

Cp 7 - Ciclos (for black belts)  
Curso 2023-2024



## Subsecuencia de suma máxima

Implemente un método que dado un array de números enteros, devuelva la suma de la subsecuencia de suma máxima. Si todos los números son negativos devolver 0.

Ejemplo:

array: {1, 1, -3, 4, 2, 2, -1, 2, -3, 2}. Resultado: 9. Porque es la suma de la secuencia 4, 2, 2, -1, 2. Su método deberá tener la siguiente signatura:

```
1 public static int MaxSumSubArray(int[] a)
2 {
3     // ...
4 }
```

## Multiplica polinomios

¿Recuerda la clase Poly? Si no, busque la cp-6. En esa clase implemente el siguiente método para multiplicar polinomios:

```
1 public static Poly Mult(Poly p1, Poly p2)
2 {
3     // ...
4 }
```

## Distancia promedio

En un array se representan pares de enteros que expresan una secuencia de coordenadas  $x, y$  de puntos en el plano. Por ejemplo, la secuencia de los puntos  $\langle 5, 3 \rangle, \langle 2, 8 \rangle, \langle -15, 7 \rangle$  se representan en el array en la forma  $\{5, 3, 2, 8, -15, 7\}$ . Implemente un método que reciba un array de puntos y un punto de coordenadas  $x, y$  y devuelva el promedio de las distancias entre cada punto del array y el punto  $x, y$ . Su método deberá tener la siguiente signatura:

```
1 public static int Average(int[] coordinates, int x, int y)
2 {
3     // ...
4 }
```

**a)**

Devuelva un nuevo array en el que los puntos han quedado ordenados por orden de cercanía al punto  $\langle x, y \rangle$ . Su método deberá tener la siguiente signatura:

```
1 public static int[] ClosenessMerge(int[] coordinates, int x, int y)
2 {
3     // ...
4 }
```

**b)**

Lo mismo que el inciso (a) pero ahora, retornando void y modificando el array original que puntos que recibe como parámetro.

## Anagramas

Un anagrama de una palabra es cualquier otra palabra que tenga las mismas letras que la primera pero en distinto orden. Ejemplos de anagramas: casa y saca, tasa y asta.

a)

Implemente un método que dadas dos palabras determine si son un anagrama:

```
1 public static bool Anagrams (string a, string b)
2 {
3     //...
4 }
```

b)

Implemente un método que dado un grupo de palabras (parámetro palabras de tipo string[]) devuelva la cantidad de palabras del mayor subconjunto de anagramas que se pueda formar:

```
1 public static int LargestSubsetAnagram (string[] words)
2 {
3     //...
4 }
```

Si se invoca a

```
1 LargestSubsetAnagram(new string[] { "amor", "cometa", "roma", "mora",
    "moceta" });
```

el resultado es 3 (por "amor", "roma" y "mora").

## Puntos más cercanos

Dado un array de puntos según el formato del ejercicio anterior, implemente un método que devuelva en un array el par de puntos entre los cuales existe la menor distancia. Asuma que en el array original existen al menos 2 puntos. Si existe más de una posible respuesta dar cualquiera de ellas. Note que la longitud del array resultante será siempre 4. Por ejemplo para un array con los valores

$\{5, 3, 2, 8, -15, 7\}$  la respuesta es el array  $\{5, 3, 2, 8\}$ . La signature del método es:

```
1 public static int[] LessDistance(int[] coordinates)
2 {
3     // ...
4 }
```