V zaporedju n celih števil poišči največje število in ga izpiši. Izpiši tudi število največjih števil, ki se pojavijo v zaporedju.

Vhod: V prvi vrstici vhoda je zapisana dolžina zaporedja $n \geq 1$. V drugi vrstici vhoda prebereš n celih števil, ločenih s presledki. Privzameš lahko, da je $n \leq 1000$ in da so vsa števila med -1000 in 1000.

Izhod: V prvo vrstico izhoda izpiši največjega med prebranimi števili. V drugo vrstico izhoda napišeš naravno število $k \geq 1$, ki pove, kolikokrat se največje število v danem zaporedju pojavi. Izpisani vrstici naj ne vsebujeta nepotrebnih presledkov.

Zgled vhoda:

```
10
18 -22 6 -34 12 44 31 44 12 44
```

Zgled izhoda:

```
44
3
```

Marko Knjigožer zelo rad bere knjige. Sam je prebral že skoraj polovico šolske knjižnice, zato je povprašal prijateljico Majo, če mu lahko priporoči kakšno novo zanimivo knjigo za poletno branje na plaži. Maja je prijatelju seveda priskočila na pomoč in mu sestavila seznam njenih najljubših knjig. Sama je precej natančna oseba, zato je Marku namesto naslovov knjig zapisala ISBN (International Standard Book Number) številke. Nekega dne pride Marko z listom v mestno knjižnico, a ugotovi, da nekaterih knjig po ISBN številki ne najde v bazi mestne knjižnice. Se je Maja kljub natančnosti zmotila pri prepisovanju ali knjižnica res nima teh knjig? Pomagaj Marku odgovoriti na to vprašanje.

ISBN številka je sestavljena iz desetih znakov $(a_1, a_2, ..., a_{10})$. Na prvih devetih mestih so števke med `0` in `9`, zadnji znak (a_{10}) pa je kontrolni. Kontrolni znak je lahko števka med `0` in `9` ali znak `X`. V primeru, da je kontrolni znak `X`, ga obravnavamo kot številsko vrednost 10. ISBN številke imajo posebno lastnost, s katero lahko ugotovimo, ali je številka veljavna ali ne. Za veljavne ISBN številke namreč velja, da je vsota produktov posameznega znaka in njegovega indeksa $(i \cdot a_i)$ deljiva s številom 11. Za veljavne ISBN številke torej velja formula:

$$\left(\sum_{i=1}^{10} i \cdot a_i\right) \mod 11 = 0.$$

Napiši program, ki preverja ISBN številke in pri neveljavnih številkah popravi kontrolni znak.

Vhod: V prvi vrstici vhoda je zapisano naravno število $n \geq 1$, ki pomeni število prihajajočih vrstic z domnevnimi ISBN številkami. V vsaki izmed naslednjih n vrstic je zapisanih po 10 znakov. Vsi znaki v vrstici, razen zadnjega, so števke med `0` in `9`, zadnji (deseti) znak pa je bodisi števka bodisi `X`. Privzameš lahko, da je n manjši od 1000.

Izhod: Izpiši natanko n vrstic. Za vsako vrstico vhoda, ki predstavlja ISBN številko, izpiši besedico `OK`, če je vhodna ISBN številka veljavna, oziroma popravljeno ISBN številko, če vhodna ISBN številka ni veljavna. Neveljavno ISBN številko popraviš tako, da ustrezno spremeniš kontrolni znak.

Zgled vhoda:

```
5
0201403757
0120884070
0691113211
235467654X
0387001638
```

Zgled izhoda:

```
OK
OK
OK
2354676549
OK
```

Molekula DNK je sestavljena iz zaporedja nukleotidov, ki jih označimo s črkami `A`, `T`, `G` in `C`. Določena zaporedja nukleotidov se v molekuli DNK ponovijo, ena večkrat, druga manjkrat. Zanima nas, kolikokrat se ponovi tisto zaporedje nukleotidov, ki vsebuje vsaj dva nukleotida in se ponovi največkrat.

Napiši program, ki sprejme poljubno dolgo zaporedje črk `A`, `T`, `G` in `C` ter izpiše število ponovitev tistega zaporedja z najmanj dvema nukleotidoma, ki se ponovi največkrat.

Vhod: Na vhodu naj program pričakuje eno samo vrstico, v kateri je zaporedje črk `A`, `T`, `G` in `C`. Na koncu zaporedja pa je znak za konec vrstice. Dolžina vrstice ni omejena, predpostavite pa lahko, da vrstica vsebuje vsaj dve črki.

Izhod: Program naj v prvo vrstico izpiše število ponovitev tistega zaporedja z najmanj dvema nukleotidoma, ki se ponovi največkrat.

Zgled vhoda:

```
ATTAT
```

Zgled izhoda:

```
2
```

Biolog Bine je postavil naslednji model razmnoževanja zajcev. Recimo, da zajklja vsak mesec skoti novo zajkljo. Zajklja po enem mesecu še nima svojih potomcev, po dveh mesecih pa že. Prav tako naj bodo zajklje nesmrtne. Kako hitro bi naraščalo njihovo število, če v prvem mesecu dobimo eno novorojeno zajkljo? Potem bi imeli v drugem mesecu še vedno le njo. V tretjem mesecu bi skotila novo zajkljo in imeli bi že dve. V naslednjem mesecu bi prva zajklja skotila še eno, medtem ko druga še ne bi bila plodna, torej bi imeli tri zajklje. V petem mesecu bi prvi dve zajklji rodili novi dve zajklji, tretja pa še nobene; torej bi imeli pet zajkelj, itn. Bineta je zanimalo, koliko zajkelj bi imel v n-tem mesecu. Za pomoč se je obrnil na kolega matematika. Ta mu je povedal, da iskana števila tvorijo Fibonaccijevo zaporedje, za katerega velja naslednja rekurzivna formula

$$Z_{n+1} = Z_n + Z_{n-1},$$

kjer smo z Z_n označili število zajkelj v n-tem mesecu. Toda ubogemu Binetu tudi to ni pomagalo, kajti nikoli ni bil najbolj vešč v računanju. Zato te prosi, da mu napišeš program, ki bo namesto njega računal število zajkelj v n-tem mesecu.

Vhod: V prvi vrstici je zapisano število testnih primerov t, ki je vedno manjše od 1000. Nato sledi t vrstic. V vsaki vrstici je zapisano število mesecev m, ki je manjše od štirih let.

Izhod: Za vsak primer izpiši število zajkelj po m mesecih. Rezultat vsakega primera naj bo v svoji vrstici. Privzameš lahko, da so rezultati vedno manjši od 2^{32}.

Zgled vhoda:

```
3
1
2
6
```

Zgled izhoda:

```
1
1
8
```

Ko je Andrej prišel v prvi letnik, je ugotovil, da pozna bolj malo sošolcev. Podobno pa se godi tudi z ostalimi. Zato ga je zanimalo, koliko takih trojic sošolcev lahko najde, da se bodo vsi poznali med sabo. Za lažji zapis je sošolce oštevilčil po vrsti od 1 dalje. Toda štetje se je izkazalo za zelo zamudno, zato te prosi, da mu napišeš program, ki bo to opravil namesto njega.

Vhod: V prvi vrstici je zapisano število sošolcev $s \leq 2007$ in število poznanstev $p \leq 3000000$, števili sta ločeni s presledkom. Nato sledi p vrstic. V vsaki vrstici je zapisan par števil a in b, ki pove, da se sošolca a in b poznata med sabo (poznanstva so vzajemna). Tudi ta par števil je ločen s presledkom.

Izhod: Izpiši, koliko iskanih trojic obstaja.

Zgled vhoda:

```
5 7
2 3
3 4
4 5
1 5
2 1
1 3
2 4
```

Zgled izhoda:

```
2
```

Pitagorov izrek pravi, da stranice vsakega pravokotnega trikotnika zadoščajo enačbi

$$a^2 + b^2 = c^2.$$

Če so a, b in c naravna števila, rečemo, da sestavljajo pitagorejsko trojico. Znani matematik Euler je postavil vprašanje, ali obstaja kvader, katerega stranice, stranske diagonale in telesna diagonala so naravna števila. Toda na to vprašanje še vedno ne poznamo odgovora. Matematik Matej, ki se je lotil reševanja tega problema, je ugotovil, da bo brez računalniškega programa naloga zelo težka, zato se je obrnil nate. Za začetek bi rad poiskal čim več pitagorejskih trojic. Prosi te, da mu za dano naravno število n izpišeš število vseh pitagorejskih trojic, ki tvorijo pravokotne trikotnike z obsegom n. Prosi te še, da ne razlikuješ med trojicami z istimi števili. Tako trojici $(3, 4, 5)$ in $(4, 3, 5)$ predstavljata isto pitagorejsko trojico. Zato imamo le en pravokotni trikotnik s celoštevilskimi dolžinami stranic in obsegom 12.

Vhod: V prvi vrstici je zapisano število testnih primerov t, ki je vedno manjše od 10000. Nato sledi t vrstic. V vsaki je zapisano naravno število $n \leq 2007$, za katerega nas zanima število ustreznih pitagorejskih trojic.

Izhod: Za vsako podano število n izpiši število pitagorejskih trojic, ki so stranice pravokotnega trikotnika z obsegom n. Rezultati naj bodo vsak v svoji vrstici.

Zgled vhoda:

```
3
20
12
420
```

Zgled izhoda:

```
0
1
5
```

Dano je zaporedje naravnih števil

$$n_1, n_2, n_3, \ldots, n_k.$$

Napiši program, ki

(a) števila zapiše v dvojiškem zapisu in

(b) poišče par (ne nujno sosednjih) števil n_i, n_j, $i \neq j$, katerih produkt ima v dvojiškem zapisu največ enic.

Vhod: V prvi vrstici vhoda je znak `a` ali `b`, ki pove, kateri del naloge naj program reši. V drugi vrstici vhoda je dolžina zaporedja k. V tretji vrstici vhoda so elementi zaporedja, ločeni s presledki. Predpostaviš lahko, da je $k \leq 1000$ in da so vsi elementi zaporedja manjši ali enaki 10000.

Izhod: Če je v prvi vrstici vhoda znak `a`, naj izhod vsebuje zadnje število zaporedja n_k, zapisano v dvojiškem zapisu. Sicer, če je v prvi vrstici vhoda znak `b`, naj izhod vsebuje maksimalno število enic v produktu dveh števil iz podanega zaporedja. Pazi, da na izhodu ne bo nepotrebnih presledkov in praznih vrstic!

Zgled vhoda:

```
b
4
2 5 3 4
```

Zgled izhoda:

```
4
```

Palindrom je "beseda" (zaporedje znakov brez presledkov), ki se prebere enako naprej in nazaj. Eden najbolj znanih palindromov v slovenskem jeziku je

`pericarežeracirep`

Poišči najdaljši palindrom, ki ga lahko najdeš v zaporedju znakov.

Vhod: V prvi vrstici vhoda prebereš dolžino zaporedja znakov n. V drugi vrstici vhoda prebereš zaporedje n znakov, ki ne vključujejo presledkov. *Palindrom* je *strnjeno* (brez vmesnih izpuščenih črk) zaporedje znakov, ki se od začetka proti koncu prebere natanko tako kot od konca proti začetku.

Privzameš lahko, da je $1 \leq n \leq 10000$ in da so vsi znaki v zaporedju male črke `a`,...,`z` angleške abecede.

Izhod: Na izhod izpišeš dolžino najdaljšega palindroma, ki se nahaja v podanem zaporedju znakov.

Zgled vhoda:

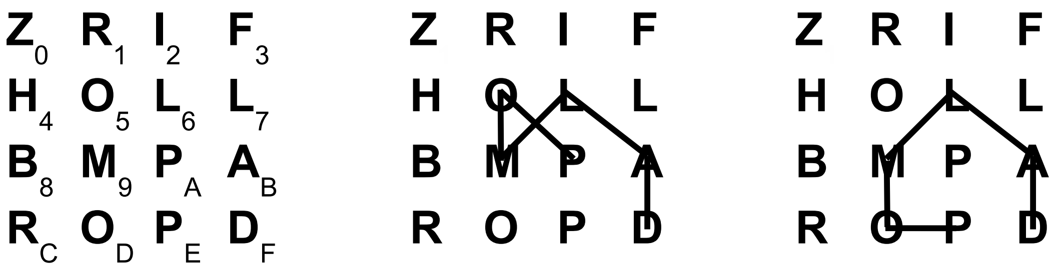
```
10
cabcbabbaa
```

Zgled izhoda:

```
5
```

Napiši program, ki v mreži znakov velikosti 4×4 poišče podano besedo. Besedo v mreži lahko sestavljajo črke, ki mejijo ena na drugo - vodoravno, navpično ali diagonalno proti levi, desni, navzgor ali navzdol. Posamezno črko mreže lahko uporabiš le enkrat.

Primer: če zaporedje črk ZRIFHOLLBMPAROPD razporedimo v mrežo 4×4 (glej sliko) in črke mreže oštevilčimo s števkami šestnajstiškega sistema *0, 1, ..., E, F*, se beseda POMLAD skriva npr. v zaporedjih črk na mestih *A, 5, 9, 6, B, F* in *E, D, 9, 6, B, F* (in še v nekaterih drugih zaporedjih).



`A596BF` `ED96BF`

Pravilna rešitev naloge je tisto zaporedje, ki predstavlja najmanjše šestnajstiško število. V zgornjem primeru je torej rešitev naloge prvo od omenjenih zaporedij, saj je `A596BF` manjše od `ED96BF` in tudi od vseh ostalih števil, ki predstavljajo preostala mesta, kjer se skriva `POMLAD`.

Vhod: Zaporedje šestnajstih črk mreže ter iskana beseda dolžine $k \geq 2$. Med zaporedjem znakov in iskano besedo je presledek.

Izhod: Najmanjše k-mestno šestnajstiško število, ki predstavlja tisto zaporedje črk v mreži, ki tvori iskano besedo (lahko se začne z `0`), oziroma `0`, če iskane besede v mreži ni.

Zgled vhoda:

```
ZRIFHOLLBMPAROPD POMLAD
```

Zgled izhoda:

```
A596BF
```

Kača se premika po pravokotnem polju dimenzije $m \times n$. Telo in rep kače sledita glavi po isti poti. Pot kače je dana z zaporedjem premikov glave. Pri vsakem premiku povemo, v kateri smeri in kako daleč se kača premakne. Zaporedje premikov je tako podano kot zaporedje znakov, ki opisujejo smer posameznega premika, in naravnih števil, ki pomenijo dolžino premika.

Na začetku vsakega premika se kača podaljša za 1 enoto. Predstavljaš si lahko, da na prvem koraku premika ostane rep na istem mestu.

Igra se konča, ko se kača zaleti v rob igralnega polja, sama vase ali pa opravi celotno zaporedje podanih premikov. Na začetku igre je kača ravna, podana sta položaj glave in repa.

Napiši program, ki za dano zaporedje premikov izračuna položaj glave in repa na koncu igre.

Podrobnosti pravil igre:

- Kača se zaleti sama vase, ko je glava na istem mestu kot katerikoli drug del telesa. Kača se zaleti v rob, ko njena glava zleze na rob.
- Predpostavimo lahko, da so zaporedni premiki med seboj pravokotni. Premiku v smereh gor-dol sledi premik v smeri levo ali desno in obratno.
- Kača se premika po korakih tako, da se na vsakem koraku hkrati premakne celo telo (z glavo in repom vred) za eno enoto v ustrezno smer.

Vhod: Spodnji levi kot igralnega polja postavimo v točko s koordinato $(1, 1)$. V prvi vrstici vhoda sta podani širina m in višina n igralnega polja. V drugi vrstici vhoda je začetni položaj kače. Zaporedoma so podane koordinate glave (x in y) in repa (x in y). Tretja vrstica vsebuje število premikov k, četrta vrstica zaporedje k znakov, ki označujejo smeri premikov (`L` za levo, `D` za desno, `g` za gor in `d` za dol). Peta, zadnja vrstica, vsebuje zaporedje k naravnih števil, ki pomenijo dolžine premikov. Števila in znaki v posamezni vrstici so ločeni s presledki. Predpostaviš lahko, da so vsa števila med 1 in 1000.

Izhod: Izhod naj vsebuje dve vrstici, ki vsebujeta koordinati x in y glave in repa ob koncu igre. V prvi vrstici naj bosta koordinati glave, v drugi pa koordianti repa. Števila naj bodo ločena z enim presledkom. Pazi, da izhod ne vsebuje nepotrebnih presledkov ali praznih vrstic!

V spodnjem zgledu kača prileze na zgornji rob igralnega polja, zato je v zgledu izhoda y-koordinata glave večja od višine igralnega polja.

Zgled vhoda:

```
8 10
6 5 6 2
5
L d D g L
3 4 4 10 3
```

Zgled izhoda:

```
7 11
7 4
```

Zaradi vse večje onesnaženosti okolja se je vlada odločila omejiti promet z motornimi vozili. Sprejeli so naslednji zakon: med tednom se lahko uporabljajo le tista motorna vozila, katerih registrska oznaka se da spremeniti v enačbo (glej definicijo in primere spodaj). Pri izvajanju zakona pa so policisti naleteli na težavo, saj v določenih primerih niso znali ugotoviti, ali je vozilo v prekršku ali ne. Napiši program, ki bo policistom pomagal poiskati kršitelje zakona.

Registrska oznaka vozila je sestavljena iz zaporedja velikih črk, presledka, števk in znaka `-`. Če iz registrske oznake odstranimo vse črke, presledke in znak `-`, dobimo tri- ali štirimestno število (morda vsebuje vodilne ničle). Registrska oznaka *se da spremeniti v enačbo*, če lahko to število z vrivanjem znakov aritmetičnih operacij $+$, $-$, $*$, $/$ in enačaja $=$ spremenimo v pravilno enačbo. Vrstnega reda števk v številu ne smemo spreminjati, znak $=$ lahko uporabimo le enkrat. Upoštevaj, da v enačbi ne moreš imeti dveh zaporednih aritmetičnih operacij. Na začetku enačbe kot tudi tik za enačajem pa lahko stoji minus: enačba `2*-3+7=1` ni pravilna, saj se minus pojavi za znakom `*`, enačba `-3=-4+1` pa je. Nekaj primerov:

`LJ A2-137` $\rightarrow 2137 \rightarrow 21 = 3 * 7 \rightarrow$ `DA`
`KR E3-42M` $\rightarrow 342 \rightarrow ? \rightarrow$ `NE`
`LJ V0-393` $\rightarrow 0393 \rightarrow 0 + 3 = 9/3 \rightarrow$ `DA`
`GO I7-832` $\rightarrow 7832 \rightarrow -7 + 8 = 3 - 2 \rightarrow$ `DA`
`MB I4-242` $\rightarrow 4242 \rightarrow 4 - 2 = 4/2 \rightarrow$ `DA`

Vhod: Program prejme pet registrskih oznak vozil, vsako v svoji vrstici.

Izhod: Za vsako registrsko oznako naj program po vrsti izpiše `DA`, če se oznaka da spremeniti v enačbo, sicer pa `NE`.

Zgled vhoda:

```
LJ A2-137
KR E3-42M
LJ V0-393
GO I7-832
MB I4-242
```

Zgled izhoda:

```
DA
NE
DA
DA
DA
```

V prvem oglišču n-kotnika sedi žaba in opazi, da se na m-tem oglišču nahaja muha. Žaba lahko skoči na katerokoli od sosednjih oglišč. Vsak žabin skok je dolžine 1. Na koliko načinov lahko žaba pride na m-to oglišče po natančno k skokih? Toda pazi — če bo žaba skočila na m-to oglišče prej kot v k skokih, bo muho že pojedla in ne bo več skakala naprej.

Vhod: V prvi vrstici je zapisano število testnih primerov $t \leq 200000$. Nato sledi t vrstic. V vsaki vrstici je zapisana trojica n, m, k, za katero velja $3 \leq n \leq 67$, $2 \leq m \leq 67$ in $1 \leq k \leq 33$. Števila n, m in k so ločena s presledki.

Izhod: Za vsako trojico n, m, k izpiši število različnih načinov, na katere lahko žaba poje muho po natanko k skokih. Rezultat vsakega primera naj bo v svoji vrstici.

Zgled vhoda:

```
3
4 2 3
6 3 3
6 3 6
```

Zgled izhoda:

```
2
0
9
```