Uppgift 1 – Euro

Harry ska åka till Portugal och behöver växla till sig 500 Euro från svenska kronor. När han kommer tillbaka från Portugal kommer han att ha 200 Euro över som han vill växla tillbaka till kronor. Harry vill naturligtvis göra de båda växlingarna så förmånligt som möjligt, så han har jämfört några olika banker (högst 5 stycken).

Vi antar att varje bank har en växlingskurs som inte ändrar sig under perioden och som är samma för in- och utväxling. Dessutom tar varje bank en avgift, som är angiven i hela kronor. Även avgiften är samma för in- och utväxling, men om man gör båda växlingarna på samma bank så betalar man avgift endast den ena gången.

Skriv ett program som avgör vilken bank Harry ska välja före respektive efter resan. Han vill alltså att skillnaden mellan det antal kronor han betalar för 500 Euro och det antal kronor han får tillbaka för 200 Euro ska vara så liten som möjligt.

Indata: Programmet ska föra en dialog, som i följande exempel. Kursen ges som ett tresiffrigt heltal, där 976 betyder att 1 Euro kostar $9,76$ kronor.

```
Antal banker: 3
Bank 1, kurs: 975
Bank 1, avgift: 20
Bank 2, kurs: 976
Bank 2, avgift: 18
Bank 3, kurs: 981
Bank 3, avgift: 10
```

Utdata: Två rader, som för indata ovan, ser ut som följer:

```
Före resan: bank 1
Efter resan: bank 3
```

Anmärkning: Med dessa val blir skillnaden 2943 kronor. Om bank 2 hade använts för båda växlingarna hade skillnaden blivit 2946 kronor.

För varje test finns det bara en lösning. Det vill säga två olika växlingar kan inte ge *samma minsta differens*.

UPPGIFT 2 – PRIMFAKTORER

Ett primtal är ett tal, som är större än 1 och endast är delbart med 1 och sig självt (t.ex. 7 och 13). Varje annat tal kan skrivas som en produkt av ett antal primtal. Detta kan ske på exakt ett sätt och de enskilda primtalen i en sådan produkt kallas primfaktorer. Till exempel $60 = 2 \cdot 2 \cdot 3 \cdot 5$ och $91 = 7 \cdot 13$.

Betrakta nu följande procedur.

- Utgå från ett *icke-primtal*.
- Räkna ut talets primfaktorer.
- Summera primfaktorerna så att ett nytt tal erhålls.
- Upprepa proceduren tills det erhållna talet är ett primtal.

Om exempelvis talet är 60 blir alltså nästa tal $2+2+3+5 = 12$. $12 = 2 \cdot 2 \cdot 3$, vilket ger nästa tal $2+2+3 = 7$. 7 är ett primtal, alltså avslutas serien.

Om man börjar med ett tal större än 4, kan man bevisa att varje tal i en sådan serie är mindre än det föregående. Skriv ett program som tar emot ett *icke-primtal* $N, (4 < N < 10000)$, genomför ovanstående procedur och skriver ut det primtal som avslutar serien.

Indata: Ett tal, det första i serien, skrivs in

```
Starttal: 60
```

Utdata: Ett tal, det sista i serien, skrivs ut

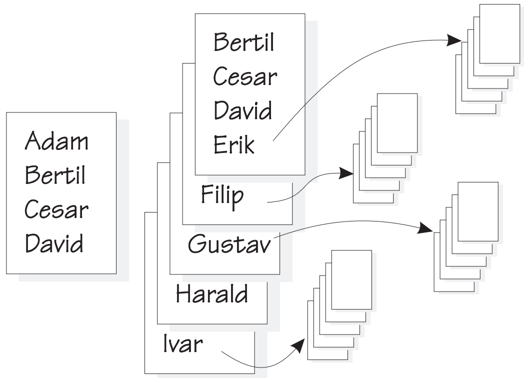
```
Sluttal: 7
```

Ett exempel till för säkerhets skull.

```
Starttal: 5019
Sluttal: 11
```

Resultatet får man efter följande serie $5019 \rightarrow 249 \rightarrow 86 \rightarrow 45 \rightarrow 11$

Uppgift 3 – Kedjebrev



FIGUR 1. *Till vänster ser vi det ursprungliga brevet. David skickar fem brev – till Erik, Filip, Gustav, Harald och Ivar. Endast Harald hoppar av och spelet går vidare*

Kedjebrev är en företeelse som vandrat jorden runt i långliga tider. Idéen går ofta ut på att tjäna pengar. Allt startar med att n personer skriver ner sina namn och adresser i ett brev. Den som står sist i listan gör m kopior och skickar dessa till m olika personer. De som erhåller dessa brev ska skicka 1 krona till den, som står först i listan, skriva om brevet och samtidigt ta bort översta namnet i listan och skriva in sitt eget längst ned, på plats n. Sedan kopieras breven och skickas till m personer.

I figur **??** ser vi ett exempel där $n = 4$ och $m = 5$. Om ingen bryter kedjan kommer Adam att få 5 kronor, Bertil 25, Cesar 125 och David hela 625 kronor. Nu brukar dock inte dessa kedjebrev bli *så* lyckade, därför att personer helt enkelt inte vill vara med – det accepterar vi. Vi antar dock, att de personer som vill vara med är helt ärliga och verkligen skickar sin krona och dessutom sänder brev till m personer. Vi kan anta att ingen deltagare på den ursprungliga listan får något brev, eller att någon annan av deltagarna få fler än ett.

Skriv ett program som tar emot uppgifter om $n, 2 \leq n \leq 6$, antal namn på listan och $m, 2 \leq m \leq 10$, antal brev som skickas ut av varje person. Dessutom ska programmet ta emot uppgifter om hur mycket pengar personerna på den ursprungliga listan verkligen erhöll. Ur denna information ska ditt program beräkna hur många personer som ”hoppade av” under de n första omgångarna.

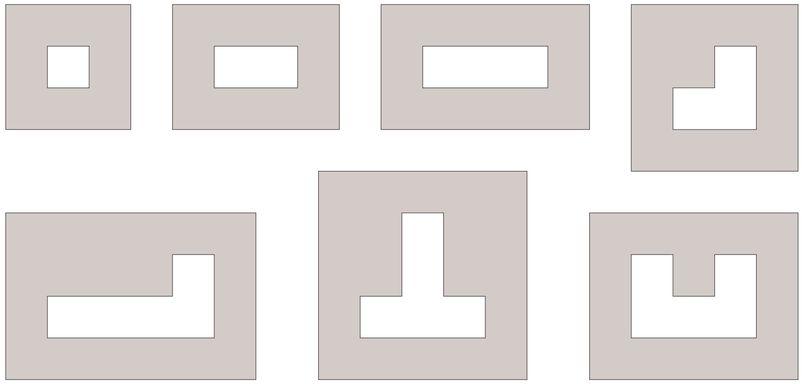
Indata:

```
Antal brev som skickas av varje deltagare: 5
Antal namn i lista: 4
Belopp till person 1: 4
Belopp till person 2: 20
Belopp till person 3: 60
Belopp till person 4: 100
```

Utdata:

```
241 personer har hoppat av
```

Uppgift 4 – Tomter till husen



FIGUR 2. *Överst till vänster ser vi tomt med Friggebod. Sedan följer den enda tomten med ett 20m^2 hus och de två möjliga tomterna med ett 30m^2 hus. I den undre raden ser vi de tre av fyra möjligheter för en tomt med ett 50m^2 hus.*

Som stadsplanerare är din arbetsdag ofta upptagen av problem kring villor och villatomter. Det har på sistone blivit allt vanligare med ”modulhus” – hus, som på kartan, i grunden består av ett antal kvadrater (10m^2). Ett hus på 10m^2, en så kallad Friggebod, kan då bara ha en form. Samma för hus på 20m^2. Men för 30m^2-hus finns det två olika former. För att spara tomtmark finns ett fullmäktigebeslut om att tomten till ett hus hus ska ha rektangulär form och omsluta huset med en ”ring” av ”modulkvadrater”. Det betyder att den enda friggebodstomten är på 90m^2, att 20m^2-huset alltid har en tomt på 120m^2. För 30m^2-husen finns två olika tomtareor, 150m^2 och 160m^2. Varje, i ett hus ingående, modul måste ha minst en sida gemensam, med en eller flera andra moduler (utom då förstås Friggeboden).

Då husets storlek växer ökar också antalet olika tomtareor. För din egen hjälp behöver Du nu ett program som tar emot uppgift om husets area och som beräknar antalet olika tomtareor som kan komma ifråga för denna storlek.

Indata: Programmet frågar efter husets area, ett tal ur mängden $[10, 20, 30 \ldots 180, 190, 200]$, jämna 10-tal kvadratmeter.

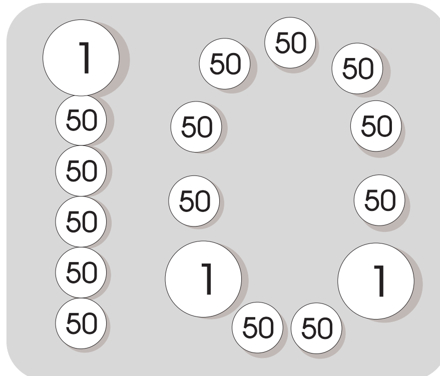
```
Husets area: 50
```

Utdata: Programmet skriver ut antalet olika areor som ges till den givna husarean

```
Det finns 4 olika tomtareor
```

Anmärkning: De fyra areorna är 200, 240, 250 och 210 (ej med i figur **??**)

UPPGIFT 5 – PRISER MED MYNT



FIGUR 3. *Priset* 10 *kronor skrivet med* 50*-öringar och enkronor. Här behövdes alltså inte extramyntet!*

I reklamkampanjer kan det vara trevligt att kunna skriva ett pris med hjälp av utlagda mynt på ett sådant sätt att myntens sammanlagda värde är detsamma som det skrivna priset (som i exemplet i figur **??**).

Din uppgift är att med hjälp av ett program avgöra hur många olika belopp (endast heltal) som kan skrivas på detta sätt. Till din hjälp har du *enkronor*, 50*-öringar* samt ett *extra mynt* (ett helt antal kronor ≤ 20), vars värde programmet ska fråga efter. Alla mynten finns i obegränsat antal och du behöver inte använda alla sorterna.

Du får dock inte forma siffrorna precis som du vill. En ”sifferformskonsult” har gett dig en tabell över hur många mynt som ska ingå i varje siffra för att den ska se snygg ut. Antalet påverkas inte av myntsorterna.

Nolla	**Etta**	**Tvåa**	**Trea**	**Fyra**	**Femma**	**Sexa**	**Sjua**	**Åtta**	**Nia**
11 mynt	6 mynt	9 mynt	11 mynt	12 mynt	10 mynt	8 mynt	8 mynt	11mynt	9 mynt

Indata: Programmet frågar efter det *extra myntets* värde.

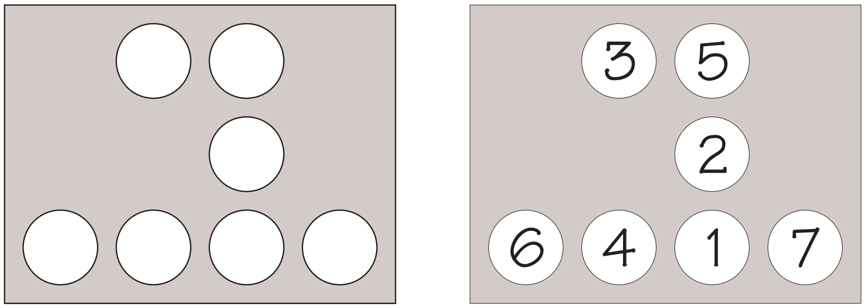
```
Extra myntets valör: 6
```

Utdata: Programmet skriver ut antalet funna belopp.

```
Antal belopp: 116
```

Anmärkning: I exemplet är de första respektive sista skrivbara talen
5, 6, 7, 8, 9, 10 . . . 138, 142, 143, 144, 146, 148

Uppgift 6 – Avstånd mellan tal



Figur 4. *Till vänster ser vi den rektangulära piltavlan och till höger lösningen.*

Den senaste attraktionen på vårt nöjesfält är ”Den Nya Pilkastningen”. Tillhörande tavla ser vi längst upp till vänster i figur **??**. Den är rektangulär, med plats för 12 ringar (3 rader med 4 ringar i varje), men där 5 av dem har tagits bort.

Tävlingen går ut på att med 7 pilar träffa var och en av ringarna med exakt en pil. Men det räcker inte! För att vinna den stora teddybjörnen, måste du träffa ringarna i en sådan ordning, att avståndet mellan den just träffade ringen r_n och föregående r_{n-1} är kortare, än avståndet mellan r_n och nästa ring r_{n+1}. Avstånden mellan ringarna är avstånden mellan deras centrum. Problem kan förstås inte uppstå förrän man ska till att kasta den tredje pilen.

Till höger i figur **??** ser vi den enda lösningen. Samtidigt ser vi att avståndet mellan 1 och 2 är kortare än avståndet mellan 2 och 3 och att avståndet mellan 5 och 6 är kortare än mellan 6 och 7.

Eftersom intresset för ”Den Nya Pilkastningen” blivit mycket stort, har man bestämt sig för att ta fram nya tavlor och Du har blivit ombedd att skriva ett program som finner lösningar till föreslagna tavlor. Man ska alltid kasta lika många pilar som det finns ringar i tavlan.

Indata: En dialog som först frågar efter antalet ”rader” $n, 2 \leq n \leq 6$ och antalet ”kolumner” $m, 4 \leq m \leq 8$ hos tavlan och sedan för varje rad beskriver utseendet med en sträng, innehållande exakt m tecken. Punkt (`.`) för tom plats och `r` för ring.

```
Antal rader: 3
Antal kolumner: 4
Rad 1: .rr.
Rad 2: ..r.
Rad 3: rrrr
```

Utdata: Består av en tabell med n rader, med m tal i varje. Talet 0 används för ”ring saknas”. Övriga tal anger ordningsnumret för pilen. Du kan anta att varje test har exakt en lösning.

```
0 3 5 0
0 0 2 0
6 4 1 7
```