

Pentru acest laborator, cerințele problemelor includ citirea din linia de comandă a căilor fișierelor de intrare și de ieșire. Pentru aceasta, se poate folosi o implementare similară cu următoarea secvență de cod:

```
int main(int argc, char * argv[])
{
    //calea fisierului de intrare este data in argv[1]
    std::ifstream fin(argv[1]);
    //calea fisierului de iesire este data in argv[2]
    std::ofstream fout(argv[2]);
    //implementarea problemei
    //...

    fin.close();
    fout.close();
    return 0;
}
```

Problema 1

Cerința

Având dat un graf orientat ponderat și un vârf sursă, calculați costul minim de la vârful sursă până la fiecare vârf accesibil din graf. Implementarea trebuie să fie eficientă din punctul de vedere al memoriei folosite și al timpului de execuție. (Implementați un algoritm de drum minim în graf: Bellman- Ford, Dijkstra.)

Formatul sursei

Soluția va fi implementată în limbajul de programare C++. Numele sursei va fi „p1.cpp”. Căile fișierelor de intrare și de ieșire vor fi date ca parametri în linia de comandă, prima fiind calea fișierului de intrare și a doua fiind calea fișierului de ieșire.

Formatul fișierului de intrare

Fișierul de intrare conține pe prima linie 3 numere separate prin spațiu: **V E S**. **V** reprezintă numărul de vârfuri pentru graful dat. **E** reprezintă numărul arcelor din graf. **S** este nodul sursă de la care se vor calcula costurile minime.

Următoarele **E** linii vor conține câte 3 numere separate prin spațiu, reprezentând câte un arc: **x y w**. **x** este nodul sursă al arcului, **y** este nodul destinație, iar **w** este ponderea. Indexarea vârfurilor se face de la 0.

Valorile din fișierul de intrare se încadrează în următoarele limite:

- $1 \leq V \leq 10000$;
- $0 \leq E \leq 150000$;
- $0 \leq S < V$;
- $0 \leq x < V$;
- $0 \leq y < V$;
- $1 \leq w \leq 100$.

Formatul fișierului de ieșire

Fișierul de ieșire va conține o singură linie cu N valori separate prin spațiu. Valoarea de pe poziția i reprezintă costul drumului de cost minim de la vârful sursă S până la vârful i. Dacă vârful i este inaccesibil din vârful S, atunci pe poziția i se va scrie șirul de caractere „INF”.

Exemplu

Fișier de intrare
5 7 0
0 2 4
0 3 1
1 0 1
1 2 4
2 0 1
2 3 1
3 4 1

Fișier de ieșire
0 INF 4 1 2

Fișierul de intrare descrie un graf orientat ponderat cu 5 vârfuri și 7 arce. Vârful sursă este 0. 0 este accesibil din 0, iar costul asociat este 0, astfel prima valoare din fișierul de ieșire este 0. Vârful 1 nu este accesibil din 0, astfel a doua valoare din fișierul de ieșire este INF. 2 este accesibil din 0 prin intermediul drumului format dintr-un singur arc 0->2, drum ce are asociat costul 4, astfel a treia valoare din fișierul de ieșire este 4. 3 este accesibil din 0 prin intermediul drumului format dintr-un singur arc 0->3, drum ce are asociat costul 1, astfel a treia valoare din fișierul de ieșire este 1. 4 este accesibil din 0 prin intermediul drumului format din două arce 0->3->4, drum ce are asociat costul $1 + 1 = 2$, astfel a treia valoare din fișierul de ieșire este 2. Costurile scrise în fișier sunt cele minime. Vârful 4 este accesibil din 0 și prin intermediul drumului 0->2->3->4, însă acesta are costul asociat $4 + 1 + 1 = 6$, cost care nu este minim.

Problema 2

Cerința

Având dat un graf rar orientat ponderat calculați costul minim de la fiecare vârf până la fiecare vârf accesibil din acesta. Implementarea trebuie să fie eficientă din punctul de vedere al memoriei folosite și al timpului de execuție. (Implementați algoritmul lui Johnson.)

Formatul sursei

Soluția va fi implementată în limbajul de programare C++. Numele sursei va fi „p2.cpp”. Căile fișierelor de intrare și de ieșire vor fi date ca parametri în linia de comandă, prima fiind calea fișierului de intrare și a doua fiind calea fișierului de ieșire.

Formatul fișierului de intrare

Fișierul de intrare conține pe prima linie 2 numere separate prin spațiu: V E. V reprezintă numărul de vârfuri ale grafului. E reprezintă numărul de arce ale grafului.

Următoarele E linii conțin câte 3 numere separate prin spațiu, reprezentând câte un arc: $x \ y \ w$. x este nodul sursă al arcului, y este nodul destinație, iar w este ponderea. Indexarea vârfurilor se face de la 0.

Valorile din fișierul de intrare se încadrează în următoarele limite:

- $1 \leq V \leq 1000$;
- $0 \leq E \leq 10000$;
- $0 \leq x < V$;
- $0 \leq y < V$;
- $-100 \leq w \leq 100$;

Formatul fișierului de ieșire

Fișierul de ieșire va conține o linie sau $E + V$ linii. Dacă graful dat în fișierul de intrare conține un circuit de cost negativ, fișierul de ieșire va conține o singură linie cu valoarea -1. În cazul în care graful nu conține niciun circuit de cost negativ, fișierul de ieșire va conține $E + V$ linii, având semnificația descrisă mai jos.

Primele E linii reprezintă muchiile grafului după reponderare. Fiecare linie va fi formată din 3 valori separate prin spațiu, reprezentând câte un arc: $x \ y \ \hat{w}$. x este nodul sursă al arcului, y este nodul destinație, iar \hat{w} este noua pondere a arcului. Arcele vor fi ordonate întâi după nodul sursă, apoi după nodul destinație.

Următoarele V linii reprezintă matricea distanțelor, fiecare dintre aceste linii conținând V valori. Linia i a matricei (linia $V + i$ a fișierului de ieșire) reprezintă costurile drumurilor de cost minim de la vârful sursă i până la toate vârfurile grafului. Valoarea de pe poziția j a liniei i reprezintă costul minim de la vârful sursă i până la vârful j . Dacă vârful j este inaccesibil din vârful i , atunci pe poziția j a liniei i se va scrie șirul de caractere „INF”.

Exemplu 1

Fișier de intrare
3 4
0 2 6
1 0 -2
1 2 5
2 0 -3

Fișier de ieșire
0 2 3
1 0 1
1 2 5
2 0 0
0 INF 6
-2 0 4
-3 INF 0

Exemplu 2

Fișier de intrare
4 5
0 1 5
1 2 -3
2 0 -1
2 3 7
3 1 -5

Fișier de ieșire
-1