Natjecanje timova studenata informatičara hrvatskih sveučilišta

Zagreb, Osijek, Rijeka, Pula

22. listopada 2017.

Zadaci

HRVATSKI SAVEZ INFORMATIČARA

FER

Zadatak A: Asfalt

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

U jednom nakrivljenom gradu, staze i ulice se ne grade tako da se sijeku pod pravim kutom, već uvijek pod kutom koji je višekratnik od 60 stupnjeva. Točnije, grad možemo smjestiti u trokutastu mrežu odnosno koordinatni sustav u kojemu x i y osi zatvaraju kut od 60 stupnjeva kao na slici dolje. Ako su A i B točke sa cjelobrojnim koordinatama onda su one *susjedi* ako je njihova Euklidska udaljenost jednaka 1. Primijetite da sva točka ima točno 6 susjeda.

U gradu postoji n *staza* — svaka staza povezuje neku točku i jednog njenog susjeda. Grad je osigurao sredstva da se staze asfaltiraju i pretvore u moderne prometnice. Međutim, nije dozvoljeno da na tim novim prometnicama postoje oštri zavoji. Točnije, nije dozvoljeno da se dvije asfaltirane staze dodiruju u točki i pritom zatvaraju šiljasti kut.

Odredite koliko je najviše staza moguće asfaltirati tako da se ne dobije niti jedan oštar zavoj.

Ulazni podaci

U prvom redu se nalazi prirodni broj n ($1 \leq n \leq 1\,000$) — broj staza. U svakom od sljedećih n redova nalaze se četiri prirodna broja x_A, y_A, x_B i y_B ($1 \leq x_A, y_A, x_B, y_B \leq 100$) — redom koordinate točaka A i B koje su povezane stazom. Točke A i B su uvijek susjedne točke, a niti jedna staza se u ulazu ne pojavljuje više puta.

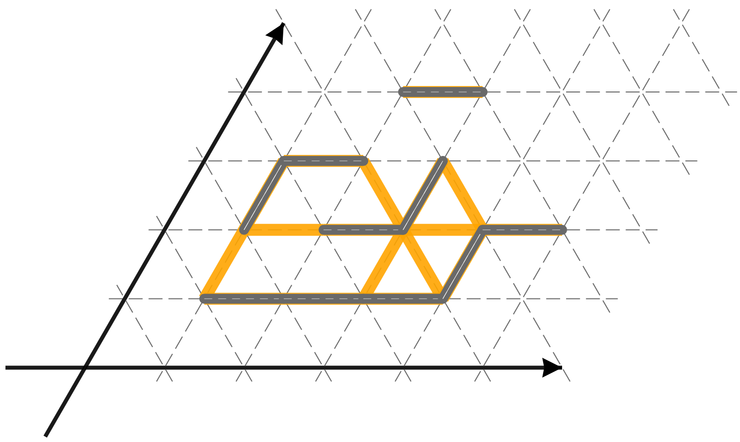
Izlazni podaci

Ispišite traženi maksimalni broj asfaltiranih staza.

Primjeri test podataka

ulaz

```
17
1 1 2 1
1 1 1 2
3 2 4 2
3 2 2 2
3 2 3 3
3 2 3 1
3 2 4 1
3 2 2 3
4 1 3 1
3 1 2 1
1 2 1 3
1 3 2 3
3 3 4 2
4 2 4 1
1 2 2 2
4 2 5 2
2 4 3 4
```



izlaz

```
10
```

Zadatak B: Cod

Vremensko ograničenje: 10 s
Memorijsko ograničenje: 512 MiB

U *deathmatch* varijanti popularne računalne igre, n igrača se međusobno bori u virtualnom svijetu. *Bodovi* igrača x se sastoji od dva cijela broja f_x i d_x — redom ukupan broj puta igrač x je ubio nekog drugog igrača i ukupan broj puta igrač x je ubijen od strane nekog drugog igrača. Dakle, kada tijekom igre igrač x ubije igrača y onda se f_x i d_y povećaju za jedan (a igrač y se, naravno, ponovno pojavljuje i nastavlja s igrom). Nije moguće da neki igrač ubije samog sebe, niti je moguće da se dva ubijanja dogode istovremeno. Igra završava čim neki igrač ostvari točno n ubijanja.

Rezultat završene igre je tablica koja se sastoji od n redaka i 2 stupca, gdje svaki redak sadrži bodove jednog igrača i to redom f_x i d_x. Igrači su u tablici poredani silazno po bodovima — najprije dolaze oni sa većom vrijednošću f_x, a među onima sa jednakom vrijednošću f_x najprije dolaze oni sa manjom vrijednošću d_x.

Djelomični rezultat je rezultat završene igre iz kojeg su izbrisane neke vrijednosti u tablici. Zadan je jedan djelomični rezultat, neka je m ukupan broj ispravnih rezultata od kojih je mogao nastati zadani djelomični rezultat. Odredite koliko je m modulo $10^9 + 7$.

Ulazni podaci

U prvom redu se nalazi prirodni broj n ($2 \leq n \leq 10$) — broj igrača. U k-om od sljedećih n redova se nalaze se dva cijela broja f_k i d_k ($-1 \leq f_k \leq n$, $-1 \leq d_k \leq n^2$) — k-ti redak djelomičnog rezultata. Izbrisane vrijednosti su označene brojem -1.

Zadani djelomični rezultat je ispravan u smislu da je dobiven od nekog rezultata završene igre na opisani način.

Izlazni podaci

Ispišite traženi broj mogućih originalnih rezultata modulo $10^9 + 7$

Primjeri test podataka

ulaz

```
2
-1 -1
-1 -1
```

izlaz

```
2
```

ulaz

```
3
-1 -1
-1 1
1 2
```

izlaz

```
2
```

ulaz

```
4
4 3
3 -1
2 -1
-1 2
```

izlaz

```
14
```

U prvom primjeru test podataka, mogući originalni rezultati su (2 0, 0 2) i (2 1, 1 2).

Zadatak C: Kaktus

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

Kaktus graf je povezani neusmjereni graf u kojem svaka dva jednostavna ciklusa imaju najviše jedan zajednički vrh. Podsjetimo se, jednostavan ciklus je ciklus u kojemu se vrhovi ne ponavljaju, a u kaktus grafu se niti jedna dva takva ciklusa ne smiju podudarati na više od jednom vrhu grafa.

Na zadanom kaktus grafu igramo igru tako da u svakom koraku odaberemo jedan brid te ga obrišemo. Brid biramo uniformno slučajno među svim preostalim bridovima (dakle svaki preostali brid brišemo sa jednakom vjerojatnošću), a svi slučajni izbori su nezavisni. Igra završava kada graf više nije povezan — postoje dva vrha između kojih ne postoji niti jedan put.

Odredite očekivani broj koraka da igra završi.

Ulazni podaci

U prvom redu se nalaze prirodni brojevi n i m ($1 \le n \le 600$, $1 \le m \le n(n-1)/2$) — broj vrhova i broj bridova grafa. Vrhovi grafa su označeni prirodnim brojevima od 1 do n. U svakom od sljedećih m redova nalaze se dva različita prirodna broja a i b ($1 \le a, b \le n$) koji označavaju brid između vrhova a i b. Svaka dva vrha će biti povezana najviše jednom bridom, a graf je kaktus graf kako je opisano u tekstu zadatka.

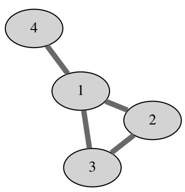
Izlazni podaci

Ispišite traženi očekivani broj koraka. Tolerirat će se apsolutno i relativno odstupanje od službenog rješenja za 10^{-6}.

Primjeri test podataka

ulaz

```
4 4
1 2
2 3
3 1
1 4
```

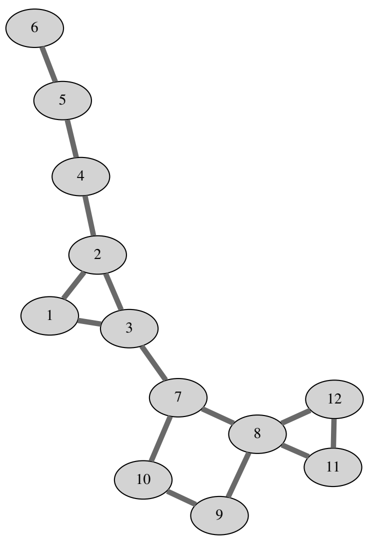


izlaz

```
1.75
```

ulaz

```
12 14
1 2
2 3
3 1
2 4
4 5
6 5
3 7
7 8
8 9
9 10
10 7
8 11
8 12
11 12
```



izlaz

```
2.17582418
```

Zadatak D: Kalendar

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

Kalendar jedne godine možemo zapisati u velikoj matrici znakova. Svaki element matrice je ili veliko slovo engleske abecede ili znamenka ili točka. Kalendar gradimo na sljedeći način:

- Svaki mjesec zauzima podmatricu od točno 8 redaka i 17 stupaca.
- Ime mjeseca (na engleskom jeziku, velikim slovima) je zapisano u prvom retku podmatrice počevši od drugog stupca podmatrice.
- Svi dani u mjesecu su zapisani u 6 grupa po dva stupca visine 7 redaka, između susjednih grupa se nalazi jedan prazan stupac (odnosno popunjen točkama).
- Svaka grupa sadrži uzastopne brojeve dana u jednom tjednu.
- Broj se sastoji od jedne ili dvije znamenke, ako je broj jednoznamenkast onda se nalazi se u desnom stupcu.
  - Prvi redak odgovara ponedjeljku.
  - Prva grupa stupaca mora sadržavati barem jedan broj dok peta i šesta grupa stupaca mogu biti prazne (na primjer ako mjesec sadrži 28 dana i počinje ponedjeljkom).
- Mjeseci godine su podijeljeni u tri reda odvojena jedim praznim retkom.
- U svakom retku se nalaze četiri uzastopna mjeseca odvojena jednim praznim stupcem.
- Na sva četiri ruba kalendara se nalazi prazna margina od jednog retka odnosno stupca.

Dakle cijeli kalendar se sastoji od točno 28 redaka i 73 stupca. Potpuni kalendar za 2017. godinu je dan dolje. Engleska imena mjeseci možete iščitati iz primjera. Prisjetimo se, godina je prijestupna ako je djeljiva sa 400, ili ako je djeljiva sa 4 i nije djeljiva sa 100. Prvi siječanj 1900-te godine je bio ponedjeljak.

```
.........................................................................
..JANUARY...........FEBRUARY..........MARCH.............APRIL............
.....2..9.16.23.30.....6.13.20.27........6.13.20.27........3.10.17.24....
.....3.10.17.24.31.....7.14.21.28........7.14.21.28........4.11.18.25....
.....4.11.18.25.....1..8.15.22........1..8.15.22.29........5.12.19.26....
.....5.12.19.26.....2..9.16.23........2..9.16.23.30........6.13.20.27....
.....6.13.20.27.....3.10.17.24........3.10.17.24.31........7.14.21.28....
.....7.14.21.28.....4.11.18.25........4.11.18.25........1..8.15.22.29....
..1..8.15.22.29.....5.12.19.26........5.12.19.26........2..9.16.23.30....
.........................................................................
..MAY...............JUNE..............JULY..............AUGUST...........
..1..8.15.22.29........5.12.19.26........3.10.17.24.31.....7.14.21.28....
..2..9.16.23.30........6.13.20.27........4.11.18.25.....1..8.15.22.29....
..3.10.17.24.31........7.14.21.28........5.12.19.26.....2..9.16.23.30....
..4.11.18.25........1..8.15.22.29........6.13.20.27.....3.10.17.24.31....
..5.12.19.26........2..9.16.23.30........7.14.21.28.....4.11.18.25.......
..6.13.20.27........3.10.17.24........1..8.15.22.29.....5.12.19.26.......
..7.14.21.28........4.11.18.25........2..9.16.23.30.....6.13.20.27.......
.........................................................................
..SEPTEMBER.........OCTOBER...........NOVEMBER..........DECEMBER.........
.....4.11.18.25........2..9.16.23.30.....6.13.20.27........4.11.18.25....
.....5.12.19.26........3.10.17.24.31.....7.14.21.28........5.12.19.26....
.....6.13.20.27........4.11.18.25.....1..8.15.22.29........6.13.20.27....
.....7.14.21.28........5.12.19.26.....2..9.16.23.30........7.14.21.28....
..1..8.15.22.29........6.13.20.27.....3.10.17.24........1..8.15.22.29....
..2..9.16.23.30........7.14.21.28.....4.11.18.25........2..9.16.23.30....
..3.10.17.24........1..8.15.22.29.....5.12.19.26........3.10.17.24.31....
.........................................................................
```

Arheolozi su pronašli djelić pravokutnog oblika koji je izrezan iz jednog takvog kalendara, također znaju da taj fragment nije rotiran niti na bilo koji drugi način izmijenjen. Odredite sve moguće godine između 1900. i 2100. uključivo iz kojeg je mogao biti izrezan taj fragment.

Ulazni podaci

U prvom redu nalaze se dva prirodna broja n i m ($2 \leq n, m \leq 10$) — broj redaka i stupaca u pronađenom fragmentu. U svakom od sljedećih n redova nalazi se niz od m znakova — jedan redak fragmenta.

Izlazni podaci

Potrebno je ispisati, uzlaznim redoslijedom, sve tražene godine, svaku u svoj redak. Test podaci će biti takvi da će uvijek postojati barem jedna moguća godina.

Primjeri test podataka

ulaz

```
2 8
DECEMBER
...2..9.
```

izlaz

```
1901
1907
1912
1918
1929
1935
1940
1946
1957
1963
1968
1974
1985
1991
1996
2002
2013
2019
2024
2030
2041
2047
2052
2058
2069
2075
2080
2086
2097
```

ulaz

```
3 2
..
29
..
```

izlaz

```
1904
1932
1960
1988
2016
2044
2072
```

Zadatak E: Letovi

Vremensko ograničenje: 5 s
Memorijsko ograničenje: 512 MiB

Mirko je na nagradnoj igri osvojio zrakoplovnu kartu za put oko svijeta. Karta se sastoji od n *kupona za let* $f_1, f_2, \ldots, f_n$ — k-ti kupon se može iskoristiti za jedan direktan let od *grada polaska* a_k do *grada dolaska* b_k. Za razliku od uobičajnih karata za put oko svijeta, grad dolaska na jednom kuponu ne mora nužno odgovarati gradu polaska na sljedećem kuponu. Svaki kupon za let Mirko može iskoristiti najviše jednom (dakle dozvoljeno je da ga uopće ne iskoristi). Međutim, kuponi se samo smiju koristiti originalnim redoslijedom — ako Mirko iskoristi kupone f_i i f_j gdje je $i < j$ onda kupon f_i mora iskoristiti prije nego što iskoristi kupon f_j.

Itinerar je niz gradova redom posjećenih na putu oko svijeta, a u kojem se isti grad može pojavljivati i više od jednom. Prvi i posljednji grad u itineraru mora biti Zagreb gdje Mirko živi, a itinerar mora sadržavati barem još jedan drugi grad. Neka je m broj različitih takvih itinerara koje Mirko može ostvariti sa svojom kartom. Odredite koliko je m modulo $10^9 + 7$.

Ulazni podaci

U prvom redu se nalazi prirodni broj n ($1 \leq n \leq 300\,000$) — broj kupona za let. U k-om od sljedećih n redova se nalaze dva različita niza znakova a_k i b_k — kodovi grada polaska i grada dolaska na k-tom kuponu. Svaki kod grada je niz od točno tri velika slova engleske abecede. Kod grada Zagreba je ZAG.

Izlazni podaci

Ispišite traženi broj itinerara modulo $10^9 + 7$

Primjeri test podataka

ulaz

```
4
ZAG SPU
SPU ZAG
ZAG SPU
SPU ZAG
```

izlaz

```
2
```

ulaz

```
8
PUY ZAG
ZAG OSI
OSI ZAG
ZAG DBV
OSI PUY
OSI DBV
DBV ZAG
PUY OSI
```

izlaz

```
4
```

U prvom primjeru test podataka, mogući itinerari su `ZAG-SPU-ZAG` i `ZAG-SPU-ZAG-SPU-ZAG`.

Zadatak F: NKD

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

U mnogim računalnim programima pojavljuje se lista *nedavno korištenih dokumenata* (NKD). Kao što joj ime kaže, lista sadrži nedavno korištene dokumente kako bi korisnik mogao brže doći do njih, izbjegavajući mukotrpno traženje među dokumentima. Lista je ograničenog kapaciteta — poznat je najveći broj dokumenata koji se mogu u istom trenutku nalaziti u listi.

Svaki put kad korisnik otvori neki dokument (bilo izborom iz NKD liste ili na neki drugi način), tada:

1. Ako se dokument već nalazi negdje u listi, pomakne se na početak liste.
2. U suprotnom, umetne se na početak liste. Ukoliko je prekoračen kapacitet liste, zadnji dokument u listi se izbacuje.

Zadan je kapacitet liste (koja je na početku prazna) i niz dokumenata koje korisnik otvara. Odredite sadržaj liste nakon otvaranja svih dokumenata u zadanom redoslijedu.

Ulazni podaci

U prvom redu ulaza nalazi se prirodni broj k ($1 \le k \le 10$) — kapacitet liste. U drugom redu nalazi se prirodni broj n ($1 \le n \le 500$) — broj dokumenata koje korisnik otvara. Svaki od sljedećih n redova sadrži ime jednog dokumenta kojeg korisnik otvara. Imena svih dokumenata bit će nizovi od najviše 10 malih slova engleske abecede, bez razmaka. Dokumenti su dani u redoslijedu u kojem ih korisnik otvara.

Izlazni podaci

Potrebno je ispisati sadržaj NKD liste nakon otvaranja svih dokumenata, svaki dokument u svoj red. Prvi ispisani red odgovara dokumentu na početku NKD liste.

Primjeri test podataka

ulaz

```
4
3
a
b
c
```

izlaz

```
c
b
a
```

ulaz

```
2
6
buba
koko
buba
ivan
ivan
koko
```

izlaz

```
koko
ivan
```

Zadatak G: Ploča

Vremensko ograničenje: 5 s
Memorijsko ograničenje: 512 MiB

Mirko je vlasnik neravnog zemljišta dimenzija $R \times S$ metara koje je popločano sa $R \cdot S$ ploča dimenzija 1×1 metar. Budući da je zemljište neravno, svaka ploča ima svoju visinu u centimetrima. Kažemo da su dvije ploče susjedne ako dijele zajedničku stranicu (svaka ploča ima najviše četiri susjedne). Kažemo da su dvije ploče *povezane* ako s prve možemo doći na drugu pri čemu u svakom koraku prelazimo s ploče na kojoj se nalazimo na njoj susjednu ploču iste visine.

Uslijed nezapamćene oluje na Mirkovo zemljište počeli su padati veliki komadi leda iz najvišeg dijela atmosfere, što nije dobra vijest budući da je Mirko bio kupovao najjeftinije kineske ploče i njihovu ugradnju bio prepustio izvjesnom harmonikašu Hamdiji. Stoga, kada komad leda padne na neku ploču, zbog siline udarca on će za 1 centimetar sniziti tu ploču i sve ploče koje su s njome povezane. Mirko je ipak optimist pa se nada da će mu ti udarci poravnati zemljište.

Vaš je zadatak preuzeti Mirkov optimizam i izračunati najmanji mogući broj komada leda koji mogu pasti tako da na kraju sve Mirkove ploče budu iste visine. Pretpostavljamo da će svaki komad leda pasti u različitom trenutku i samo na jednu ploču.

Ulazni podaci

U prvom redu nalaze prirodni brojevi R i S ($2 \leq R, S \leq 1\,000$) — dimenzije zemljišta. Slijedi R redova od po S prirodnih brojeva iz intervala $[1, 10^9]$ kojima su dane visine odgovarajućih ploča u centimetrima.

Izlazni podaci

Ispišite traženi minimalan broj udaraca leda nakon kojeg su sve ploče iste visine.

Primjeri test podataka

ulaz

```
3 3
3 3 2
2 3 3
2 3 1
```

izlaz

```
2
```

Nakon što komad leda padne na središnju ploču, sve ploče osim posljednje bit će visine 2. Nakon još jednog pada na središnju ploču, sve su ploče visine 1.

Zadatak H: Presjek

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

Adrian vodenim bojicama crta konveksne poligone u koordinatnom sustavu. U nedostatku vremena neće im popunjavati unutrašnjost, ali želi naglasiti redoslijed kojim su poredani tako da nijansa pojedine točke ruba poligona ovisi o broju poligona koji tu točku prekrivaju. Točnije, recimo da Adrian crta poligone $p_1, p_2, \ldots, p_n$, on će pojedini segment poligona p_j nacrtati nijansom t ako se taj segment (osim možda njegovih krajnjih točaka) nalazi u unutrašnjosti točno t od kasnijih poligona $p_{j+1}, \ldots, p_n$.

Kada Adrian crta neki segment nijansom t on potroši $\frac{1}{t+1}$ jedinica crne boje po jedinici duljine segmenta. Odredite ukupnu količinu boje koju Adrian treba potrošiti kako bi nacrtao sve poligone.

Ulazni podaci

U prvom redu se nalazi prirodni broj n ($1 \leq n \leq 10$) — broj poligona. Slijedi n blokova gdje k-ti blok sadrži opis poligona p_k. U prvom redu bloka se nalazi prirodni broj m ($3 \leq m \leq 20$) — broj vrhova poligona. U svakom od sljedećih m redova se nalaze dva cijela broja x i y ($0 \leq x, y \leq 100$) — koordinate jednog vrha poligona. Vrhovi poligona su zadani u pozitivnom smjeru (suprotno od kazaljke na satu), a poligon je uvijek konveksan te ne sadrži uzastopne paralelne stranice.

Možete pretpostaviti da se različiti poligoni nikad ne dodiruju duž vrha ili stranice. Točnije ako su A i B dužine odnosno stranice različitih poligona p_i i p_j onda A i B uopće nemaju zajedničkih točaka ili se sijeku u točno jednoj točki koja leži u unutrašnjosti dužina A i B.

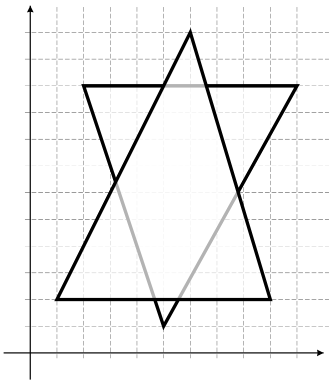
Izlazni podaci

Ispišite traženu ukupnu količinu boje. Tolerirat će se apsolutno i relativno odstupanje od službenog rješenja za 10^{-6}.

Primjeri test podataka

ulaz

```
2
3
2 10
5 1
10 10
3
1 2
9 2
6 12
```

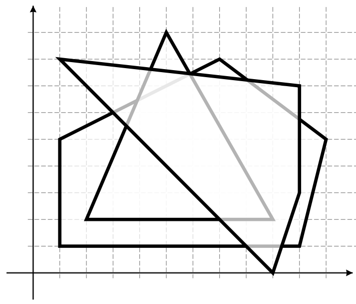


izlaz

```
51.98133148218655
```

ulaz

```
3
5
1 1
10 1
11 5
7 8
1 5
3
2 2
9 2
5 9
4
1 8
9 0
10 3
10 7
```



izlaz

```
69.85925507402564
```

Zadatak I: Sretan

Vremensko ograničenje: 3 s
Memorijsko ograničenje: 512 MiB

Za niz cijelih brojeva $a_1, a_2, \ldots, a_n$ kažemo da je *sretan* ako je svaki element niza (osim prvog i zadnjeg) jednak zbroju njemu susjednih elemenata: $a_j = a_{j-1} + a_{j+1}$.

Zadan je niz cijelih brojeva, u svakom koraku možemo odabrati jedan element niza te ga povećati ili smanjiti za jedan. Odredite minimalan broj koraka potreban da dobijemo sretan niz.

Ulazni podaci

U prvom redu se nalazi prirodni broj n ($3 \leq n \leq 300\,000$) — broj elemenata niza. Sljedeći red sadrži n cijelih brojeva $a_1, a_2, \ldots, a_n$ ($-10^9 \leq a_j \leq 10^9$) — zadani niz.

Izlazni podaci

Ispišite traženi minimalni broj koraka.

Primjeri test podataka

ulaz

```
4
5 1 -4 -5
```

izlaz

```
0
```

ulaz

```
4
3 2 0 3
```

izlaz

```
6
```

U drugom primjeru test podataka, sa 6 koraka možemo doći do sretnog niza (2 2 0 -2).

Zadatak J: Stablo

Vremensko ograničenje: 1 s
Memorijsko ograničenje: 512 MiB

Stablo je struktura koja se sastoji od n *vrhova* označenih brojevima od 1 do n i $n-1$ *bridova* postavljenih tako da između svaka dva vrha stabla postoji jedinstven put. Dodatno, u svaki vrh je upisan točno jedan znak i to veliko slovo `A` ili veliko slovo `B`.

Stablo je *uravnoteženo* ako ne postoji brid koji povezuje vrhove označene istim slovom. Stablo možemo pokušati uravnotežiti nizom koraka gdje u svakom koraku odaberemo jedan brid i zamijenimo znakove zapisane u vrhovima koje brid povezuje.

Odredite minimalan broj koraka potrebnih da se uravnoteži zadano stablo.

Ulazni podaci

U prvom redu nalazi se prirodni broj n ($1 \leq n \leq 300\,000$) — broj vrhova stabla. Sljedeći red sadrži niz od n znakova gdje je svaki znak veliko slovo `A` ili `B` — j-ti znak u nizu je znak inicijalno zapisan u vrhu j.

Svaki od sljedećih $n-1$ redova sadrži dva različita prirodna broja x i y ($1 \leq x, y \leq n$) — oznake vrhova direktno povezanih bridom. Vrhovi i bridovi čine stablo kao što je opisano u tekstu zadatka.

Izlazni podaci

Ispišite traženi minimalni broj koraka. Ako nije moguće uravnotežiti stablo, ispišite `-1`.

Primjeri test podataka

ulaz

```
6
AABBAB
1 2
1 3
3 4
3 5
1 6
```

izlaz

```
2
```

ulaz

```
4
AABB
1 2
1 3
1 4
```

izlaz

```
-1
```

ulaz

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

izlaz

```
5
```