# Ejercicios con Arrays (parte 2)

Implementa los siguientes programas:

## Partir un array.

* 1. Crea un array ‘datos’ con 40 números enteros.
  2. Rellénalo con los números consecutivos del 1 al 40.
  3. PARTE el array:  
     1. Escoge una posición aleatoria entre el 1 y el 40.  
     2. Intercambia los datos de la parte a la derecha del array con los de la parte izquierda.  
     Por ejemplo: Si parto el array por la **posición 15** obtendría el siguiente resultado:  
     {16,17,18…40,1,2,3…15}  
     3. Imprime el resultado.

Puedes usar un segundo array para ayudarte con la operación.

## Buscador de secuencias.

* 1. Crea un array de 50 números enteros.
  2. Inicialízalo con valores aleatorios entre 0 y 1.
  3. Imprímelo como si fuera un número binario de 50 dígitos.
  4. Busca la **secuencia más larga de unos** y muestra la información de la siguiente forma:  
     - *“100101001100-111111-01010010100110010010010010001010101001001”  
     - ”He encontrado 6 unos consecutivos a partir de la posición 13”*
  5. Busca la secuencia más larga de repeticiones de ‘10’ (1010101010…) y muestra los resultados de forma similar al apartado anterior

## Ordenar un array con el método de la Burbuja

El algoritmo de la burbuja es uno de los métodos de ordenación más conocidos y uno de los primeros que aprenden los programadores.   
  
Consiste en comparar pares de elementos adyacentes en un array y si están desordenados intercambiarlos hasta que estén todos ordenados.

Pseudocódigo:  
 A : array de n elementos indizados de 1 a n (ojo!! Al programar, es de 0 a n-1)

para i desde 1 hasta n-1 hacer: //las n-1 pasadas

para j desde 1 hasta n-i hacer: //el recorrido

si A[j] > A[j+1] entonces //Si no están en orden

intercambiar A[j] y A[j+1] //Se intercambian

fin para

fin para

Link de interés: <http://latecladeescape.com/t/El+algoritmo+de+la+burbuja+%28BubbleSort%29>

* 1. Crea un array ‘datos’ para 10 números enteros.
  2. Inicialízalo con valores aleatorios entre 1 y 100.
  3. Imprímelo
  4. Ordénalo con el método de la Burbuja
  5. Imprímelo de nuevo (ordenado).
  6. Modifica el código para que imprima el array en cada paso del algoritmo y ver cómo se va ordenando paso a paso.

## Arrays 2D

* 1. Crea un array ‘datos’ de 2 dimensiones 10x10.
  2. Rellénalo con números enteros aleatorios entre 1 y 10.
  3. Imprímelo como una cuadrícula o tabla.

## Arrays 2D (2)

* 1. Crea un array ‘datos de 2 dimensiones (por ejemplo 10x10).
  2. Rellena la primera fila con números enteros aleatorios entre 1 y 10.
  3. Para las filas siguientes:  
     El valor de cada casilla se calcula como la suma de los valores de las casillas superior y superior izquierda.  
     El valor de la primera columna es el de la celda inmediatamente superior.  
     Rellena el array siguiendo esa norma.

|  |  |  |  |
| --- | --- | --- | --- |
| **1** | **2** | **4** | **Ésta fila se rellena de forma aleatoria** |
| 1 | 3 | 6 | Éstas filas se calculan en función de la de arriba.  (3 = 1+2) |
| 1 | 4 | 9 |

* 1. **Triángulo de Pascal:** Inicializa la primera fila con los valores [1,0,0,0,0…0] y rellena el array como en el caso 5.3.  
     Al mostrar los resultados no imprimas los ceros.

