

PIZZAKOERIER

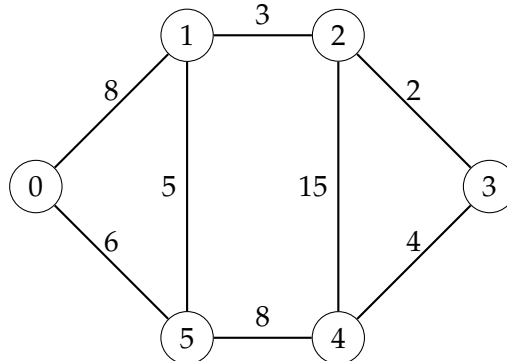


Opgave

In een stad zijn er meerdere pizzeria's: zij posten de bestellingen op het net en jij kan als zelfstandig pizzakoerier kiezen welke leveringen je voor je rekening wenst te nemen. Uiteraard wil je je keuze zo te maken dat je je winst maximaliseert. Daarvoor gebruik je een app waarvan nu een stukje geïmplementeerd moet worden ...

De stad kan je je voorstellen als een gewogen bidirectionele graaf.

- Elke locatie (klant of pizzeria) komt overeen met een node en krijgt een uniek geheel getal als identificatiecode.
- Wegen tussen locaties worden voorgesteld door bogen.
- Het gewicht van een boog tussen nodes X en Y stelt de tijd voor die je nodig hebt om je te verplaatsen van X naar Y of van Y naar X : er is geen sprake van eenrichtingsverkeer.



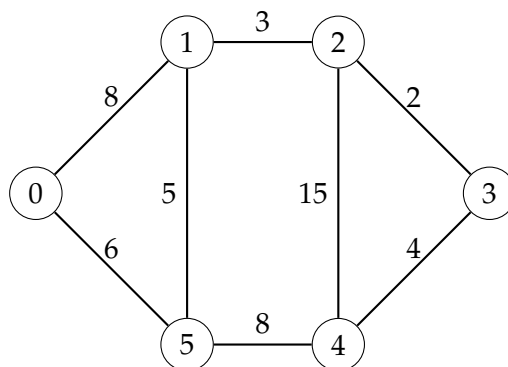
Klanten geven hun bestelling op tijd in. Een bestelling wordt voorgesteld door een tuple (a, b, t, w) :

- a is de locatie van de pizzeria waar de klant pizza('s) heeft besteld.
- b is het leveringsadres.
- t is het tijdstip waarop de pizza exact moet geleverd worden. Pizza's mogen te vroeg noch te laat geleverd worden.
- w is de winst die je maakt voor de levering.

Je begint elke werkschift thuis, voor het gemak aangeduid als locatie 0 en tijdstip 0. Je bent vrij te kiezen welke leveringen je uitvoert: de overblijvende leveringen zullen wel door een andere koerier gedaan worden - of niet, who cares.

Je kan telkens maar één bestelling per keer afhandelen. Het is dus niet toegelaten om twee of meer verschillende bestellingen op te halen (in dezelfde of meerdere pizzeria's) en deze dan te leveren aan twee of meer klanten. M.a.w. je rijdt altijd alternerend tussen pizzeria en klant.

Voorbeeld



Beschouw de bestellingen $(2, 4, 20, 10)$ en $(5, 4, 35, 15)$. We noteren je huidige toestand als (ℓ, t) , waarbij ℓ je locatie en t het huidige tijdstip voorstelt.

Stel, je voert de eerste bestelling uit: je begint op $(\ell = 0, t = 0)$, je rijdt $(0, 0) \rightarrow (1, 8) \rightarrow (2, 8 + 3 = 11)$. Je weet dat je op locatie 4 moet zijn tegen tijdstip 20, wat enkel mogelijk is indien je $2 \rightarrow 3 \rightarrow 4$ rijdt. Deze weg duurt 6 minuten, dus je wacht tot $t = 14$ om de bestelling op te pikken, en je rijdt $(2, 14) \rightarrow (3, 16) \rightarrow (4, 20)$ waar je de pizza's aan de gelukkige klant overhandigt. Je hebt nu €10 winst gemaakt.

Je kan echter de tweede bestelling niet meer tijdig afhandelen: het is reeds $t = 20$ en je moet $4 \rightarrow 5 \rightarrow 4$ rijden, wat je minimaal 16 tijdseenheden kost. Je kan de tweede bestelling onmogelijk geleverd krijgen voor $t = 36$ terwijl de klant wenst deze om $t = 35$ te ontvangen.

De optimale aanpak bestaat er dus in om de eerste bestelling aan een andere koerier over te laten: je rijdt $(0, 0) \rightarrow (5, 6)$, wacht tot $t = 27$, je pikt de pizza's op, je rijdt $(5, 27) \rightarrow (4, 35)$ en je verdient €15.

Invoer

VOORBEELDINVOER

```

1
6 8
0 1 8
1 2 3
2 3 2
3 4 4
4 5 8
0 5 6
1 5 5
2 4 15
2
2 4 20 10
5 4 35 15

```

De eerste regel bevat een positief geheel getal dat het aantal testgevallen voorstelt. Per testgeval volgt hierop:

- Een eerste regel bevat twee door één spatie gescheiden gehele getallen N en M . N stelt het aantal nodes en M het aantal bogen voor van de stadsgrafe.
- Hierop volgen M regels met op elke regel drie door één spatie gescheiden gehele getallen x , y en d , met $0 \leq x < y < N$ en $0 \leq d$. Deze regel betekent "Je kan je verplaatsen tussen x en y in d tijdseenheden."
- Vervolgens is er een regel met een positief geheel getal K . Dit stelt het aantal bestellingen voor.
- Ten slotte volgen er K regels met op elke regel vier door één spatie gescheiden gehele getallen a , b , t en w , met $0 \leq a < N$, $0 \leq b < N$, $0 \leq t$ en $0 \leq w$. Deze regel betekent "Een klant heeft bij pizzeria op locatie a een bestelling geplaatst die moet geleverd worden op locatie b op tijdstip t . Je maakt er w winst op."

Uitvoer

Per testgeval voer je één regel uit met twee gehele getallen gescheiden door één spatie. Het eerste getal is de index van het testgeval. Het eerste testgeval heeft index 1. Het tweede getal stelt de maximale winst voor.

VOORBEELDUITVOER

```

1 15

```
