

Kamen

Imamo N kamenčića različitih oblika i sa potpuno različitim težinama. Ne postoje dakle dva kamenčića iste težine. N je neparan broj. Kamenčići su označeni brojevima od 1 do N . Kako bi našli srednje težak kamenčić, onaj koji je teži od $(N-1)/2$ i lakši od $(N-1)/2$ kamenčića izvodimo sljedeći test :

Imamo vagu pomoću koje možemo odredit koji od dva izabrana kamenčića je teži, odnosno lakši. Nakon nekoliko mjerenja možemo sa sigurnošću odrediti neke kamenčiće koji ne mogu biti srednje teški.

Na primjer, sljedeća tabela pokazuje rezultate M vaganja. U svakom od vaganja jedan kamenčić se pokazao težim od drugog izabranog kamenčića. Ukupan broj kamenčića je 5. ($M=4$ i $N=5$).

Kamenčić 2 je teži od kamenčića 1.

Kamenčić 4 je teži od kamenčića 3.

Kamenčić 5 je teži od kamenčića 1.

Kamenčić 4 je teži od kamenčića 2.

Na osnovu rezultata datih u tabeli nije uvijek moguće odrediti srednje težak kamenčić ali je moguće za neke od kamenčića sa sigurnošću reći da nisu srednje teški. Na primjer, kamenčić 1 i kamenčić 4 nisu srednje teški jer : 2, 4 i 5 su teži od 1 a 1, 2 i 3 su lakši od 4. Prema tome možemo zanemariti 1 i 4.

Vaš zadatak je da napišete program koji će brojati kamenčiće koji ne mogu biti srednje težine.

Ulazni podaci

Ulazna datoteka je kamen.in. Prva linija ulazne datoteke sadrži dva cijela broja, N ($1 \leq N \leq 99$) koji označava ukupan broj kamenčića i M koji označava ukupan broj izvršenih vaganja. U sljedećih M linija data su dva broja tako da prvi označava teži kamenčić a drugi označava lakši kamenčić prilikom vaganja.

Izlazni podaci

Izlazna datoteka je kamen.out. Sadrži samo jednu liniju sa jednim brojem, brojem kamenčića za koje možemo sa sigurnošću reći da nisu srednje težine.

Primjer 1

Input(kamen.in)
5 4
2 1
4 3
5 1
4 2
Output(kamen.out)
2

Primjer 2

Input(kamen.in)
5 3
2 1
1 3
3 4
Output(kamen.out)
2

Ograničenja na resurse

Vaš program se treba izvršavati za ne više od 1s i ne smije koristiti više od 64 MiB memorije po svakom testnom slučaju.