

# 1 Straż pożarna

Mamy zdefiniowaną listę miast. W każdym z nich istnieje jednostka straży pożarnej. Utrzymanie każdej z nich jest bardzo kosztowne, dlatego chcemy zredukować ich liczbę. W danym mieście można zlikwidować jednostkę, o ile straż pożarna może do niego dojechać wystarczająco szybko z innego miasta. Twoim zadaniem jest napisanie programu, który pozwoli na jak najbardziej ekonomiczne rozmieszczenie jednostek straży pożarnej w zadanych miastach.

Program powinien pobrać dane wejściowe z pliku „in.json”. Dane wejściowe są w formacie obiektu JSON, przykładowo:

```
{
  "miasta" : ["A", "B", "C", "D", "E"],
  "drogi" : [
    {
      "miasta" : ["A", "B"],
      "czas_przejazdu" : 2
    },
    {
      "miasta" : ["A", "C"],
      "czas_przejazdu" : 3
    },
    {
      "miasta" : ["A", "D"],
      "czas_przejazdu" : 4
    },
    {
      "miasta" : ["A", "E"],
      "czas_przejazdu" : 1
    }
  ],
  "timeout" : 5,
  "max_czas_przejazdu" : 10
}
```

gdzie

- „miasta” – lista miast, początkowo w każdym mieście znajduje się jednostka straży pożarnej;  $0 < \text{liczba miast} < 1000000$
- „drogi” – list dróg; pole „miasta” zawiera informację, jakie miasta łączy droga, a pole „czas\_przejazdu” jaki jest czas przejazdu drogą;  $0 < \text{„czas\_przejazu”} < 100$
- „max\_czas\_przejazdu” – liczba oznaczająca jaki jest maksymalny dopuszczalny czas przejazdu straży pożarnej zanim dotrze do celu;  $0 < \text{„max\_czas\_przejazu”} < 10000$
- „timeout” – czas w sekundach po jakim program musi zakończyć swoje działanie (tolerancja 1%); w tym czasie program powinien zapisać najlepsze rozwiązanie jakie udało mu się znaleźć;  $0 < \text{„timeout”} < 10000$

Wynikiem działania programu powinna być lista miast, w których straż pożarna musi pozostać. Program powinien umieścić wynik w pliku „out.json” w postaci listy w formacie JSON, np.:

["A", "D"]

#### ZAŁOŻENIA

1. drogą można poruszać się w dowolnym kierunku, w obu kierunkach czas przejazdu jest taki sam
2. wynik jest poprawny, jeśli dla każdego miasta spełniony jest jeden z poniższych warunków:
  - istnieje lista dróg łącząca go z innym miastem, gdzie znajduje się jednostka straży pożarnej i suma czasów przejazdu tymi drogami jest mniejsza lub równa „max\_czas\_przejazdu”
  - w tym mieście znajduje się jednostka straży pożarnej
1. wynik jest optymalny, jeśli jest poprawny i liczba miast jest najmniejsza możliwa
1. miasta są bezpośrednio połączone co najwyżej jedną drogą

Rozwiązanie składa się z następujących elementów:

- kod źródłowy w języku Java lub Scala
- dokumentacja uruchomieniowa: plik README zawierający informacje jak skompilować kod i uruchomić program oraz jak uruchomić testy

Całość powinna być spakowana do pliku w formacie ZIP i wysłana jako załącznik do e-maila.

Kryteria oceny:

- poprawność wyniku (tak jak zostało to opisane w sekcji założenia, punkt 2.)
- jakość dokumentacji uruchomieniowej
- oprócz tego na ocenę zadania wpływają następujące elementy:
  - optymalność wyniku
  - przejrzystość i elegancja kodu
  - jakość testów jednostkowych

Dodatkowe punkty można zdobyć za implementację wykorzystującą współbieżność.