

Szablon rozwiązania

egzP4a.py

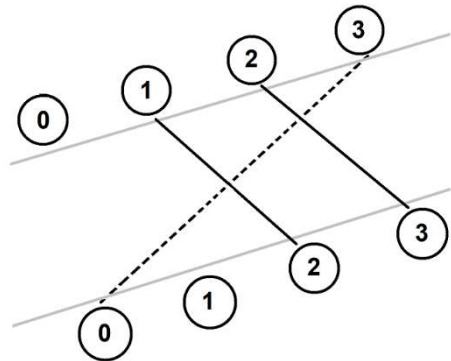
Złożoność akceptowalna (1.5pkt):

$O(n^2)$

Złożoność wzorcowa (+2.5pkt):

$O(n \log n)$, gdzie n to liczba połączeń w tablicy T

Po dwóch stronach pewnej rzeki znajdują się różne miasta. Zarówno na brzegu północnym, jak i południowym, jest ich tyle samo. Projekt budżetowy uchwalony na najbliższy rok zakłada zbudowanie pewnej liczby mostów łączących miasta po przeciwnych stronach rzeki. Jako, że projekt jest finansowany ze środków Unii Europejskiej, rządowi zależy, aby mostów zbudować jak najwięcej. Grupa ekspertów składająca się z inżynierów budowlanych wytypowała pewne połączenia, uznane za możliwe do zrealizowania. Oczywiście nie wszystkie z nich można wybudować, ponieważ żadne dwa mosty nie mogą się ze sobą przecinać (wykluczamy wszystkie sytuacje, w których jeden most znajdowałby się nad innym tj. nie mogą się przecinać także po rzutowaniu na mapę 2D). Wszystkie mosty muszą być również idealnie proste.



W ramach zadania należy zaimplementować funkcję:

```
def mosty( T )
```

która oblicza liczbę mostów, które zostaną wybudowane w ramach projektu budżetowego.

1. Tablica T zawiera listę potencjalnych mostów, wyrażoną w postaci krotek (u, v) gdzie u jest indeksem miasta na brzegu północnym, a v jest indeksem miasta na brzegu południowym.
2. Dla celów oszacowania złożoności należy założyć, że całkowita liczba miast może być dużo większa niż liczba mostów wytypowanych przez inżynierów.
3. Oznaczenia miast po przeciwnych stronach rzeki nie są ze sobą powiązane, co w szczególności oznacza, że może wystąpić krotka np. $(0, 0)$

Rozważmy następujące dane:

```
T = [ (1, 2), (2, 3), (3, 0) ]
```

Wywołanie `mosty(T)` powinno zwrócić wynik **2**. Można zbudować mosty pomiędzy miastami **1-2** oraz **2-3**. Zbudowanie mostu między miastami **3-0** wyklucza pozostałe dwa ze względu na przecięcie (patrz rysunek powyżej).

Podpowiedź. Czy można w jakiś sposób przeformułować ten problem?