



HPC³ 2024

Problema B, español

Correo electrónico de poder

Puntos máximos: 15

Eres un cartero que viaja de isla en isla con un nuevo tipo de barco experimental que tiene dos modos diferentes: estándar y eficiente. Tu deber como cartero te obliga a entregar el correo en cada isla dentro del área en la que vives. Esto significa que debes estar presente en cada isla con tu barco al menos una vez.

Su embarcación tiene una cantidad de combustible representada por un número entero que comienza en 0.

A partir de la isla 1, tienes tres métodos para viajar entre islas:

- Si está con su barco, puede viajar con su barco en modo estándar. Esto lleva S ($0 \leq S < 10^3$) combustible y T_S ($0 \leq T_S < 10^3$) minutos y solo se puede hacer si tienes al menos S combustible.
- Si está en su bote, puede viajar con su bote en modo de eficiencia. Esto consume todo su combustible y T_E ($0 \leq T_E < 10^3$) minutos y solo puede hacerse si tiene más de 0 combustible.
- Puedes nadar solo. Esto te llevará T_W ($0 \leq T_W < 10^3$) unos minutos.

Para cada una de las N ($0 \leq N < 10^6$) islas a las que debes viajar, hay F_i ($0 \leq F_i < 10^4$) unidades de combustible en un depósito en esa isla que recoges inmediatamente al estar allí, con o sin tu barco.

Como eres una persona eficiente, quieres entregar todo el correo en el menor tiempo posible.

Subproblema 1

Vives en un archipiélago donde cada una de las islas está en fila. Esto significa que si estás en una isla i solo puedes viajar a islas $i - 1$, $i + 1$.

Se le proporcionan los valores para N , S , T_S , T_E , T_W , y F . Encuentre la cantidad mínima de tiempo que lleva entregar el correo a cada isla.

Formato de entrada

La primera línea de cada entrada contiene 5 números enteros N , S , T_S , T_E , y T_W .

La segunda línea de cada entrada contiene N números enteros: El contenido de la matriz F .

```
N S TS TE TW
F[0] F[1] F[2] ... F[N-1]
```

Formato de salida

La primera y única línea de cada salida contiene 1 entero T .

```
T
```

Donde T es el tiempo mínimo en el que puedes entregar el correo.

Ejemplos de casos de prueba

Entrada 1

```
5 4 2 9 1
1 2 4 2 1
```

Salida 1

```
28
```

La ruta óptima es eficiencia a 2 (2), nadar a 3 (6), nadar a 4 (8), nadar a 3, nadar a 2, estándar a 3 (4), estándar a 4 (0), nadar a 5 (1), nadar a 4 y eficiencia a 5 (0). $9 + 1 + 1 + 1 + 1 + 2 + 2 + 1 + 1 + 9 = 28$. Por lo tanto, el programa debería devolver 28.

Entrada 2

```
5 4 2 1 1
1 2 4 2 1
```

Salida 2

```
4
```

La ruta óptima es eficiencia a 2 (2), eficiencia a 3 (4), eficiencia a 4 (2) y eficiencia a 5. $1 + 1 + 1 + 1 = 4$. Por lo tanto, el programa debería devolver 4.

Subproblema 2

Vives en una región en la que las islas se pueden representar como un gráfico no dirigido. Esto significa que desde una isla determinada, solo puedes viajar a islas que comparten un borde con tu isla.

Se le proporcionan los valores de N , S , T_S , T_E , T_W , y F . Además, se le proporciona una matriz de pares G que denota los bordes entre islas. Si un elemento en G es (a, b) , hay un borde entre las islas a y b . Encuentre la cantidad mínima de tiempo que lleva entregar el correo a cada isla.

Formato de entrada

La primera línea de cada entrada contiene 5 números enteros N , S , T_S , T_E , y T_W .

La segunda línea de cada entrada contiene N números enteros: El contenido de la matriz F .

La tercera línea de cada entrada contiene N pares de números enteros: El contenido de la matriz G .

```
N S T_S T_E T_W
F[0] F[1] F[2] ... F[N-1]
G[0][0] G[0][1] G[1][0] G[1][1] ... G[N][0] G[N][1]
```

Formato de salida

La primera y única línea de cada salida contiene 1 entero T .

```
T
```

Donde T es el tiempo mínimo en el que puedes entregar el correo.

Ejemplos de casos de prueba

Entrada 1

```
3 2 2 10 1
1 0 3
1 2 2 3 1 3
```

Salida 1

```
6
```

La ruta óptima es nadar hasta 3 (4), nadar hasta 1, estándar hasta 3 (2), estándar hasta 2. $1 + 1 + 2 + 2 = 6$. Por lo tanto, el programa debería devolver 6.

Entrada 2

```
7 3 3 8 1
2 4 4 0 2 3 2
1 2 1 3 2 3 2 4 3 4 4 5 4 6 5 7 6 7
```

Salida 2

```
27
```

Una ruta óptima es nadar a 2 (6), nadar a 1, estándar a 2 (3), estándar a 3 (4), estándar a 4 (1), nadar a 6 (4), nadar a 4, estándar a 5 (3), estándar a 7 (2), eficiencia a 6. $1 + 1 + 3 + 3 + 3 + 1 + 1 + 3 + 3 + 8 = 27$. Por lo tanto, el programa debería devolver 27.