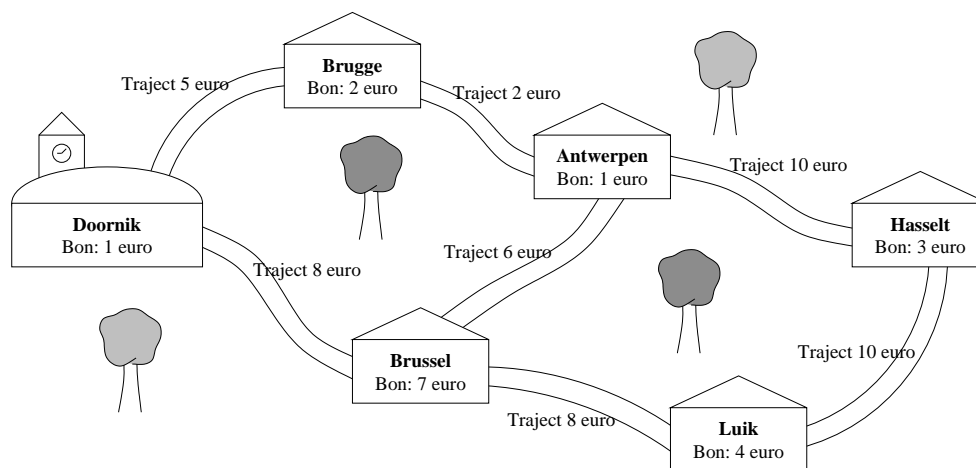


Taak 1.2 – Treinen (Trains) – (100 pts)

Naar aanleiding van een massale staking van de spoorwegen besluit de NMBS zijn imago wat op te smukken met kortingsbonnen waarmee je de kosten van een traject sterk kan verminderen. Deze bonnen worden uitgedeeld in het station en, hoewel niet cumuleerbaar, kunnen zoveel keren als men wil gebruikt worden nadat ze zijn opgehaald. Omwille van onenigheden tussen de vakbond, de regio en de lokale gemeenschap, kunnen deze bonnen van station tot station verschillen: het station van Antwerpen kan perfect bonnen van 1 euro uitdelen, terwijl het station van Brussel er van 7 euro uitdeelt.

Deze opgehaalde bonnen kunnen gebruikt worden op elke sectie van het traject, volgend op de ophaling, maar zonder negatieve kosten te krijgen. De bon van 7 euro die in Brussel wordt uitgedeeld kan dus gebruikt worden op het traject Brussel-Leuven (dat 4 euro kost), maar dan kost dat traject 0 euro, men gaat je uiteraard geen 3 euro uitbetalen.

Bijvoorbeeld, er van uitgaande dat je van Doornik naar Hasselt wil reizen.



De meest directe route lijkt langs Brugge en Antwerpen te gaan. Je vertrekt in Doornik, waar je een bon van 1 euro ophaalt, die je gebruikt op de eerste sectie naar Brugge, die je dus $5 - 1 = 4$ euro kost. In Brugge neem je de bon van 2 euro en gebruik je die om gratis naar Antwerpen te reizen. Je hebt er niets aan om de bon in Antwerpen op te halen, gezien je nog steeds de bon van 2 euro hebt die je kan gebruiken op het laatste traject dat je bijgevolg $10 - 2 = 8$ euro kost. De totale kost van het traject wordt dus $4 + 0 + 8 = 12$ euro.

Veronderstel nu dat je langs Brussel gaat om de bon van 7 euro op te halen. Doornik-Brussel kost je 7 euro. Je reist vervolgens gratis verder naar Antwerpen en uiteindelijk ga je naar Hasselt voor 3 euro. Het totale traject kost nu dus 10 euro. Je kan zien dat dit de goedkoopste complete reis is die je kan maken.

Taak

Jouw programma moet betalen wat de minimale kost is om tussen twee steden te reizen.

Limieten en beperkingen

- $1 \leq n \leq N_{MAX}$, Het aantal stations
- $1 \leq s, f \leq n$, De nummer van het station van vertrek en van aankomst
- $0 \leq d_i \leq D_{MAX}$, De waarde van de bon beschikbaar in station i
- $1 \leq c_{i,j} \leq C_{MAX}$, De kost van het traject tussen station i en j , ongeacht de richting en voor het toepassen van eender welke korting.

	N_{MAX}	D_{MAX}	C_{MAX}	
Subtaak A (32 pts)	50	1 000 000	1 000 000	Elke bon heeft dezelfde waarde.
Subtaak B (17 pts)	50	1 000 000	1 000 000	
Subtaak C (51 pts)	200	1 000 000	1 000 000	

Maximale uitvoeringsduur: **0.5 seconden**. Geheugenlimiet: **128 MB**.

Input

De input die aan je programma gegeven wordt heeft het volgende formaat:

- De eerste regel bevat een getal n : het aantal steden in het spoorwegnetwerk.
- De tweede regel bevat twee getallen s en f , met een spatie gescheiden. s is het nummer van het station van vertrek. f is het station waar je wil aankomen.
- De derde regel bevat n gehele getallen $d_1 d_2 \dots d_n$ gescheiden door spaties.
- De vierde regel bevat het getal k , het aantal verbindingen tussen twee steden.
- De k volgende regels bevatten drie getallen in de vorm $x y c$ die aangeeft dat er een verbinding is tussen station x en station y die c kost, ongeacht de richting. Er zijn nooit meerdere verbindingen tussen twee steden en x is altijd kleiner dan y ($1 \leq x < y \leq n$).
- De input eindigt met een nieuwe lijn.

Het is gegarandeerd dat je altijd station f kan bereiken vanaf s .

Output

De output bestaat uit één regel die een enkel getal bevat, de minimale kost tussen s en f . de output eindigt met een nieuwe regel.

Voorbeeld (cfr bovenstaand schema)

Voor de volgende invoer

```

6
1 6
1 2 7 1 4 3
7
1 2 5
1 3 8
2 4 2
3 4 6
3 5 8
4 6 10
5 6 10
    
```

moet jouw programma het volgende teruggeven:

```

10
    
```