

29 juli Programmering, Nao - Extrauppgifter

1.

Valentina tycker om choklad. Hon har därför alltid en öppnad chokladkartong i skafferiet. När den tar slut köper hon i hemlighet en ny och låtsas som ingenting. Valentinas bror Danil, som är misstänksam av naturen, förundras över att den där kartongen aldrig tar slut. Därför börjar han vid varje besök att räkna antalet chokladbitar som är kvar. Skriv ett program som, givet hans observationer, beräknar det minsta antalet nya kartonger Valentina måste ha köpt under perioden.

Indata

På första raden följer en rad med N heltal (lika många heltal som observationer), där heltalen motsvarar antalet chokladbitar i asken (mellan 1 och 100) vid varje observation, i den ordning de görs.

Utdata

Programmet ska skriva ut ett heltal: det minsta antal nya kartonger Valentina bevisligen måste ha köpt under perioden.

Körningsexempel:

Observationer ? 17 15 16 16 18 17 14 12 13 9

Valentina måste åtminstone ha köpt 3 chokladaskar.

Observationer ? 100 67 56 73 54 98 83 32 67 34 21 21 1 54 67 45 93 86 35 63 2

Valentina måste åtminstone ha köpt 7 chokladaskar.

Observationer ? 88 79 65 61 57 32 87 54 98 21 56 56 56 73

Valentina måste åtminstone ha köpt 4 chokladaskar.

2.

Leonid kandiderar till ordförandeposten i matematiksällskapet och vill inte riskera att förlora omröstningen. Han har lyckats få reda på vilken kandidat varje medlem tänker rösta på och tänker helt enkelt muta ett antal medlemmar så att de röstar på honom istället. Skriv ett program som beräknar hur många röster som måste köpas (d.v.s. medlemmar som behöver mutas) för att Leonid ska vinna omröstningen. För att vinna krävs att man får fler röster än var och en av de andra kandidaterna.

Indata Först ska man mata in antal kandidater ($1 \leq n \leq 20$). Sedan ska man mata in n heltal (mellan 0 och 1000): antalet röster varje kandidat skulle få utan mutor. Det första talet anger Leonids röster.

Utdata

Programmet ska skriva ut ett heltal: det minsta antalet röster som behöver köpas för att Leonid ska få fler röster än var och en av de övriga kandidaterna.

Körningsexempel:

Antal kandidater ? 4

Röster som Leonid skulle fått utan mutor ? 2

Röster som de andra skulle fått utan mutor ? 5 7 8

Leonid måste åtminstone köpa 5 röster.

Antal kandidater ? 6

Röster som Leonid skulle fått utan mutor ? 11

Röster som de andra skulle fått utan mutor ? 10 14 0 2 8

Leonid måste åtminstone köpa 2 röster.

Antal kandidater ? 9

Röster som Leonid skulle fått utan mutor ? 6

Röster som de andra skulle fått utan mutor ? 4 3 8 4 4 16 18 2

Leonid måste åtminstone köpa 8 röster.

3.

I egyptisk matematik hade de så kallade heltalsreciprokerna

$$\frac{1}{1}, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \dots$$

särskild betydelse. Övriga bråk skrevs som summor av dessa tal, t.ex.

$$\frac{21}{20} = \frac{1}{2} + \frac{1}{4} + \frac{1}{5} + \frac{1}{10}$$

Skriv ett program som tar emot ett bråk (förkortat så långt som möjligt) och avgör om det kan skrivas som en sådan summa, under villkoren att bara de tio första heltalsreciprokerna (alltså t.o.m. $1/10$) får användas, och att varje tal endast får användas en gång. Om det går ska programmet skriva ut de ingående termerna i summan, annars ska det skriva ut Omöjligt. Om det finns flera lösningar ska programmet skriva ut vilken som helst av dem.

Två körningsexempel:

Täljare ? 67

Nämnare ? 60

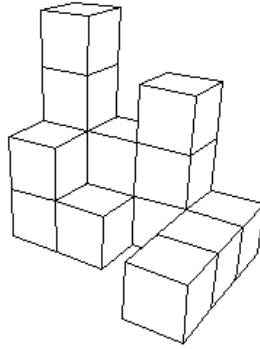
Termer: 1/2 1/4 1/5 1/6

Täljare ? 2

Nämnare ? 5

Omöjligt

4.



Figur 1

Betrakta den tredimensionella figuren ovan. Den är ihopsatt av ett antal $1 \times 1 \times 1$ kuber i ett 3d-rutnät. Vi begränsar oss till figurer som har staplar fästa i "marken", så ingen kub har luft under sig.

Figurens volym är förstås enkel att beräkna, men här är vi intresserade av dess begränsningsarea, d.v.s. antalet 1×1 kvadrater som är synliga utifrån (inklusive underifrån). Skriv ett program som beräknar detta, givet beskrivningen av en figur. Figurens basarea kan maximalt vara 50×50 , och staplarnas höjd är maximalt 50.

Körningsexempel:

Höjdinformation ?

4 2 3 1

2 1 0 1

0 0 0 1

Begränsningsarean blir totalt: 54

Höjdinformation ?

1 3

4 5

Begränsningsarean blir totalt: 42

Höjdinformation ?

4 6 2 7 8 3 5 12 9

5 7 6 2 4 3 1 4 7

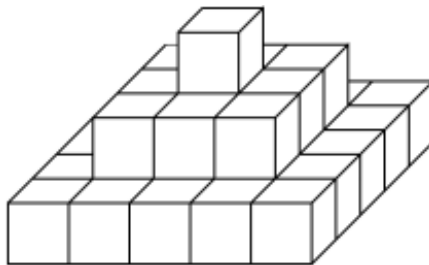
3 2 6 3 9 6 5 3 8

10 25 13 7 7 0 0 1 9

4 36 12 4 18 3 5 0 0

Begränsningsarean blir totalt: 658

5.



Figur 2: Denna pyramid med höjden 3 innehåller 35 stenblock. Cheopspyramiden i Egypten har höjden 210.

När man ska inleda ett större projekt, exempelvis bygga en pyramid, är det bäst att tänka efter en gång extra. Du ska skriva ett program som beräknar hur hög pyramid man kan bygga om man har tillgång till ett visst antal stenblock.

Vi antar att pyramiden är kompakt, d.v.s. det finns inga hålrum inuti. Vidare byggs den enligt principen i figuren ovan. Varje lager är alltså kvadratisk med en sidlängd som är två block mindre än det underliggande lagrets. Det översta lagret består alltid av ett ensamt block.

Programmet ska fråga efter antalet tillgängliga block (högst hundra miljoner) och skriva ut höjden (i block räknat) för den största pyramid som kan byggas. Det gör ingenting om det blir block över, men det får inte saknas ett enda block.

Körningsexempel:

Antal block ? 83

Höjd: 3

Antal block ? 165

Höjd: 5

Antal block ? 456

Höjd: 7

Antal block ? 12347930

Höjd: 210

6.

En sphinx är en varelse med ett lejons kropp och en människas huvud. Liknande kombinationer återfinns i andra kulturer, t.ex. faunen (get+människa) och gripen (örn+lejon). Man skulle kunna tro att den ringa förekomsten av sådana arter i naturen beror på viss svårighet i fortplantningen, men i denna uppgift antar vi raka motsatsen.

Med hänvisning till genetikens lagar antar vi att när två kombinationsvaror parar sig så ärver avkomman framkroppen från den ena föräldern och bakkroppen från den andra. En grodmus som parar sig med en fiskhöna kan alltså ge upphov till antingen en grodhöna eller en fiskmus.

Skriv ett program som, givet en grupp kombinationsvaror (sommiga hannar och andra honor) beräknar antalet olika arter som kan existera i nästa generation om alla par av hane-hona antas få ungar. Svaret ska också inkludera de arter som ingår i den ursprungliga gruppen.

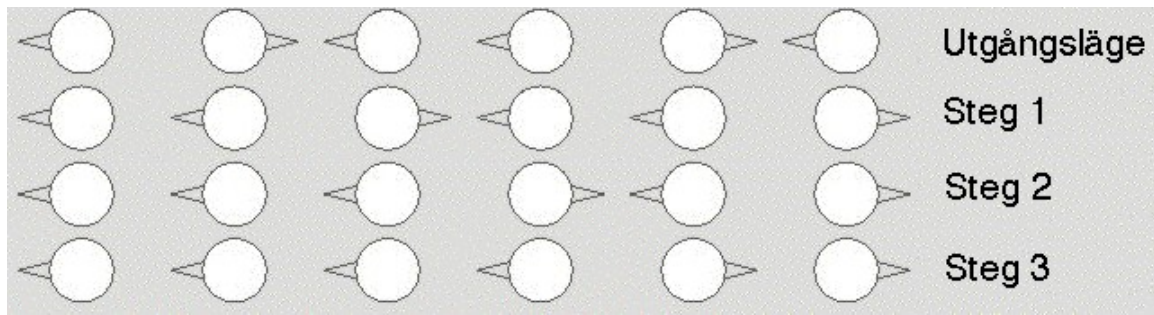
Programmet ska fråga efter antalet hannar och vilken art varje hane tillhör, samt likadant för honorna. Arten anges som en sträng med två bokstäver (valda bland A-Z), där första bokstaven beskriver framkroppen och andra bokstaven bakkroppen (t.ex. ML för människolejon). Programmet ska skriva ut det totala antalet olika arter som kan finnas när parning har skett. Observera att ordningen på bokstäverna spelar roll – MF och FM är olika arter.

Körningsexempel:

```
Antal hannar ? 2
Hanne 1 ? ML
Hanne 2 ? FG
Antal honor ? 3
Hona 1 ? VM
Hona 2 ? FM
Hona 3 ? VF
Antal arter: 11
```

```
Antal hannar ? 5
Hanne 1 ? HM
Hanne 2 ? FB
Hanne 3 ? OH
Hanne 4 ? GO
Hanne 5 ? SO
Antal honor ? 6
Hona 1 ? BF
Hona 2 ? GM
Hona 3 ? VM
Hona 4 ? VF
Hona 5 ? FG
Hona 6 ? HM
Antal arter: 36
```

7.



Figur 3: Uppställning av soldater

Ett antal soldater står på ett led alla vända mot furiren, när han ger ordern "höger om". Eftersom många av soldaterna har svårt att skilja på höger och vänster blir det mest slumpen som avgör åt vilket håll de vänder sig. Ett exempel visas på första raden i figuren ovan. De soldater som på så sätt hamnar "öga mot öga" med en granne förstår båda två att de vänt sig åt fel håll och gör därför helt om (180 grader), för att kanske hamna öga mot öga med den andra grannen. Denna procedur fortsätter (steg 1 och 2 i figuren) och upphör då inga soldater längre är vända mot varandra (steg 3).

Indata

En rad där varje tecken är antingen V eller H. Dessa anger åt vilket håll soldatens näsa pekar i utgångsläget. Maximalt 100 tecken.

Utdata Programmet ska skriva ut hur många steg som behövs från utgångsläget tills lugnet infinner sig.

Körningsexempel:

Soldater? VHVVHV

Det behövs totalt 3 steg för att lugna soldaterna

Soldater? HVHVHVHHVHVHVHVHHHHVHHVHVVVHHVHHVHVHVHVHVHHH

Det behövs totalt 24 steg för att lugna soldaterna
