

Omtentamen - Programmering

DVA117

Akademien för innovation, design och teknik

Onsdag 2014-01-15

Skrivtid: 14.10 – 17.30
Hjälpmedel: Inga
Lärare: Caroline Uppsäll, 021-101456
Robert Suurna, 021-151790

Preliminära betygsgränser

Betyg 3: 25p
Betyg 4: 34p
Betyg 5: 42p
Max: 46p

Bonuspoäng

Inga bonuspoäng kan tas med till omtentamen.

Allmänt

- All kod skall skrivas i standard ANSI C.
- Påbörja varje ny uppgift på nytt blad och skriv bara på ena sidan av pappret. Deluppgifter (a, b, c,...) kan skrivas på samma blad
- Referera inte mellan olika svar.
- Om du är osäker på vad som avses i någon fråga, skriv då vad du gör för antagande.
- *Oläsliga/oförståeliga/ostrukturerade svar rättas inte.*
- Kommentera din kod!
- Tips: Läs igenom hela tentan innan du börjar skriva för att veta hur du ska disponera din tid.

Lycka till!

Prioritet och associativitet hos operatorerna i C

Högsta prioritet högst upp i tabellen, lägsta prioritet längst ner i tabellen. Grupperade operatorer har samma prioritet.

Table A.5 Summary of C Operators (Programming in C, Stephen G. Kochan)

Operator	Description	Associativity
()	Function call	Left to right
[]	Array element reference	
->	Pointer to structure member reference	
.	Structure member reference	
-	Unary minus	Right to left
+	Unary plus	
++	Increment	
--	Decrement	
!	Logical negation	
~	Ones complement	
*	Pointer reference (indirection)	
&	Address	
sizeof	Size of an object	
(type)	Type cast (conversion)	
*	Multiplication	Left to right
/	Division	
%	Modulus	
+	Addition	Left to right
-	Subtraction	
<<	Left shift	Left to right
>>	Right shift	
<	Less than	Left to right
<=	Less than or equal to	
>	Greater than	
>=	Greater than or equal to	
==	Equality	Left to right
!=	Inequality	
&	Bitwise AND	Left to right
^	Bitwise XOR	Left to right
	Bitwise OR	Left to right
&&	Logical AND	Left to right
	Logical OR	Left to right
?:	Conditional	Right to left
=	Assignment operators	Right to left
*= /= %=		
+= -= &=		
^= =		
<<= >>=	Comma operator	Right to left
,		

Uppgift 1 [7p]

- a) Vad har preprocessorn för uppgift när ett C-program kompileras?
- b) Varför deklarerar vi variabler i C, dvs. varför ger vi dem en viss typ (ange minst två anledningar)?
- c) Beskriv kortfattat begreppet *makro* och hur det används i C-kod. Ta även med ett exempel på hur man skriver ett makro.
- d) Minnesmodellen för ett C-program innehåller en del som vi kallar för *heap*. Vad används den till?
- e) Hur fungerar variabler som är deklarerad med lagringsklassen `static`?

Uppgift 2[4p]

Vilka värden har följande uttryck i C (vad evakuerar varje uttryck till)?

För samtliga uttryck gäller: `int x=40, y=2, z=4, a=1, b=6;`

- a) `z > 8 || x == y * 20`
- b) `(a != x || b != y) && !(b < x && a == 1)`
- c) `b >= a * 5 && y == ((x / 4) - 8) && !a`
- d) `x - a > b || z + a < b + x || z <= y`

Uppgift 3 [3p]

Antag följande rader kod:

```
char str[38]="Kan jag";  
strcat(str, " programmera C?\n");
```

- a) Vilket specialtecken bör alla strängar avslutas med för att fungera ihop med *str*-funktioner?
- b) Vilken längd har strängen, dvs. vad returnerar `strlen(str)`?
- c) Vad skrivs ut av satsen: `printf("%c", str[8]);`
- d) Hur kan man "tömma" (reset) strängen `str`?
- e) Hur många bitar (bits) används i minnet för variabeln `str` när båda raderna körts?

Uppgift 4 [10p]

Er lärare behöver ett program som kan hålla ordning på studenternas resultat i en kurs. För varje student i kursen ska man kunna spara namn (förnamn och efternamn), personnummer samt betyg på momenten LAB (U eller G) och TEN (U, 3, 4 eller 5) samt slutbetyget i kursen (3, 4 eller 5).

- a) Skriv koden för en lämplig typ (struct) som kan hålla informationen om en student i klassen enligt beskrivningen ovan. (2p)
- b) Skriv koden för att skapa ett fält (array) för kursen DVA105 som kan hålla 65 studenter genom att använda funktionen `calloc()`. Felhantering för allokering av dynamiskt minne är ett krav. (2p)
- c) Skapa en funktion som sätter betyget på momentet TEN. Funktionen ska som inparametrar minst ta studentens personnummer samt betyget. Returvärde från funktionen ska vara 1 om operationen lyckades och 0 om den inte lyckades. Ni kan här anta att arrayen från b) är fylld med data. (4p)
- d) Skriv en kodrad som tilldelar betyget 'G' på momentet LAB till studenten på den sista posten. Använd piloperatorn. (2p)

Uppgift 5 [3p]

Skriv om switch-satsen i koden nedan till en *if-else-if*-sats.

```
int main(void)
{
    int val = 0;

    puts("MENY");
    puts("-----");
    puts("1 - Lägg till post");
    puts("2 - Ta bort post");
    puts("3 - Sök efter post");
    puts("4 - Skriv ut samtliga poster");
    puts("5 - Avsluta programmet");

    scanf("%d", &val);

    switch(val){
        case 1: laggtill();
                break;
        case 2: taBort();
                break;
        case 3: sok();
                break;
        case 4: skrivUt();
                break;
        case 5: avsluta();
                break;
        default: puts("Felaktigt val");
    }

    return 0;
}
```

Skriv om
denna del

Uppgift 6 [10p]

Delfråga a till b nedan handlar om följande programkod.

```
#include <stdio.h>
#include <stdlib.h>

int main(void)
{
    float summa = 0, tal = 0;
    int antal = 0, i = 0, avsluta = 0;
    FILE fp = NULL;
    char filNamn[100] = {};

    printf("Ange namn på fil att läsa: ");
    scanf("%s", filNamn);

    if((fp = fopen(filNamn, "r")) != NULL)
    {
        printf("Ange antal tal: ");
        scanf("%d", &antal);
        i = antal;
        while(i > 0)
        {
            if(fscanf(fp, "%f", &tal) == EOF)
                summa += tal;
            i--;
        }
        printf("Summan: %.4f", summa);
    }
    else
    {
        printf("Filen %s gick inte att öppna, programmet avslutas", filNamn);
        avsluta = 1;
    }

    if(avsluta != 1)
    {
        printf("\nAnge fil att skriva resultat till: ");
        scanf("%s", filNamn);

        if((fp = fopen(filNamn, "w")) != NULL)
            printf(fp, "Summan av de %d första talen är %.f", antal, summa);
        else
        {
            printf("Filen %s gick inte att öppna, programmet avslutas", filNamn);
            avsluta = 1;
        }
        fclose(fp);
    }

    printf("Programmet avslutas");
    return 0;
}
```

- I koden finns ett antal fel (tankefel och fel som av en kompilator ger error). Det kan också vara någonting som saknas. Hitta felen, skriv ner hela den felaktiga raden och visa hur den ska rättas till (ändras).
Påståenden om fel som i själva verket är korrekt kod ger poängavdrag (lägsta poäng är 0).
- Beskriv vad programmet utför. dvs. vad matas in och vad blir resultatet (inte rad för rad).
- Ge ett exempel från koden på en variabeldeklaration om det finns.
- Ge ett exempel från koden på en variabelinitiering om det finns.
- Ge ett exempel från koden på en definition om det finns.
- Vad är skillnaden på en textfil och en binärfil? När vill man använda binärfil och när vill man använda textfil?

Uppgift 7 [4p]

Skriv ett C-program som skriver ut multiplikationstabeller för tabellerna 1 till 5. Varje tabell ska ha intervall 0-10. I din lösning ska du använda *nästlade loopar*. En av looparna ska vara av typen *for-loop* och en av typen *do-while-loop*.

Mellan varje tabell ska det skrivas ut en linje, men det ska inte skrivas ut en linje högst upp och inte en längst ner.

Skriv ut på formen :

```
1*0= 0
1*1= 1
1*2= 2
1*3= 3
1*4= 0
1*5= 1
1*6= 2
1*7= 3
1*8= 3
1*9= 3
1*10= 3
-----
```

```
2*0= 0
2*1= 2
2*3= 4
Osv...
```

Uppgift 8 [5p]

Nedan finner du ett huvudprogram. Din uppgift är att skriva en funktion `maxfunc`. Ett anrop till funktionen ska göras i huvudprogrammet på markerad plats. Du ska alltså skriva själva funktionen samt anropet. Funktionen måste fungera med resterande huvudprogram. Ändringar till huvudprogrammet får inte göras.

Funktionen

Skriv både metoden och anropet.

```
#define SIZE 5

int main(void)
{
    int arr[SIZE];
    int stor = 0, i;
    float medel = 0;

    for(i = 0; i < SIZE; i++)
    {
        printf("Ange tal: ");
        scanf("%d", &arr[i]);
    }
    //METODANROP TILL maxfunc
    printf("Största talet: %d\nMedeltal: %.2f", stor, medel);
}
```