

Problema 1 – Traseu**100 puncte**

Primăria orașului City și-a propus înființarea celebrului serviciu *hop-on hop-off* pentru turiștii care vizitează orașul (un autobuz care are opriri în puncte fixate unde turiștii pot să urce în autobuz sau să coboare din acesta). În oraș sunt mai multe obiective turistice ce merită a fi incluse în traseu. Din motive lesne de înțeles, comisia din primărie care se ocupă de stabilirea traseului dorește ca acest traseu să includă minim trei obiective distincte.

Cunoscându-se obiective turistice din oraș și străzile care leagă aceste obiective și pe care poate circula autobuzul, să se determine traseele **distincte** dintre care comisia va putea selecta unul.

Date de intrare

Fișierul de intrare **traseu.in** conține pe prima linie, separate printr-un spațiu, două numere naturale n și m reprezentând, respectiv, numărul de obiective din oraș și numărul de străzi din oraș care leagă aceste obiective. Pe următoarele m linii se găsesc câte două numere naturale distincte, separate între ele prin câte un spațiu, reprezentând numerele de ordine ale unor obiective între care există o stradă pe care autobuzul poate circula.

Date de ieșire

Fișierul de ieșire **traseu.out** va conține pe prima linie o valoare t reprezentând numărul de trasee ce pot fi stabilite. Pe fiecare din următoarele t linii se va descrie câte un traseu prin scrierea mai multor valori, separate între ele prin câte un spațiu, reprezentând, în ordine, obiectivele incluse în respectivul traseu.

Restricții și precizări

- $3 \leq n \leq 1000$
- $0 \leq m \leq 10000$
- În dreptul fiecărui obiectiv turistic din oraș este amenajat **un punct de oprire** pentru autobuz.
- Între două obiective există **cel mult o stradă**.
- Pe fiecare stradă din oraș care leagă două obiective se poate circula în **ambele sensuri**.
- Un traseu este format din mai multe **obiective distincte**.
- Două obiective pot fi incluse succesiv într-un traseu dacă acestea sunt legate printr-o stradă.
- Primul și ultimul obiectiv dintr-un traseu sunt legate printr-o stradă (**autobuzul reia, ciclic, traseul**).
- Un traseu este distinct în raport cu celelalte dacă are în componența sa **cel puțin o stradă care nu apare în niciun alt traseu**.
- Traseele pot fi descrise în orice ordine.
- Dacă există mai multe variante de stabilire a traseelor poate fi afișată oricare dintre acestea.
- Pentru rezolvarea corectă doar a primei cerințe se acordă 50% din punctaj. Nu se acordă alte punctaje parțiale.
- Memorie disponibilă: 8 Mb.

Exemplu

traseu.in	traseu.out
7 7	2
2 3	2 1 3
4 5	4 5 3 2 1
5 3	
1 3	
2 1	
6 7	
1 4	

Timp maxim de execuție/test : 1 secundă

Problema 2 – Dual Core**100 puncte**

Algorel are un calculator nou cu procesor dual core, care poate rula două procese simultan. Dacă cele două procese au timpul de rulare t_1 și t_2 timpul de rulare paralelă este $t = \max(t_1, t_2)$. *Algorel* are o listă de n procese pentru care se cunosc timpii de rulare t_i , ($i=1\dots n$) și trebuie să scrie un program care să determine **timpul total minim de rulare al proceselor**, știind că două procese succesive pot fi rulate simultan.

Ajutați-l pe *Algorel* și scrieți un program care determină t_{min} timpul total minim de rulare al proceselor pe procesorul dual core și timpul total de rulare al proceselor pe un singur procesor.

Date de intrare

Fișierul **dualcore.in** conține pe prima linie **un număr natural n** , reprezentând numărul total de procese care vor fi rulate, iar pe a doua linie **n numere naturale** care reprezintă timpul de rulare al fiecărui proces, separate printr-un singur spațiu liber.

Date de ieșire

Fișierul de ieșire **dualcore.out** va conține:

- **o singură linie cu două valori separate printr-un spațiu:** două numere întregi, ce reprezintă timpul total minim de rulare al proceselor pe procesorul dual core și timpul de rulare al proceselor pe un singur procesor.

Restricții și precizări

- numărul de procese $1 \leq n \leq 100.000$
- timpul de rulare al unui proces t_i este $1 \leq t_i \leq 100.000$, $i=1, n$
- pentru 40% din teste $n \leq 45$
- memorie disponibilă: 8 Mo

Exemplu

dualcore.in	dualcore.out	Explicație:
5 1 4 5 3 2	9 15	Se grupează procesele 2,3 și 4,5 obținând un timp total minim: $1 + \max(4,5) + \max(3,2) = 9$. Timpul total de rulare cu un singur procesor este 15.

Timp maxim de execuție/test : 1 secundă