

Szablon rozwiązania:

kol3a.py

Złożoność akceptowalna (1.5pkt):

$O(n^2)$, gdzie n to liczba planet.

Złożoność wzorcowa (+2.5pkt):

$O(m \log n)$, gdzie m to długość listy E a n to liczba planet.

Układ planetarny Algon składa się z n planet o numerach od 0 do $n - 1$. Niestety własności fizyczne układu powodują, że nie da się łatwo przelecieć między dowolnymi dwiema planetami. Na szczęście mozolna eksploracja kosmosu doprowadziła do stworzenia listy E dopuszczalnych bezpośrednich przelotów. Każdy element listy E to trójka postaci (u, v, t) , gdzie u i v to numery planet (można założyć, że $u < v$) a t to czas podróży między nimi (przelot z u do v trwa tyle samo co z v do u). Dodatkową nietypową własnością układu Algon jest to, że niektóre planety znajdują się w okolicy osobliwości. Znajdując się przy takiej planecie możliwe jest zagięcie czasoprzestrzeni umożliwiające przedostanie się do dowolnej innej planety leżącej przy osobliwości w czasie zerowym.

Zadanie polega na zaimplementowaniu funkcji:

```
def spacetravel( n, E, S, a, b )
```

która zwraca najkrótszy czas podróży z planety a do planety b , mając do dyspozycji listę możliwych bezpośrednich przelotów E oraz listę S planet znajdujących się koło osobliwości. Jeśli trasa nie istnieje, to funkcja powinna zwrócić `None`.

Rozważmy następujące dane:

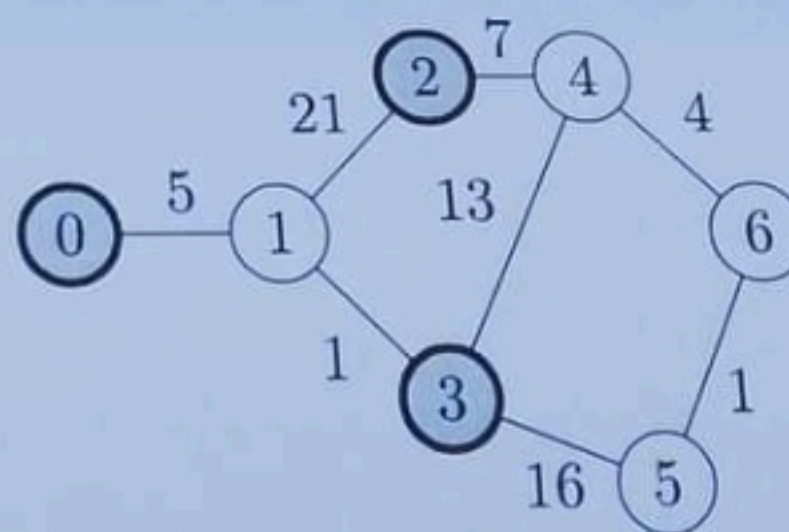
```
E = [(0,1, 5),  
      (1,2,21),  
      (1,3, 1),  
      (2,4, 7),  
      (3,4,13),  
      (3,5,16),  
      (4,6, 4),  
      (5,6, 1)]
```

```
S = [ 0, 2, 3 ]
```

```
a = 1
```

```
b = 5
```

```
n = 7
```



wywołanie `startravel(n, E, S, a, b)` powinno zwrócić liczbę 13. Odwiedzamy po kolei planety 1, 3, 2, 4, 6 i kończymy na planecie 5 (z planety 2 do 3 dostajemy się przez zagięcie czasoprzestrzeni). Gdyby $a = 1$ a $b = 2$ to wynikiem byłby czas przelotu 1.

Podpowiedź. Ile zagięć czasoprzestrzeni warto maksymalnie rozważać? A ile minimalnie?