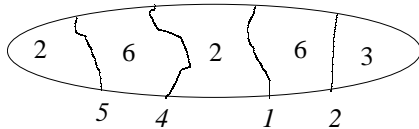


Demoliendo muros

Descripción del problema

En una isla de forma longitudinal existen **n** países ubicados de tal manera que cada uno limita con un vecino a la izquierda y otro a la derecha, salvo los países de los extremos que limitan solo con un país vecino. Cada par de países está dividido por un muro, de un largo dado. Cada país tiene una población dada.



Los países han decidido unificarse, para lo cual deben demoler los muros, de a uno por vez, con el costo mínimo total.

Un experto matemático ha llegado a la curiosa conclusión de que el costo de demoler cada muro se calcula como el largo del muro multiplicado por el cuadrado de la suma de las poblaciones de los países que separa.

$$\text{<costo unificar A y B> = } \text{<largo muro entre A y B> * } (\text{<pobl. de A> + <pobl. de B>})^2$$

Notar que cada vez que se unifican dos países, el nuevo país tiene como población la suma de las poblaciones de los países que se han unificado.

En la situación del diagrama, el costo de demoler el muro 1 (comenzando por la izquierda) es  $320 = 5 * (2 + 6)^2$ , y genera un nuevo país con población igual a 8.

El costo total es la suma de los costos de demoler todos los muros.

Se pide escribir un programa **MUROS.EXE** que determine la secuencia con la que tienen que unificarse los países para que el costo total sea mínimo.

Datos de entrada

Se recibe un archivo **MUROS.IN** del directorio actual, que contiene:

- Primera línea: el número **n** de países ( $2 \leq n \leq 1000$ ) que existen en la isla.
- **n** líneas, cada una con un número  $\leq 1000$  que indica la población del país correspondiente, desde el país 1 hasta el país **n**.
- **n - 1** líneas, cada una con un número  $\leq 100$  correspondiente al largo del muro que separa

cada uno de los países vecinos, desde el muro entre el país 1 y el 2 ("muro 1"), hasta el muro entre el país (n-1) y el país n ("muro n-1").

Datos de salida

El programa debe generar el archivo **MUROS.OUT**, en el directorio actual, con:

- 1 línea con el costo mínimo total
- **n - 1** líneas con un número de muro en c/u; listados en el orden sugerido de demolición para obtener este costo mínimo total.

Ejemplo

En el caso de que el archivo **MUROS.IN** contenga:

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

El archivo **MUROS.OUT** podrá contener:

```
1243
4
1
2
3
```

En esta solución (hay otras) proponen demoler primero el muro 4, luego el muro 1, luego el muro 2 y por último el muro 3, con un costo total mínimo de 1243.

Nota

Obtendrán 50 puntos los programas que solo calculen el costo total mínimo, y 100 puntos los además provean un orden de demolición de muros para dicho costo total.

Se privilegiarán los programas que ejecuten en tiempo más corto.