

CSCB039 Алгоритми и програмиране

Контролна работа № 3

Задача 1. Даден е свързан граф G , който съдържа V върха, номерирани от 0 до $V-1$ и E ребра. Броят на върховете може да бъде до 1000 върха и всяко ребро участва в точно един цикъл. Освен това всеки цикъл има най-много два върха, които участват и в друг цикъл. Напишете програма, която да намери най-дългия път в G , който не минава повторно през никой връх в графа.

Вход: На входа на програмата трябва да се задават стойност N , представляваща броят на върховете (номерирани от 1 до N), стойност M - броят на ребрата на графа ($5 \leq N \leq 1000$) и M реда, където на всеки ред се задават две стойности, представляващи двата края на ребро. Всяко от ребрата е двупосочно, като между два върха не може да има повече от едно ребро и няма ребра, двата края на които са в един и същ връх. Например,

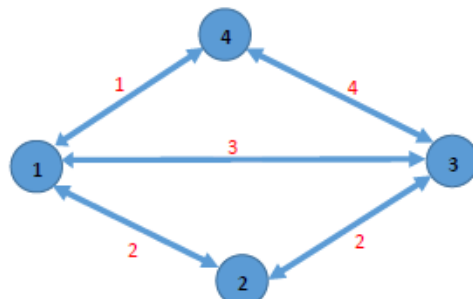
```
N = 6  
M = 7  
5 1  
2 4  
3 6  
6 2  
2 5  
4 1  
2 3
```

Изход. Програмата трябва да изведе една стойност, представляваща, намерената дължина на максимален път в графа, който не повтаря върхове. Например,

The cost of the longest path is: 5

Задача 2. ЕНЕРГО-ПРО Варна решава да обнови своя автопарк, като замени аварийните автомобили с Tesla Model 3. Сашо, който е част от аварийната група, отговаря за обекти, разположени в няколко съседни населени места. Той разполагал с картата на района, където са обозначени населените места, пътищата, които ги свързват, и дължините на тези пътища. Между

всяка двойка населени места съществува максимум един двупосочен път и няма пътища, които водят обратно към едно и също населено място.



Когато се получи сигнал за авария, Сашо трябва да състави списък с различни маршрути до населеното място с аварията, като дължината на всеки маршрут не трябва да надвишава максималния пробег на електромобила Tesla Model 3. Маршрутите трябва да водят само до мястото на аварията и да не преминават през едно и също населено място повече от веднъж. Напишете програма, която ще изготви списък с всички възможни маршрути, чиято дължина не надвишава максималния пробег на електромобила Tesla Model 3.

Вход: На входа на програмата се задават две цели числа, разделени с интервал – броят V на населените места в района на аварийната група, номерирани от 1 до V и броят R на пътищата между двойки населени места. След това се задават R реда, като всеки ред съдържа по три цели числа C_1 , C_2 и D , които са разделени с интервали. C_1 и C_2 са номерата на две населени места, свързани с път, а D – разстоянието между тях. След тези R реда се въвеждат три цели числа: номерът S на населеното място, от което тръгва аварийната група, номерът T на населеното място, в което е възникнала аварията и дължината M на пътя, който електромобилът може да измине без презареждане.

Ограничения:

$$\begin{aligned} 2 &\leq V \leq 64, \\ 2 &\leq R \leq 64 \\ 1 &\leq D \leq 9999 \end{aligned}$$

Изход: Програмата трябва да изведе намерените маршрути, подредени в нарастващ ред по дължините им, а при равни дължини – в лексикографски нарастващ ред. Редът започва с дължината на маршрута, следвана от 'i' и интервал. Следват номерата на населените места, които формират маршрута, разделени с интервали. Ако няма намерени маршрути, които удовлетворяват условията, програмата трябва да изведе *No routs*.

Вход:

```
4 5
1 2 2
1 3 3
1 4 1
2 3 2
3 4 4
1 3 4
4 5
1 2 2
1 3 3
1 4 1
2 3 2
3 4 4
1 4 10
```

Изход:

```
3: 1 3
4: 1 2 3
1: 1 4
7: 1 3 4
8: 1 2 3 4
```

Задача 3. В един град има n кръстовища, някои от които са свързани с преки двупосочни улици. Между две кръстовища може да има повече от една пряка улица. През пролетта има големи наводнения заради прииждащата река и някои от улиците остават под вода. За всяка улица е известно времето, за което общинската фирма ще успее да я почисти. Необходимо е част от наводнените улици да се изчистят от придошлата вода, за да може да се достига от всяко кръстовище до всяко друго. Задачата на общинската фирма е за минимално време на осигури възможност за достигане от всяко кръстовище до всяко друго.

Напишете програма, която по дадени n кръстовища и m улици, намира минималното време, за което общинската фирма ще се справи със задачата. Програмата трябва да обработва няколко тестови случая.

Вход: От първия ред на стандартния вход се въвежда броят на тестовите случаи. За всеки тестов случай следват няколко реда. Първият от тях съдържа три числа n , m и k , съответно броя на кръстовищата, общия брой на улиците и броя на улиците, които са наводнени. Всеки от следващите m реда съдържа три числа (x_i, y_i, t_i) – номерата на кръстовищата, между които има улица и времето, за което фирмата ще почисти улицата, ако е наводнена. От следващия ред се въвеждат номерата на улиците, които са наводнени. Улиците са номерирани от 1 до m в реда, по който се въвеждат.

Вход:

```
1
7 8 4
0 2 5
2 1 2
1 3 8
3 4 12
6 4 1
5 4 10
6 5 4
2 3 6
1 4 6 7
```

Изход: За всеки тестов случай на един ред на стандартния изход програмата трябва да изведе минималното време, за което фирмата ще се справи със задачата.

Изход:

```
21
```

Ограничения:

- $1 \leq n \leq 1000$
- $1 \leq m \leq 10000$
- $1 \leq k \leq m$
- $1 \leq t_i \leq 30000$