

Királyok harca

Bergengóciában sok városállam található, közöttük levő utakkal. Két városállam királya elhatározza, hogy meghódítja az összes várost. A királyok az első évben meghódítják a tőlük 1 távolságra levő városokat (azaz a közvetlen szomszédjukat). A második évben az ezektől 1 távolságra levőket, ... és így tovább. Ha olyan városhoz érnek, amit mindketten meghódítanának, akkor azé lesz a város, aki azt több szomszédjából tudja támadni. Ha mindketten ugyanannyi szomszédból támadnák, akkor az a város örökre szabad, független lesz. Pontosabban fogalmazva, ha a V várost az A király katonái h_1 olyan közvetlen szomszédos városból támadják, amelyet már elfoglaltak, a B király katonái pedig h_2 olyan közvetlen szomszédos városból támadják, amelyet már elfoglaltak, akkor az A királyé lesz a V város ha $h_1 > h_2$, a B királyé lesz ha $h_2 > h_1$, egyébként ($h_1 = h_2$ esetén) a V város örökre szabad marad, oda egyik király katonái sem léphetnek be később sem.

Írj programot, amely megadja, hogy a két király mely városokat tudja elfoglalni! A saját városuk nem számít elfoglalt városnak.

Bemenet

A standard bemenet első sorában a városok száma ($3 \leq N \leq 10\,000$), a két király városának sorszáma ($1 \leq A \neq B \leq N$) és a közvetlen utak száma ($1 \leq M \leq 100\,000$) van. A következő M sorban egy-egy út két végpontján levő város sorszáma van ($1 \leq X_i \neq Y_i \leq N$). A városokat az $1, \dots, N$ számokkal azonosítjuk.

Kimenet

A standard kimenet első sorába az A város királya által elfoglalt városok számát kell kiírni! A második sorba ezen városok sorszámait kerüljenek, növekvő sorrendben! A harmadik sorba a B város királya által elfoglalt városok számát kell kiírni! A negyedik sorba ezen városok sorszámait kerüljenek, növekvő sorrendben! Ha az első, illetve a 3. sorban 0 van, akkor a 2., illetve 4. sor üres sor legyen!

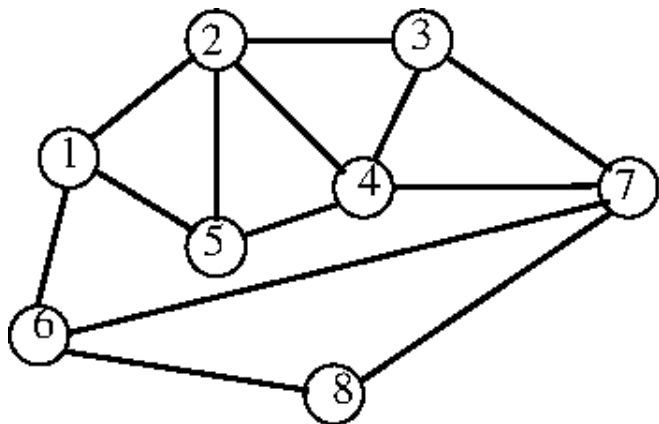
Példa

Bemenet

```
8 11 1 7
1 2
1 5
1 6
2 4
3 2
4 3
5 4
3 7
7 6
6 8
7 8
```

Kimenet

```
3
2 4 5
2
3 8
```



Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MiB

Pontozás

A pontok 20%-a szerzhető olyan tesztekre, ahol $N \leq 100$

A pontok további 25%-a szerzhető olyan tesztekre, ahol $N \leq 1000$

Összegzések

Készíts programot, amely kiszámítja, hogy adott N pozitív egész szám hányféleképpen írható fel olyan összegként, amelynek minden tagja különböző! Két összegzést nem tekintünk különbözőnek, ha azok csak a tagok sorrendjében különböznek.

Bemenet

A *standard bemenet* első sorában egyetlen egész szám van, az N ($1 \leq N \leq 500$) értéke.

Kimenet

A *standard kimenet* első és egyetlen sorába azt az M értéket kell írni, ahányféleképpen felírható az N olyan összegként, amelyben minden tag különböző!

Példa

Bemenet	Kimenet
10	10

Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MB

Pontozás

A pontok 25%-szerezhető olyan tesztekre, ahol $N \leq 10$.

A pontok további 25%-szerezhető olyan tesztekre, ahol $N \leq 100$.

Gyorséttermek

Egy új gyorsétteremlánc szeretne nyitni két éttermet Hosszúfalván. Hosszúfalva N lakosa mind egy út mentén él, az i . lakos házának távolsága az út kezdetétől x_i . Előfordulhat, hogy többen is laknak egy házban. A gyorsétteremláncnak Q különböző terve van a két létesítendő étterem elhelyezkedésére, az i . tervben a két étterem távolsága az út kezdetétől A_i és B_i . A két étterem megnyitásakor Hosszúfalva minden lakója rögtön elrohan a hozzá közelebbi étterembe (ha egyenlő távolságra vannak, akkor bármelyikbe mehet). Az étteremlánc vezetői szeretnék tudni minden tervre, hogy összesen mennyi távolságot tennének meg a lakók, ha az éttermeket abban a két pozícióban létesítenék.

Készíts programot, amely meghatározza a lakók által megtett össztávolságot minden terv esetén!

Bemenet

A *standard bemenet* első sora két egész számot tartalmaz, a lakók N számát ($1 \leq N \leq 100\,000$) számát és a tervek Q számát ($1 \leq Q \leq 100\,000$). A második sorban N pozitív egész szám van, a lakosok házainak pozíciói ($1 \leq x_i \leq 10^9$). A következő Q sor egy-egy tervet ír le, tehát meg van adva a két létesítendő étterem pozíciója, amelyek pozitív egész számok ($1 \leq A_i, B_i \leq 10^9$). Előfordulhat, hogy a két étterem pozíciója azonos.

Kimenet

A *standard kimenet* Q sort tartalmazzon, és minden sorban egyetlen egész szám legyen, az adott terv esetén a lakosok által megtett össztávolság.

Példa

Bemenet	Kimenet
5 2	81
12 50 70 1 10	33
10 100	
60 8	

Korlátok

Időlimit: 0.5 mp.

Memórialimit: 64 MB

Pontozás

A pontok 25%-a szerezhető olyan tesztekre, ahol $N, Q \leq 1000$.

A pontok további 15%-a szerezhető olyan tesztekre, ahol $A_i = B_i$ minden tervben.

Tükörszóra alakítás

Adott egy S karaktersorozat, amelyet tükörszóra szeretnénk alakítani. A tükörszó (más néven palindrom) olyan karaktersorozat, amely visszafelé olvasva megegyezik önmagával. A tükörszóra alakítás érdekében csupán egyetlen műveletet végezhetünk: kiválasztjuk S néhány (legalább egy) szomszédos karakterből álló részét, és ezen belül megváltoztatjuk a betűk sorrendjét. A kiválasztott részen kívül a betűk az eredeti sorrendben maradnak. Minimalizálni szeretnénk a kiválasztott részbe eső karakterek számát.

Készíts programot, amely meghatározza a lehető legrövidebb folytonos részét az S karaktersorozatnak, amelynek betűit átrendezve tükörszót kaphatunk!

Bemenet

A *standard bemenet* egyetlen sora az S karaktersorozatot tartalmazza. S minden eleme az angol ábécé kisbetűiből kerül ki (a . . z), és a hossza legfeljebb 200 000 ($1 \leq |S| \leq 2 \cdot 10^5$). Garانتált, hogy S átrendezhető tükörszóra a fent leírt művelettel.

Kimenet

A *standard kimenet* egyetlen sorába egy pozitív egész számot kell írni, az S legrövidebb, nemüres folytonos részének hosszát, amelyen belül a betűket átrendezve tükörszót kaphatunk.

Példa

Bemenet	Kimenet
abbaaca	4
Bemenet	Kimenet
abcba	1

Magyarázat

A fenti első példa esetén az $S[3..6]$ részt átrendezve kaphatunk tükörszót: ab**ba**aca \rightarrow ab**ac**aba

A második példában S már kezdetben is tükörszó, így nem kell módosítanunk, de ekkor is egy nemüres részét kell választanunk, tehát kiválaszthatunk egy tetszőleges betűt, amit helyben hagyunk.

Korlátok

Időlimit: 0.5 mp.

Memórialimit: 64 MB

Pontozás

A pontok 15%-a szerezhető olyan tesztekre, ahol S -ben két b betű van, az összes többi betű a .

A pontok további 25%-a szerezhető olyan tesztekre, ahol S hossza legfeljebb 50.

A pontok további 25%-a szerezhető olyan tesztekre, ahol S hossza legfeljebb 3000.