repite 3 veces, en el array devuelto también estará repetido 3 veces. Si no existe ningún número que contiene 7 en el array x, se devuelve un array con el número -1 como único elemento.

6. Implementar un método que permita insertar un elemento entero en un vector ordenado.
7. Implementar un método que permita buscar la primera ocurrencia de un elemento entero en un vector ordenado: algoritmo de la búsqueda dicotómica.
8. Implementar un método que permita buscar la primera ocurrencia de un elemento entero en un vector desordenado: algoritmo de búsqueda secuencial.
9. Implementar un método que permita rotar un vector de 10 números enteros, n posiciones hacia la derecha. El valor n se leerá mediante teclado.
10. Implementar un método que permita rotar un vector de 10 números enteros, n posiciones hacia la izquierda. El valor n se leerá mediante teclado.