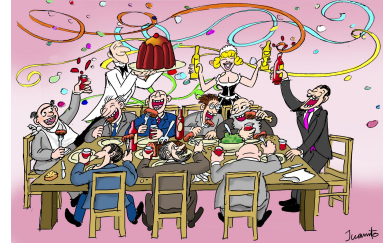


# Cenando con los amigos

Cada año nos reunimos los amigos a cenar unos días antes de Navidad. El año pasado, como casi todos los anteriores, acabamos hablando de lo que comeríamos durante las fiestas y de lo difícil que es mantener el tipo en estos días. Sin embargo, algunos afirman que hay otras épocas peores en el año, como los cocidos en carnaval, o los helados en verano. Al final, hicimos una apuesta. Cada uno apuntaría durante este año los gramos que engordase o adelgazase cada día.



Esta noche nos vamos a reunir de nuevo para cenar y tenemos que analizar los datos para ver quien pierde la apuesta y paga la cena. Para cada secuencia de datos, buscaremos los días consecutivos en los que cada uno engordamos más, es decir, entre que dos días la suma de los pesos es máxima (mayor que entre otros dos días cualquiera). Si existen dos secuencias de igual suma, nos quedaremos con la que tenga menor longitud. Si existen dos secuencias de igual suma e igual longitud nos quedamos con la que ocurre antes en el año (la primera que aparece). De esta forma veremos en que época del año engordamos más cada uno. Como todos los amigos somos grandes comedores, nunca hacemos régimen durante muchos días seguidos, por lo que todos acabamos engordando algo por lo menos un día al año.

## *Requisitos de implementación.*

Implementar una función que reciba en un vector los datos, encuentre la subsecuencia en la cual la suma de valores es máxima y devuelva la suma de esta subsecuencia, su punto de comienzo y su número de días. Cada valor del vector debe tratarse una única vez, y no debe utilizarse ningún vector auxiliar.

Los datos de la secuencia pueden ser positivos o negativos, pero siempre habrá al menos un valor positivo. Ten en cuenta que una subsecuencia deja de ser interesante cuando su suma es cero o negativa, ya que al haber números positivos en el vector estos serán siempre mejores que cualquier valor negativo. Al tratar cada dato, no importa si el dato es positivo o negativo, sino si la suma acumulada teniendo en cuenta ese dato es positiva o negativa. Si la suma acumulada es positiva la subsecuencia sigue siendo interesante, si es cero o negativa debemos cambiar de subsecuencia para no acumular algo negativo.

## Entrada

La entrada consta de una serie de casos de prueba. Cada caso de prueba tiene dos líneas. En la primera se indica el número elementos del vector. En la segunda se indican los valores.

Los valores pueden almacenarse en una variable de tipo int. El tamaño del vector es mayor a igual que uno y se garantiza que siempre hay al menos un valor estrictamente positivo.

## Salida

Para cada caso de prueba se escribe en una línea la suma de la subsecuencia, el día en que comienza y la longitud de la subsecuencia.

### Entrada de ejemplo

```
6 // num elementos del vector
-3 2 1 -1 -3 2
6
2 -1 2 -3 1 -1
7
-1 1 -3 4 1 -1 2
1
4
3
-1 1 -1
3
0 2 3
8
2 3 -6 0 3 2 -1 1
```

### Salida de ejemplo

```
3 1 2
3 0 3
6 3 4
4 0 1
1 1 1
5 1 2
5 0 2
```

**Autor:** Isabel Pita.