HPC³ 2024
Problem B, English
Power Mail
Maximum Points: 15

You are an island-to-island postman with an experimental new type of boat that has two different modes: Standard and efficiency. Your duty as a postman compels you to deliver mail to each island within the area that you live in. This means that you must be present at each island with your boat at least once. Your boat has an amount of fuel represented by an integer that starts at 0. Starting on island 1, you have three methods of traveling between islands:

- If you are with your boat, you can travel with your boat in standard mode. This takes S $(0 \leq S < 10^3)$ fuel and T_S $(0 \leq T_S < 10^3)$ minutes and can only be done if you have at least S fuel.
- If you are with your boat, you can travel with your boat in efficiency mode. This takes all of your fuel and T_E $(0 \leq T_E < 10^3)$ minutes and can only be done if you have more than 0 fuel.
- You can swim alone. This takes T_W $(0 \leq T_W < 10^3)$ minutes.

For each of the N $(0 \leq N < 10^6)$ islands that you must travel to, there are F_i $(0 \leq F_i < 10^4)$ units of fuel in a deposit on that island that you collect immediately upon being there, with or without your boat.
Since you are an efficient person, you want to deliver all of the mail in the smallest amount of time.

Subproblem 1

You live in an archipelago where each of the islands are in a row. This means that if you are at island i you can only travel to islands $i-1,\ i+1$.

You are given values for $N,\ S,\ T_S,\ T_E,\ T_W$, and F. Find the minimum amount of time it takes to deliver mail to each island.

Input format

The first line of each input contains 5 integers N, S, T_S, T_E, and T_W.
The second line of each input contains N integers: The content of array F.

```
N  S  Ts  TE  TW
F[0]  F[1]  F[2]  …  F[N-1]
```

Output format

The first and only line of each output contains 1 integer T.

```
T
```

Where T is the smallest amount of time that you can deliver the mail in.

Example Test Cases

Input 1

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

Output 1

```
28
```

The optimal path is efficiency to 2 (2), swim to 3 (6), swim to 4 (8), swim to 3, swim to 2, standard to 3 (4), standard to 4 (0), swim to 5 (1), swim to 4, and efficiency to 5 (0). 9 + 1 + 1 + 1 + 1 + 2 + 2 + 1 + 1 + 9 = 28. So, the program should return 28.

Input 2

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

Output 2

```
4
```

The optimal path is efficiency to 2 (2), efficiency to 3 (4), efficiency to 4 (2), and efficiency to 5. 1 + 1 + 1 + 1 = 4. So, the program should return 4.

Subproblem 2

You live in a region where the islands can be represented as an undirected graph. This means that from a given island, you can only travel to islands that share an edge with your island.

You are given values for N, S, T_S, T_E, T_W, and F. Additionally, you are given an array of pairs G that denotes the edges between islands. If an element in G is (a, b), there is an edge between islands a and b. Find the minimum amount of time it takes to deliver mail to each island.

Input format

The first line of each input contains 5 integers N, S, T_S, T_E, and T_W.
The second line of each input contains N integers: The content of array F.
The third line of each input contains N integer pairs: The content of array G.

```
N  S  Tₛ  Tₑ  Tᵥᵥ
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]
```

Output format

The first and only line of each output contains 1 integer T.

```
T
```

Where T is the smallest amount of time that you can deliver the mail in.

Example Test Cases

Input 1

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

Output 1

```
28
```

The optimal path is swim to 3 (4), swim to 1, standard to 3 (2), standard to 2.
1 + 1 + 2 + 2 = 6. So, the program should return 6.

Input 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
```

Output 2

```
27
```

An optimal path is swim to 2 (6), swim to 1, standard to 2 (3), standard to 3 (4), standard to 4 (1), swim to 6 (4), swim to 4, standard to 5 (3), standard to 7 (2), efficiency to 6. 1 + 1 + 3 + 3 + 3 + 1 + 1 + 3 + 3 + 8 = 27. So, the program should return 27.