
zadaci

zadatak	super95	kokos	trokut	banana
izvorni kôd	super95.pas super95.c super95.cpp	kokos.pas kokos.c kokos.cpp	trokut.pas trokut.c trokut.cpp	banana.pas banana.c banana.cpp
ulazni podaci	standardni ulaz			
izlazni podaci	standardni izlaz			
vremensko ograničenje (Intel Celeron 2.66Ghz)	1 sekunda	1 sekunda	5 sekundi	2 sekunde
memorijsko ograničenje (heap)	32 MB			
memorijsko ograničenje (stack)	8 MB			
broj bodova	60	70	80	90
	300			

super95

Vodeća naftna kompanija u jednoj dalekoj zemlji odlučila se za novi i revolucionarni način određivanja cijene njihovog najkorištenijeg benzina SUPER 95 s ekstra dodatkom najfinijeg sumpora.

Naime, na sastanku premijera s vodećim ljudima te kompanije donesena je odluka da će se od sada nadalje cijena tog benzina određivati tako da će **premijer odabrati jedno slovo** i da će se svaki dan određivati nova, viša cijena benzina koja će ovisiti o tom slovu i o broju dana proteklom od spomenutog sastanka.

Promotrimo niz svih prirodnih brojeva koji izgovoreni **počinju** zadanim slovom.

Npr. za slovo **P** niz je:

5, 15, 50, 51, 52, ..., 59, 500, 501, ...

Napišite program koji će za neko zadano slovo odrediti **N-ti** broj u nizu konstruiranom iz tog slova (to će biti cijena benzina N-tog dana).

ulazni podaci

U prvom i jedinom retku ulaza se nalazi zadano slovo (jedno od slova '**D**', '**J**', '**M**', '**O**', '**P**', '**S**' ili '**T**') i prirodni broj **N**.

Napomena: ulazni podaci će biti takvi da će rješenje biti **manje od 10^{12}** (koristite 64-bitne cjelobrojne tipove podataka – `int64` u Pascalu, `long long` u C/C++).

izlazni podaci

U prvi i jedini redak treba ispisati traženi broj.

test primjeri

ulaz	ulaz	ulaz
S 1	P 13	M 1000006
izlaz	izlaz	izlaz
7	500	1000000005

kokos

Zadano je N riječi, svaka od njih duljine $2K$ znakova.

Usmjereni graf u kojem je svaki vrh označen nekim slovom nazivamo "**kokos**" ako **za svaku** zadanu riječ **postoji usmjereni put** u tom grafu takav da se čitanjem oznaka vrhova kroz koje taj put prolazi **dobije zadana riječ** i da za vrhove na tom putu vrijedi:

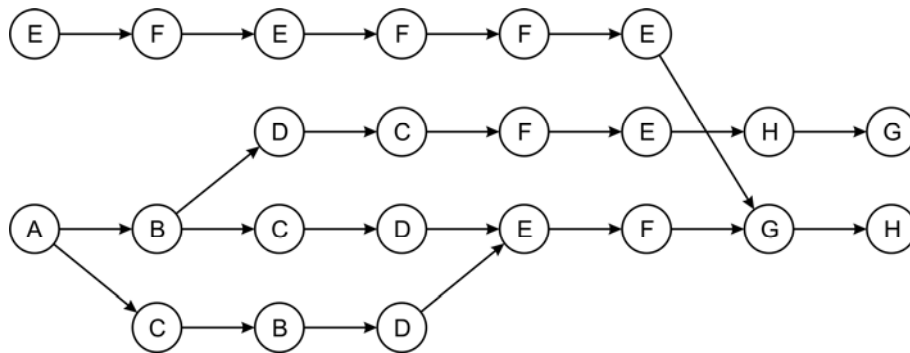
- ulazni stupanj prvog vrha je 0
- ulazni stupanj sljedećih $K-1$ vrhova je 1
- izlazni stupanj sljedećih $K-1$ vrhova je 1
- izlazni stupanj zadnjeg vrha je 0

Drugim riječima, putevi se smiju granati samo na prvih K slova, a smiju se sastajati tek nakon prvih K slova.

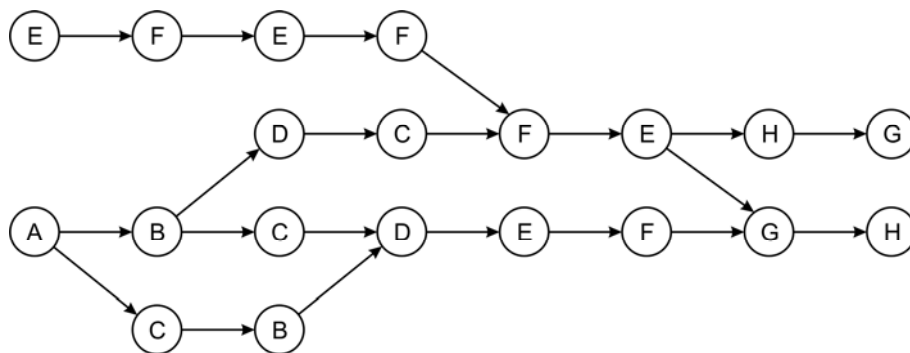
Za zadani skup riječi, kokos nazivamo **minimalnim** ako je njegov broj vrhova **što je manji mogući**.

Napišite program koji će za zadani skup riječi odrediti **broj vrhova** u minimalnom kokosu.

Primjer minimalnog kokosa (zadane riječi su kao u trećem test primjeru):



Na prvi pogled može nam se učiniti da smo graf mogli dodatno sažeti na sljedeći način:



ali takav graf **nije** kokos jer se putevi sastaju na 4. slovu (D) i jer se granaju na 6. slovu (E).

kokos

ulazni podaci

U prvom retku se nalaze dva prirodna broja N i K , $1 \leq N \leq 10\,000$, $1 \leq K \leq 100$.

U svakom od sljedećih N redaka nalazi se po jedna zadana riječ. Jedini dozvoljeni znakovi unutar riječi su velika slova engleske abecede.

izlazni podaci

U prvi i jedini redak treba ispisati traženi broj vrhova iz teksta zadatka.

test primjeri

ulaz

2 4
ABCDEFGH
EFGHIJKL

izlaz

16

ulaz

4 3
XXZZXX
XXYYZZ
AABBCZ
ABCZZZ

izlaz

18

ulaz

4 4
ABCDEFGH
ACBDEFGH
ABDCFEGH
EFEFFEGH

izlaz

23

trokut

Zadano je N točaka u ravnini.

Napišite program koji će od zadanih točaka **odabrati tri** takve da trokut kojem su one vrhovi sadrži **što je više moguće** zadanih točaka.

Za točke **u vrhovima** ili **na stranicama** nekog trokuta smatramo da se nalaze unutar njega.

ulazni podaci

U prvom retku se nalazi prirodni broj N , $3 \leq N \leq 300$.

U svakom od sljedećih N redaka nalaze se koordinate točke – prirodni brojevi manji ili jednaki od 100 000. Sve zadane točke će biti međusobno različite.

Napomena: ulazni podaci će biti takvi da će postojati barem tri nekolinearne točke.

izlazni podaci

U prvi redak treba ispisati broj točaka unutar traženog trokuta.

U drugi redak treba ispisati tri broja – **redne brojeve** vrhova traženog trokuta **u bilo kojem redoslijedu**.

test primjeri

ulaz	ulaz	ulaz
6	9	13
1 3	1 1	1 3
2 3	2 2	2 4
2 1	3 3	3 1
3 1	2 1	4 1
3 2	3 2	4 2
4 4	3 1	4 3
izlaz	4 2	4 4
	4 1	4 5
	5 1	5 1
5		5 2
1 4 6	izlaz	6 1
		6 5
	9	7 3
	1 9 3	izlaz
		9
		3 11 8

banana

Prometna mreža u zemlji banana sastoji se od N gradova (označenih brojevima od 1 do N) i $N-1$ cesta koje ih povezuju, a **između svaka dva** grada postoji **jedinstveni** put.

Usljed dugogodišnjeg neodržavanja ceste su prilično oštećene, a za svaku cestu su poznata dva broja A i B – broj A označava **trenutno** vrijeme (u sekundama) potrebno za prolazak te ceste, dok broj B označava **najkraće moguće** vrijeme potrebno za prelazak ceste ako se **sva oštećenja** na cesti poprave.

Količina vremena koja se uštedi popravkom neke ceste je proporcionalna količini novaca koja se potroši za taj popravak – za **jednu kunu** uštedi se **jedna sekunda**, a iznos uložen u popravak neke ceste mora biti **cjelobrojan**.

Npr. ako je trenutno za prolazak nekom cestom potrebno 1000 sekundi i ako potrošimo 10 kuna na popravak te ceste, tada nam nakon toga za prolazak treba samo 990 sekundi. Naravno, vrijeme prolaska te ceste **ne može biti manje** od gore spomenutog najkraćeg vremena B .

Zadana je ukupna količina novca na raspolaganju. Želimo novac uložiti tako da **vrijeme** potrebno da se od grada **broj 1** dođe do njemu (nakon izvršenih popravaka) **najudaljenijeg** grada bude **što je manje moguće**.

Napišite program koji će odrediti traženo vrijeme.

ulazni podaci

U prvom retku se nalaze dva cijela broja N i K , $2 \leq N \leq 100\,000$, $0 \leq K \leq 1\,000\,000$, broj gradova i ukupna količina novca na raspolaganju.

U svakom od sljedećih $N-1$ redaka nalaze se cijeli brojevi X , Y , A i B , $0 \leq B \leq A \leq 10\,000$. To znači da između gradova X i Y postoji cesta kojoj su pridruženi brojevi A i B , opisani u tekstu zadatka.

izlazni podaci

U prvi i jedini redak treba ispisati traženo vrijeme iz teksta zadatka.

test primjeri

ulaz

3 200
1 2 200 100
2 3 450 250

izlaz

450

ulaz

5 11
1 2 10 5
1 3 3 2
1 4 9 6
3 5 7 3

izlaz

6

ulaz

11 12
1 2 7 5
1 3 20 15
2 4 10 8
2 5 5 3
2 6 6 2
4 7 3 0
4 8 7 2
5 9 8 4
5 10 9 8
5 11 6 5

izlaz

17