

## Programmering 1

### Laboration 5: Räckor

Denna laboration är tänkt att både utföras och **redovisas** under de schemalagda timmar som finns anslagna för ändamålet. Ifall du inte har möjlighet att delta i laborationstillfällena, eller inte hinner slutföra laborationen under den schemalagda tiden måste du redovisa dina resultat i en **skriftlig laborationsrapport** som skickas in via Moodle. Ett obligatoriskt delkrav för att kunna bli godkänd i kursen är att **alla laborationer är godkända**.

**Obs: Vissa uppgifter (markerade med en asterisk) kommer att gås igenom gemensamt under laborationstillfällena. Dessa uppgifter godkänns alltså automatiskt om du deltar i laborationstillfället.**

#### 1. *Endimensionell räkka 1*

**Att göra:** Skriv ett program som läser in sju heltal och lagrar dessa i en räkka, och därefter skriver ut alla tal som finns i räkkan. Använd dig av två loopar; en för inmatning och en för utskrift. Utskriften kan t.ex. se ut enligt nedanstående:

**Mata in sju värden: 4 29 12 9 3 1 2**  
**Inmatade värden: 4 29 12 9 3 1 2**

#### 2. *Endimensionell räkka 2*

**Att göra:** Modifiera ditt program från uppgift 1 så att det endast skriver ut värden som ligger mellan två gränsvärden. Utskriften kan t.ex. se ut enligt nedanstående:

**Mata in sju värden: 4 29 12 9 3 1 2**  
**Mata in undre gräns: 3**  
**Mata in övre gräns: 10**  
**Dessa värden ligger inom intervallet: 4 9 3**

**Mata in sju värden: 4 29 12 9 3 1 2**  
**Mata in undre gräns: 0**  
**Mata in övre gräns: 100**  
**Dessa värden ligger inom intervallet: 4 29 12 9 3 1 2**

**Mata in sju värden: 4 29 12 9 3 1 2**  
**Mata in undre gräns: 100**  
**Mata in övre gräns: 200**  
**Dessa värden ligger inom intervallet: Inga värden hittades!**

### 3. *Flerdimensionella räckor 1*

**Att göra:** Skriv ett program som skapar en tvådimensionell räckor med 5 rader och 10 kolumner. Initialisera räckan så att varje element har ett värde som är lika med  $\text{radindex} * \text{kolumnindex}$ , och skriv slutligen ut innehållet i räckan.

```
0 0 0 0 0 0 0 0 0 0    <= printf("%3d", ...)
0 1 2 3 4 5 6 7 8 9
0 2 4 6 8 10 12 14 16 18
0 3 6 9 12 15 18 21 24 27
0 4 8 12 16 20 24 28 32 36
```

### 4. (\*) *Lookup table*

**Att göra:** Modifiera barnbidragsräknaren från inlämningsuppgift 3 så att beräkningslogiken hämtar de olika barnbidragsbeloppen (för 1, 2, 3, 4, 5+ barn) från en “lookup tabell” i form av en räckor.

Skriv programmet så att reglerna kan modifieras endast genom att ändra på räckans innehåll och storlek, utan att beräkningslogiken behöver förändras.

(Frivillig extra övning: Gör samma sak för att ange tillgängliga valörer för ‘växelberäkningen’ från inlämningsuppgift 2)

## 5. (\*) *Flerdimensionella räckor 2*

**Att göra:** Skriv ett program som skapar en tvådimensionell räckor med de dimensioner som användaren anger. Räckan skall innehålla **char**-värden. Varje element i räckan skall initialiseras till ett bindestreck (-). Låt därefter programmet köra i en oändlig loop som gör följande:

- 1) Låter användaren mata in en koordinat (rad och kolumn)
- 2) Lagrar en stjärna (\*) på motsvarande position i räckan
- 3) Skriver ut innehållet i hela räckan

Programmet behöver inte innehålla någon kontroll av indatan, du kan utgå från att användaren alltid matar in giltiga värden.

Gör programmet användarvänligt genom att låta användaren mata in värden mellan 1 och maxhöjd/maxbredd. Med andra ord skall användaren inte behöva känna till att räckans första element egentligen har koordinaterna (0,0) och inte (1,1). Det kan dock vara en bra idé att lägga till denna finess först efter att grundfunktionaliteten fungerar.

En programkörning kan se ut som följer:

**Bredd? 5**

**Höjd? 3**

**Ge nästa koordinat: 1 1**      <= Första raden, första kolumnen

\* \_ \_ \_ \_

\_ \_ \_ \_ \_

\_ \_ \_ \_ \_

**Ge nästa koordinat: 1 2**      <= Första raden, andra kolumnen

\*\* \_ \_ \_

\_ \_ \_ \_ \_

\_ \_ \_ \_ \_

**Ge nästa koordinat: 2 5**      <= Andra raden, femte kolumnen

\*\* \_ \_ \_

\_ \_ \_ \_ \*

\_ \_ \_ \_ \_

**Ge nästa koordinat: 2 6**      <= Oops, för högt kolumnnummer!

\*\* \_ \_ \*

\_ \_ \_ \_ \*

\* \_ \_ \_ \_

<= ???

**Att redovisa:** Ditt program **och** din teori för varför den sista inmatningen i exempelpörningen ger detta oväntade(?) resultat.