Örebro universitet
Institutionen för teknik
Thomas Padron-McCarthy (Thomas.Padron-McCarthy@tech.oru.se)

Tentamen i
Programmering C

för D1 m fl, även distanskursen

måndag 27 augusti 2007 kl 14:00 - 19:00 i L001

Gäller som tentamen för:
PRG045 Programmering C, provkod 0110 (och 0100, och 0101)
TDP118 Programmering C, distans, provkod 0110 (och 0100)
TDD121 Tillämpad datavetenskap A, provkod 0700 (och 0400)

Hjälpmedel:	Inga hjälpmedel.
Poängkrav:	Maximal poäng är 40. För godkänt betyg (3 respektive G) krävs 20 poäng.
Resultat och lösningar:	Meddelas på kursens hemsida, http://www.aass.oru.se/~tpy/c/2006-2007-p2/, senast måndag 10 september 2007.
Visning och frågestund:	Tisdag 11 september 2007 kl 12:00-12:30 i mitt rum (T2220). Efter visningen kan tentorna hämtas på expeditionen. Man kan också få sin rättade tenta hemskickad.
Examinator och jourhavande:	Thomas Padron-McCarthy, telefon 070-73 47 013.

- Skriv tydligt och klart. Lösningar som inte går att läsa kan naturligtvis inte ge några poäng. Oklara och tvetydiga formuleringar kommer att misstolkas.
- Skriv den personliga tentamenskoden på varje inlämnat blad. Skriv *inte* namn eller personnummer på bladen.
- Skriv bara på en sida av papperet. Använd inte röd skrift.
- Antaganden utöver de som står i uppgifterna måste anges.
- Skriv gärna förklaringar om hur du tänkt. Även ett svar som är fel kan ge poäng, om det finns med en förklaring som visar att huvudtankarna var rätt.

LYCKA TILL!

Prioritet och associativitet hos operatorerna i C

De viktigaste operatorerna:

Prioritet	Kategori	Operator	Associativitet
Högsta	Unära postfixoperatorer	(), [], ->, ., ++, --	vänster
	Unära prefixoperatorer	!, ++, --, +, -, *, &, sizeof, (*typ*)	höger
	Multiplikation mm	*, /, %	vänster
	Addition mm	+, -	vänster
	Jämförelser	<, <=, >=, >	vänster
	Likhetsjämförelser	==, !=	vänster
	Logiskt OCH	&&	vänster
	Logiskt ELLER	||	vänster
Lägsta	Tilldelning	=, +=, -=, *=, /=, %=	höger

Uppgift 1 (1 p)

Skriv formeln nedan som en tilldelningssats i C.

$$y = 1 - x + \frac{\sqrt{1-x}}{1-x}$$

Uppgift 2 (1 p)

Vilka värden har följande uttryck?

a) 1 - (2 + 3) * 4

b) 1 * (2 % 3) % 4

Uppgift 3 (1 p)

i och **k** är heltalsvariabler. Ange värdet på i och k då följande kod har körts.

```
i = 3;
k = 0;
while (i > 10) {
    if (i < 7)
        i = i * 2;
    else
        i = i + 2;
    k++;
}
```

Uppgift 4 (1 p)

Sveriges främste samlare av jordglobar vill lagra data om sina jordglobar med hjälp av ett C-program. En jordglob har de här egenskaperna:

- Jordglobens inköpsdatum, som lagras som en textsträng, exempelvis **2007-08-27**.
- Staden där jordgloben köptes. Namn på städer kan vara trettio bokstäver långa.
- Jordglobens diameter i centimeter. Den anges med ett flyttal.
- En bedömning av hur snygg jordgloben är, på en skala från 1 till 5. Den lagras som ett heltal.

Vi behöver en posttyp (med C-terminologi en **struct**) som kan användas för att lagra data om en jordglob. Definiera posttypen. Den ska heta **struct Jordglob**.

Uppgift 5 (1 p)

Definiera en variabel av typen **struct Jordglob** och initiera den med data om en jordglob som är köpt i **Kumla 2007-08-27**, som är **35** cm i diameter, och som fick snygghetsbetyget **4**.

Uppgift 6 (2 p)

Skriv en funktion som heter **visa_jordglob**, som skriver ut en jordglob (som i uppgift 4) på skärmen. Funktionen tar jordglobsposten som parameter, och skriver ut den, med lämpliga ledtexter så man ser vad som är vad.

Så här ser funktionshuvudet ut:

```
void visa_jordglob(struct Jordglob jordgloben)
```

Uppgift 7 (2 p)

Skriv en funktion som heter **las_jordglob**, och som läser in data om en jordglob. Funktionen ska skriva ut lämpliga ledtexter på standardutmatningen, och läsa in data från standardinmatningen (som normalt är kopplad till tangentbordet).

Du får själv välja om du vill att funktionshuvudet ska se ut så här:

```
struct Jordglob las_jordglob()
```

eller så här:

```
void las_jordglob(struct Jordglob *p)
```

Uppgift 8 (2 p)

Skriv en funktion som heter **minst**, som tar data om två jordglobar som argument, i form av två stycken **struct Jordglob**. och som returnerar den av dem som har minst diameter. Om båda jordglobarna är lika stora, spelar det ingen roll vilken som returneras.

Uppgift 9 (2 p)

Skriv en main-funktion som har två lokala variabler av typen **struct Jordglob**, och som läser in data om två jordglob till dessa variabler med hjälp av funktionen **las_jordglob**. Avslutningsvis ska den använda funktionen **visa_jordglob** för att skriva ut data om den minsta av de två jordglobarna. Använd funktionen **minst** för att få fram den minsta.

I den här och alla andra uppgifter på tentan gäller: Om du ska anropa en funktion från en tidigare uppgift, behöver du inte skriva koden för den funktionen på nytt. Du får också anropa funktionen även om du inte gjort uppgiften där man skulle skriva den.

Uppgift 10 (2 p)

Skriv en funktion som heter **volym**, som tar data en jordglob som argument, och som returnerar volymen på jordgloben, mätt i kubikcentimeter.

Formeln för volymen av ett klot med radien **r**:

$$4\pi r^3$$

Uppgift 11 (5 p)

Läraren ska måla sitt hus, och behöver räkna ut hur mycket färg som går åt. Skriv ett C-program som hjälper till med det.

Först ska programmet fråga efter hur många kvadratmeter som en liter färg räcker till. Sen ska det upprepat fråga efter höjd och bredd på husets väggar, tills användaren matar in en vägg med höjden noll. Avslutningsvis ska programmet skriva ut hur många liter färg som behövs.

Exempel på hur dialogen kan se ut:

```
Ange kvadratmeter per liter färg: 3.5
Ange väggens höjd: 3
Ange väggens bredd: 10
Ange väggens höjd: 1
Ange väggens bredd: 5
Ange väggens höjd: 0
Det behövs 10.00 liter färg.
```

Uppgift 12 (5 p)

Skriv ett C-program som först frågar efter hur många jordglobar som ska matas in, och sen låter användaren mata in det antalet jordglobar, med hjälp av funktionen **las_jordglob**. Jordglobarna ska lagras i en array. Programmet ska kunna hantera högst 100 stycken jordglobar.

När alla jordglobarna är inlästa, ska programmet skriva ut hur många jordglobar som fått vart och ett av betygen 1-5.

Uppgift 13 (5 p)

Skriv ett C-program som läser in data om jordglobar från användaren, med hjälp av funktionen **las_jordglob**, och sparar dem på en fil. Välj själv om det ska vara en text- eller binärfil. Tala om vilket du valde.

Välj själv hur användaren ska avsluta inmatningen, dvs hur man anger att man inte vill mata in fler jordglobar, och beskriv vilket sätt du valde.

Glöm inte att beskriva hur inmatningen avslutas, och om du valde en textfil eller en binärfil!

Uppgift 14 (5 p)

Skriv ett C-program som läser filen från uppgift 13, och skriver ut följande uppgifter:

- Jordglobarnas genomsnittliga volym
- Var den största jordgloben är inköpt

Uppgift 15 (5 p)

Vi ska provköra programmet i uppgift 14, som läser en fil, men orkar inte mata in en massa jordglobar för hand. Därför gör vi en ny version av programmet i uppgift 13, som ska skapa filen. I stället för att användaren matar in jordglobar, ska programmet slumpa fram dem.

Skriv alltså ett C-program som slumpar fram data om mellan 10 och 10000 jordglobar, och sparar dem på en fil. Alla värden ska vara slumpmässiga men rimliga.

Ledtråd: Funktionen **rand** ger ett slumpmässigt heltal, och funktionen **srand**

initierar slumpsekvensen.