

Het konijnenhok

Bij een foutgelopen experiment van dr. Cedric Parker-Powell (P.P. voor de vrienden) werden zijn proefkonijnen blootgesteld aan een overdosis radio-actieve straling. Op het eerste zicht leek er niets mis met de konijnen, maar bij nader onderzoek blijkt hun reproductiesysteem grondig ontregeld: dat verloopt nu in fasen.

Stel dat er op een gegeven ogenblik n mutant-konijnen in het hok zitten. Als n even is, dan gaan de konijnen in hongerstaking tot er nog maar de helft over is. Als n oneven is, dan kweken ze bij tot ze aangegroeid zijn tot $3n + 1$ konijnen. Dit proces herhaalt zich natuurlijk tot er nog maar 1 konijn over is – ze moeten met 2 zijn om zich voor te planten!

P.P. weet nu al dat zijn konijnenhok niet groot genoeg is om alle konijnen te herbergen. Kan jij hem vertellen op welk maximale aantal konijnen hij voorzien moet zijn? Let op, de konijnpopulatie vertoont rare sprongen. Bijvoorbeeld, vertrekkende van 7 konijnen, worden achtereenvolgens de volgende aantallen doorlopen:

7 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

Het hok moet in dit geval dus voorzien zijn op 52 konijnen!

Om P.P. te helpen zal je het initiële aantal konijnen moeten weten, maar al wat P.P. je kan vertellen is dat het ergens in het interval $[l, u]$ ligt.

Probleem

Wat is het grootste aantal konijnen dat er op een willekeurig moment kan voorkomen, gegeven dat het initiële aantal konijnen in het interval $[l, u]$ ligt?

Invoer

De invoer bestaat uit twee lijnen. Op de eerste lijn staat l en op de tweede lijn u , met $l, u \in [1, 500]$.

Uitvoer

De uitvoer bestaat uit één regel met een natuurlijk getal m dat het maximale aantal konijnen aangeeft.

Voorbeelden

Invoer:

1
2

Uitvoer:

2

Invoer:

2
3

Uitvoer:

16

Invoer:

3
10

Uitvoer:

52

Hoeken

Probleem

Schrijf een programma dat een ASCII figuur produceert. Het programma heeft een parameter n die de grootte van de figuur bepaalt. Gegeven zijn de figuren voor enkele waarden van n . Je moet zelf afleiden hoe de figuur schaalt met n .

Invoer

De invoer is een geheel getal $n \in [1, 100]$, dus inclusief 1 en 100.

Uitvoer

De gevraagde ASCII figuur moet afgedrukt worden op de standaard uitvoer. De uitvoer mag een of meer lege regels bevatten voor en na de regels van de eigenlijke figuur. Op het einde van elke regel mogen een willekeurig aantal spaties staan. Het aantal spaties aan het begin van de regel is wel van belang: het meest linkse * of | karakter staat tegen de linker marge en wordt niet voorafgegaan door een spatie.

Voorbeelden

Invoer:

1

Uitvoer:

```
* *  
*  
* *
```

Invoer:

2

Uitvoer:

```
** **
*  *
  **
  **
*  *
** **
```

Invoer:

3

Uitvoer:

```
*-*  *-*
|/   \|
*    *
  *-*
  | |
  *-*
*    *
|\   /|
*-*  *-*
```

Invoer:

4

Uitvoer:

```
*--*  *--*
| /    \|
| /    \|
*     *
  *--*
  | |
  | |
  *--*
*     *
|\    /|
| \   /|
*--*  *--*
```

Om te zorgen dat je zeker weet waar en hoeveel spaties er moeten staan, staat de uitvoer voor invoer 4 hierna nog eens met elke spatie vervangen door een b:

```
*--*bbbb*--*
|b/bbbbbbb|
|/bbbbbbbb|
*bbbbbbbbb*
bbbb*--*
bbbb|bb|
bbbb|bb|
bbbb*--*
*bbbbbbbbb*
|\bbbbbb|
|b\bbbbbb|
*--*bbbb*--*
```

Pachinko

Pachinko is een Japans spel dat wat lijkt op een flipperkast. Het spel is heel eenvoudig: je laat een bal bovenaan in één van de kolommen vallen, de bal valt naar beneden, botst eventueel tegen obstakels, en valt uiteindelijk in een opening met een bepaalde beloning. De bedoeling is natuurlijk om een zo groot mogelijke beloning te krijgen.

Probleem

Bepaal de kolom waarin de bal terecht moet komen om een maximale score te bekomen.

Invoer

De eerste twee lijnen van de invoer zijn respectievelijk het aantal rijen r en het aantal kolommen k van de pachinko-machine. De daarop volgende r lijnen geven de rijen van de machine. Elke rij bestaat uit k posities die de volgende waarde kunnen hebben:

- ‘.’ : lege ruimte, de bal valt hier gewoon door.
- ‘*’ : obstakel, de bal ketst schuin af naar de kolom juist links of rechts van het obstakel (elk met 50% kans). Maw., als de bal (voorgesteld door o) zich op een ogenblik in de linkse situatie bevindt, dan komt hij op het volgende ogenblik in een van de twee rechtse situaties:

```
.o.      ...      ...
.*.      ==>   o*.      of      .*o
...      ...      ...
```

Als er links (rechts) geen kolom is, dan gaat de bal natuurlijk enkel naar rechts (links). Links of rechts van een obstakel staat nooit een ander obstakel.

- ‘1’ ... ‘9’ : een beloning met waarde 1 ... 9, de bal stopt hier.

De bal stopt altijd op een beloning.

Je programma moet werken voor $r, k \in [1, 10]$.

Uitvoer

Het kolomnummer van een kolom met maximale verwachte opbrengst.

Voorbeelden

Invoer:

4
5
.....
.*...
..*..
78023

Uitvoer:

1

Invoer:

5
4
*...
.*..
..*.
...*
4848

Uitvoer:

3

De slimme postbode

Pietje Rolog, de postbode van Heverlee, bedielt de post in de Celestijnenlaan. Hij heeft het niet zo begrepen op die nieuwe efficiënte postrondes. Het liefst van al zou hij de hele dag bezig zijn met het bedelen van post in dezelfde straat. Hij heeft al een idee hoe hij dat gaat aanpakken. In plaats van de brieven te bezorgen in volgorde van oplopende huisnummers, gaat hij ze in een andere (hopelijk langere) volgorde bedelen. Jij gaat hem erbij helpen om de langst mogelijke route te vinden, of ten minste een langste route als er meerdere zijn. Er staan n huizen in de straat met huisnummers van 1 tot n . Je moet de huisnummers in die volgorde teruggeven waarvoor de totale afstand tussen de opeenvolgende huizen maximaal is. De afstand tussen twee huizen is het verschil in hun huisnummers.

Bijvoorbeeld, bij 3 huizen, zijn de huisnummers 1, 2 en 3. Als Pietje Rolog ze in volgorde bezoekt, legt hij een afstand af van $(2 - 1) + (3 - 2) = 2$ eenheden. De volgorde 1, 3, 2 is langer omdat hij dan $(3 - 1) + (3 - 2) = 3$ eenheden moet afleggen. Omdat er geen langere volgorde is, is dit dus een gezochte volgorde.

Merk op dat de afstand van het begin van de straat tot het eerst bezochte huis en van het laatst bezochte huis tot het einde van de straat niet meetellen. Pietje wordt namelijk 's morgens aan het eerste huis afgezet met het post-minibusje, en 's avonds aan het laatste huis opgepikt.

Probleem

Gegeven n , zoek een permutatie $(x_1 \cdots x_n)$ van $(1 \cdots n)$ waarvoor

$$\sum_{i=1}^{n-1} |x_{i+1} - x_i|$$

maximaal is.

Invoer

De invoer is een geheel getal $n \in [2, 8]$.

Bonus: Indien je programma ook elke $n \in [2, 100]$ aankan binnen 10s op onze testmachine, wordt 60 minuten van je tijd afgetrokken in de rangschikking.

Uitvoer

De uitvoer is een rij van gehele getallen gescheiden door spaties en afgesloten door een newline.

Voorbeelden

Invoer:

2

Uitvoer:

1 2

Invoer:

3

Uitvoer:

2 1 3

Invoer:

4

Uitvoer:

2 4 1 3

Invoer:

5

Uitvoer:

2 4 1 5 3

De robijn

Eén manier om je weg door een doolhof te vinden is om je *linkerhand* steeds in contact te houden met de muur van de doolhof. Op deze manier lukt het steeds om vertrekkend van een ingang terug bij dezelfde of een andere ingang uit te komen. De regel werkt echter niet altijd om de robijn in de doolhof te vinden.

We willen wel eens weten of de linkerhand-regel toelaat om de robijn te vinden voor een gegeven doolhof.

Probleem

Je krijgt een doolhof met juist één ingang die ook als uitgang dienst doet: die wordt aangeduid in de invoer als een opening 'O'. Daarnaast kunnen er bijkomende uitgangen zijn. Ga na vertrekkende bij de van de ingang of je onderweg de robijn kan zien, voor je weer bij een uitgang staat. Je moet telkens strikt de linkerhand-regel naleven. Verplaatsingen gebeuren enkel horizontaal en vertikaal, niet diagonaal, dus als X de positie is waar je je bevindt, dan:

```
...           ...           ...           .X.           ...
.X.   wordt   X..   of   ..X   of   ...   of   ...
...           ...           ...           ...           .X.
```

Je kan de robijn *zien* als er zich in rechte lijn (horizontaal of vertikaal) geen obstakel bevindt tussen jou en de robijn.

Invoer

De invoer bestaat uit een doolhof; de eerste invoerlijn heeft twee getallen: de breedte b en de hoogte h van de doolhof, met $b, h \in [4, 100]$.

De volgende h regels van de invoer bestaan telkens uit b karakters. De betekenis van deze karakters is de volgende:

- spatie: lege ruimte.
- 'R' is de robijn, er is juist 1 robijn in de doolhof.
- 'M' betekent muur.
- 'O' is een opening (zowel ingang als uitgang).

-
- 'U' is een uitgang.

De buitenste omtrek van de doolhof bestaat volledig uit 'M', 'O' en 'U' karakters. De 'O' en 'U' karakters komen nergens anders voor, ze staan nooit naast elkaar of in de hoeken van de doolhof.

Uitvoer

Druk 1 af als je op weg naar een uitgang de robijn kan zien. Anders druk je 0 (nul) af.

Voorbeelden

Invoer:

```

31 15
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
M                                     M
M                                     M
M                                     M
M   MMMM   MMMM                       M
M   M       M                           M
M   M   R   M                           M
M   M       M                           M
M   M       M                           M
M   MMMMMMMMM                           M
M                                     M
M                                     M
M                                     M
M                                     MMMMMMMMMMMMM
M                                     M
MMMMMMOMMMMMMMMMMMMMMMMMMMUMMMMMMM

```

Uitvoer:

```
1
```