

zadaci

zadatak	osmosmjerka	waclaw	lampice	šibice
izvorni kôd	osmo.pas osmo.c osmo.cpp	waclaw.pas waclaw.c waclaw.cpp	lampice.pas lampice.c lampice.cpp	sibice.pas sibice.c sibice.cpp
ulazni podaci	standardni ulaz			
izlazni podaci	standardni izlaz			
vremensko ograničenje (Athlon MP 2x2.1 GHz)	1 sekunda			
memorijsko ograničenje (heap)	32 MB			
memorijsko ograničenje (stack)	8 MB			
broj bodova	30	45	55	70
	200			

osmosmjerka

Osmosmjerka je popularna enigmatska razbibriga koja se sastoji od slova poredanih u N redaka i N stupaca te liste riječi koje se traže.

Osmosmjerka se rješava tako da se za svaku riječ iz liste pronađu **sva** pojavljivanja u osmosmjerci i prekriže **sva** odgovarajuća slova koja čine tu riječ. Riječ može početi bilo gdje i pružati se u bilo kojem od **osam smjerova** (gore, dolje, lijevo, desno te 4 dijagonalna smjera).

Rješenje osmosmjerke je riječ koju dobijemo kada spojimo **sva neprekrižena slova**, redom od prvog do zadnjeg retka i u svakom retku slijeva nadesno.

Napišite program koji će pronaći rješenje zadane osmosmjerke.

ulazni podaci

U prvom retku nalazi se prirodni broj N, $1 \leq N \leq 10$, veličina osmosmjerke.

U svakom od sljedećih N redaka nalazi se niz od N znakova koji predstavlja jedan redak osmosmjerke.

U sljedećem retku nalazi se prirodni broj R, $1 \leq R \leq 100$, broj zadanih riječi.

U svakom od sljedećih R redaka nalazi se jedna riječ koja se traži, duljine najviše 10 znakova.

U osmosmjerci i u riječima će se pojavljivati samo mala slova engleske abecede ('a'-'z').

izlazni podaci

U prvi i jedini redak potrebno je ispisati rješenje osmosmjerke.

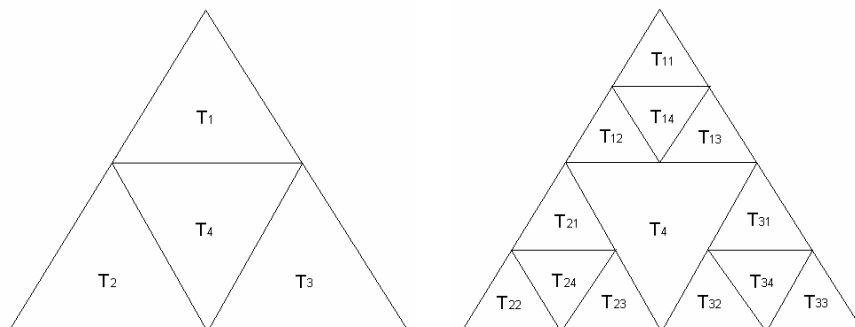
Napomena: ulazni podaci će biti takvi da će rješenje uvijek biti niz od najmanje jednog znaka.

test primjeri

ulaz	ulaz	ulaz
2	5	4
ab	patka	rast
cd	guska	plso
1	macka	tajv
ad	klopa	pnez
izlaz	krava	6
	4	rast
	patka	alan
bc	guska	nt
	macka	past
	krava	taj
	izlaz	bonus
	klopa	izlaz
		povez

Waclaw Sierpinski je poljski matematičar koji se volio igrati s trokutima. Jednoga dana je počeo crtati trokute prema sljedećem algoritmu:

- Nacrtamo početni jednakostranični trokut T.
- Dužinama povežemo polovišta njegovih stranica te dobijemo četiri nova trokuta koja označimo sa T₁, T₂, T₃ i T₄ kao na slici.
- Na trokutima T₁, T₂ i T₃ ponovimo postupak, dobivajući nove trokute T₁₁, T₁₂, T₁₃, T₁₄, T₂₁, T₂₂, T₂₃, T₂₄, T₃₁, T₃₂, T₃₃, T₃₄.
- S trokutima koji završavaju sa 1, 2 ili 3 nastavljamo postupak dalje u beskonačnost. Tako dobiveni geometrijski lik naziva se trokut Sierpinskog.



Kažemo da se trokut A **prislanja** na trokut B ako B **ne sadrži** A i ako postoji stranica od A koja je **čitava ujedno i dio neke stranice** od B. Na primjer, trokut T₂₃ se prislanja na trokute T₂₄ i T₄, ali se ne prislanja na trokute T₂ i T₃₂. Korisno je primijetiti da ako se trokut A prislanja na trokut B, ne mora nužno vrijediti i da se trokut B prislanja na trokut A.

Napišite program koji će za zadani trokut pronaći **sve** trokute na koje se on prislanja.

ulazni podaci

U prvom i jedinom retku nalazi se niz znakova koji predstavlja zadani trokut. Niz će se sastojati od najmanje dva i najviše 50 znakova.

izlazni podaci

Potrebno je ispisati sve trokute na koje se prislanja zadani trokut, svaki u svoj redak, bilo kojim redoslijedom.

test primjeri

ulaz

T4

izlaz

T1

T2

T3

ulaz

T11

izlaz

T14

ulaz

T312

izlaz

T4

T314

T34

lampice

Hrpa lampica složena je u tablicu koja se sastoji od dva retka i N stupaca. Svaka lampica može biti upaljena ili ugašena.

Počevši od stanja u kojem su sve lampice ugašene, želimo neke od njih upaliti tako da čine prekrasan uzorak. U jednom koraku možemo **promijeniti stanje** nekog niza **uzastopnih** lampica (jedne ili više njih) u istom retku ili istom stupcu.

Napišite program koji će za zadani uzorak odrediti koliko je **najmanje** koraka potrebno da od ugašenih lampica dobijemo taj uzorak.

Donja tablica prikazuje kako u sedam koraka dobijemo uzorak iz trećeg test primjera:

0 00000000000000000000 00000000000000000000	1 11100000000000000000 00000000000000000000	2 11100010000000000000 00000010000000000000	3 11100010000000000000 01111101100000000000
4 11101101111000000000 01111101100000000000	5 11101101111000111110 01111101100000000000	6 11101101111000101110 01111101100000010000	7 11101101111000101010 01111101100000010100

ulazni podaci

U prvom retku se nalazi prirodni broj N , $1 \leq N \leq 10,000$, broj stupaca u tablici.

U svakom od sljedeća dva retka nalazi se niz od N znakova, koji predstavljaju stanje odgovarajuće lampice u uzorku.

Lampice koje na kraju trebaju biti upaljene su označene znakom '1', dok su lampice koje trebaju biti ugašene označene znakom '0'.

izlazni podaci

U prvi i jedini redak potrebno je ispisati traženi minimalni broj koraka.

test primjeri

ulaz

3
100
000

izlaz

1

ulaz

5
11011
11011

izlaz

3

ulaz

20
11101101111000101010
01111101100000010100

izlaz

7

šibice

Šibice su složene tako da čine tablicu od tri retka i tri stupca kao na slici.

```
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
```

Dva uzastopna znaka '-' (minus) predstavljaju jednu vodoravno postavljenu šibicu, dok dva uzastopna znaka '|' (okomita crta) predstavljaju jednu okomito postavljenu šibicu. Znakovi '+' (plus) označavaju pozicije gdje se šibice dodiruju krajevima dok je unutrašnjost tablice popunjena znakovima '.' (točka).

Napišite program koji će, za zadane brojeve N i K, **ukloniti točno N šibica** tako da preostale šibice čine **točno K kvadrata** i to tako da **svaka preostala šibica** bude dio **barem jednog kvadrata**.

ulazni podaci

U prvom retku se nalaze dva prirodna broja N i K, $1 \leq N < 24$, $1 \leq K < 14$, broj šibica koje treba ukloniti i broj potrebnih kvadrata.

izlazni podaci

Potrebno je (u 10 redaka od po 10 znakova) ispisati završnu tablicu koja je jednaka tablici iz teksta zadatka, osim što su znakovi koji odgovaraju uklonjenim šibicama zamijenjeni znakom '.' (točka).

Napomena: ulazni podaci će biti takvi da će rješenje, iako ne nužno jedinstveno, uvijek postojati.

test primjeri

ulaz

20 1

izlaz

```
+--+-+ . . + . . +
|. . |. . |. . |
|. . |. . |. . |
+--+-+ . . + . . +
. . . . . . . . . .
. . . . . . . . . .
+. . +. . +. . +
. . . . . . . . . .
. . . . . . . . . .
+. . +. . +. . +
```

ulaz

5 4

izlaz

```
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
|. . . . |. . |
|. . . . |. . |
+--+-+--+-+--+
|. . . . . . |
|. . . . . . |
+--+-+--+-+--+
```

ulaz

4 6

izlaz

```
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
|. . |. . |. . |
|. . |. . |. . |
+--+-+--+-+--+
|. . . . . . |
|. . . . . . |
+--+-+--+-+--+
```