

## HI1024 Programmering, grundkurs TEN2

2015-01-08

KTH STH Haninge 13.15-18.00

### Tillåtna hjälpmedel:

En A4 handskriven på ena sidan med egna anteckningar

Kursboken "C PROGRAMMING A Modern Approach" K. N. King – helt utan anteckningar

Tomma papper och penna.

**Logga in med tentamenskontot ni får av skrivvakten. Det kommer att ta tid att logga in – ha tålamod!** Logga inte ut förrän ni är klara med hela tentan. Svaren på uppgifterna ska vara program bestående av **en** fil som kallas uppg1.c, uppg2.c, uppg3.c och uppg4.c. **Dessa ska sparas direkt under H:**. Även om ni inte löser hela uppgiften så kan ni få delpoäng men ni måste se till att koden kompilerar och kan köras. Både output och koden bedöms och det är viktigt att denna är välskriven och uppfyller uppgiftens instruktioner. Välj bra variabelnamn och struktur men lägg inte så stor vikt vid kommentarer. **Kom ihåg ert tentamenskonto.** Resultaten kommer att presenteras kopplade till dessa.

Lycka till

Betygsgränser (Max 18p): 8p - Fx, 9p - E, 11p - D, 12p - C, 14p - B, 16p - A

Examinator: Nicklas Brandefelt

C99

Vill du kunna använda C99 i Code::Blocks måste du göra följande:

Settings – Compiler – Other options

Lägg till: -std=c99 i fönstret och tryck OK

1. **Fakultet (4p)**

Fakulteten av ett tal beräknas enligt  $n! = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 1$ . Ex:  $4! = 4 \cdot 3 \cdot 2 \cdot 1 = 24$

Skriv ett program som beräknar fakulteten av ett icke negativt heltal. Fakulteten av noll ska bli ett och om användaren anger ett negativt tal skall programmet svara med "Error". En körning av ditt program bör kunna se ut enligt nedan:

Ange tal:4

Fakulteten blir 24

## 2. Cirklar som struct (4p)

I den här uppgiften ska vi representera cirklar med en struct. En cirkel har en viss position i ett koordinatsystem och en radie. Positionen är platsen för cirkelns mittpunkt.

```
struct cirkel{
    int x,y;
    int radie;
};
typedef struct cirkel Cirkel;
```

a) Skriv en funktion skrivUt(...) som tar en Cirkel som in-parameter och skriver ut den till standard ut. Observera att denna funktion inte ska läsa in något från standard in. Anropa din funktion med en cirkel från main. Skapa cirkeln enligt exemplet nedan.

Ex: En cirkel med position (3,2) och radien 5 skrivs ut som:

```
Cirkel vid (3,2) med radie 5
```

b) Skriv en funktion flytta(...) som tar en pekare till en cirkel och två heltal (dx och dy) som in-parametrar och som flyttar cirkeln (den cirkel som pekaren pekar på) dx enheter i x-led och dy enheter i y-led. Observera att detta är en hjälpfunktion: den ska inte läsa in något från standard in eller skriva ut något till standard out. Anropa din funktion med cirkeln från a) och flytta den ett steg i x-led och två steg i y-led innan du skriver ut den.

c) Använd Cirkel-structen och dina funktioner ovan och skriv ett program som slumpar fram tio cirklar och lagrar dessa i en array. x och y ska slumpas så att värdena 0,1,2,...10 alla är lika sannolika. Radien ska slumpas så att värdena 1, 2, och 3 är lika sannolika. Programmet skall sedan flytta alla cirklar lika många steg i positiv x-led som värdet på respektive cirkels radie. En cirkel med radien 2 och position (2,4) skall alltså flyttas till (4,4). Avsluta med att skriva ut alla cirklar. Förutom att du ska använda funktionerna ovan skall du också använda arrayer och loopar på ett bra sätt för att få full poäng på uppgiften.

### 3. Glosor (6p)

Denna uppgift går ut på att skriva ett program som låter användaren öva på glosor. För att förenkla behöver programmet inte innehålla någon filhantering. Programmet skall först presentera en meny där valen skall vara:

1. Mata in nya glosor
2. Öva från svenska till engelska
3. Avsluta

De olika valen:

1. Här får användaren mata in först den svenska glosan och sedan den engelska ända tills användaren istället för att mata in en svensk glosa slår \* <enter>. Då skall programmet återgå till menyn. Väljer man detta val flera gånger skall glosorna läggas till den befintliga gloslistan.

2. Här slumpas svenska glosor fram och användaren får efter varje försök skriva den engelska glosan. För full pott skall varje glosa slumpas precis en gång. När programmet slumpat lika många glosor som det finns så presenterar programmet antal procent rätt och återgår till huvudmenyn. Finns inga glosor återgår programmet direkt till huvudmenyn.

3. Programmet avslutar.

Programmet skall använda funktioner och arrayer på ett bra sätt. Programmet får inte använda globala variabler. Programmet behöver inte hantera att användaren matar in fler än hundra glosor eller att glosorna är fler än 30 tecken långa.

#### 4. Labyrint (4p)

I den här uppgiften skall du skriva en funktion som hittar ut ur en labyrint. Labyrinten representeras av en 11\*11 char-matris där '\*' representerar en vägg och mellanslag: ' ' representerar en gång och 's' representerar startpositionen. Man kan från varje position i labyrinten gå upp, ned, höger eller vänster om det inte är en vägg i vägen (ej diagonalt). Funktionen skall returnera 1 om det finns en väg ut från 's' och 0 om det inte gör det. I mappen W:\HI1024 finns ett skal med två exempel-labyrinter definierade (den första saknar lösning och den andra har en lösning). Din funktion skall naturligtvis klara en godtycklig labyrint.