



Tentamen - Programmering

DVA117

Akademien för innovation, design och teknik

Måndag 2018-03-19

An English translation of the entire exam follows after the questions in Swedish

Skrivtid: 14.10 – 19.30

Hjälpmedel: Valfritt icke-elektroniskt material

Lärare: Caroline Uppsäll, 0704616110
(kan nås på telefon om du frågar tentavakten)

Antal uppgifter: 11

Preliminära betygsgränser

Betyg 3: 14p

Betyg 4: 21p

Betyg 5: 25p

Max: 28p

Allmänt

- All kod skall skrivas i standard ANSI C.
- Skriv tydligt vilken uppgift/deluppgift ditt svar anser.
- Skriv endast på bladets ena sida.
- 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!
- Det är inte tillåtet att använda goto-satser och globala variabler
- Tips: Läs igenom hela tentamen innan du börjar skriva för att veta hur du ska disponera din tid.

Lycka till!

/Caroline

Uppgift 1 [1p]

Vad är skillnaden på följande två variabler (x1 och x2)? Förklara.

```
int x1;  
float x2;
```

Poäng/delpoäng på uppgiften beror på hur utförligt svaret är.

Uppgift 2 [1p]

Vilken är skillnaden på att använda if-else if-else och switch-case för att utföra selektion (val) i ett program – mer än att deras syntax ser olika ut.

Uppgift 3 [1p]

```
int a = 10, b = 25, c = 15, d = 5;  
  
((b%2 == 0) || (a>=d-c)) && ((d<c%10) || ((1==1) && !(a!=b+c+d)))
```

Vad evaluerar det logiska uttrycket till (sant eller falskt)?

Uppgift 4 [4p]

Nedan används en while-loop för att åstadkomma ett loop-beteende. Skriv om loop-beteendet (koden innan för `*****LOOP*****`) på två olika sätt. Dels med hjälp av for-konstruktionen och dels med hjälp av do-while konstruktionen. Du kan anta att användaren inte skriver in negativa värden till start och decrementStep. Du kan också anta att inputen till decrementStep inte är 0, däremot kan inputen till start vara 0.

```
#include <stdio.h>  
int main(void)  
{  
    int start, decrementStep, i;  
    scanf("%d", &start);  
    scanf("%d", &decrementStep);  
  
    *****LOOP*****  
    i = start;  
  
    while(i > 0)  
    {  
        //do something, tex. printf("%d ", i);  
        i = i - decrementStep;  
    }  
    *****LOOP*****  
}
```

Uppgift 5 [2p]

Antag att du har en binärfil som du öppnar med ett program som är till för att läsa textfiler (t.ex. notepad eller TextEdit). Informationen i filen kommer då upplevas som korrupt. Förklara varför (använd inte mer än 10 meningar).

Uppgift 6 [1p]

Förklara med några få meningar (max 5) vad en algoritm är.

Uppgift 7 [4p]

Skapa funktionen `fillArray`.

Arrayen ska fyllas med slumpade heltal mellan `minLimit` och `maxLimit` (inklusive dessa).

Om `minLimit` är 5 och `maxLimit` är 10 ska alltså talen 5, 6, 7, 8, 9 och 10 kunna slumpas.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define SIZE 10

int main(void)
{
    int arr[SIZE];
    int minLimit, maxLimit;
    printf("Enter limits for randomized integers: ");
    scanf("%d %d", &minLimit, &maxLimit);

    fillArray(arr, minLimit, maxLimit);

    return 0;
}
```

Uppgift 8 [2p]

Förklara (med ord och potentiellt bild) hur man skulle kunna göra för att ta reda på hur många meningar en sträng består av. En mening kan avslutas med punkt (.), frågetecken (?) eller utropstecken (!).

Uppgift 9 [3p]

Skriv en funktion som byter plats på två heltal. Visa också hur funktionsanropet ser ut i nedanstående program. Funktionens returtyp ska vara void.

Om input till nedanstående program är: 10 och 20 så ska utskrifterna bli:

Before funk: 10 and 20

After funk: 20 and 10

```
#include <stdio.h>

int main(void)
{
    int number1, number2;
    printf("Enter two numbers: ");
    scanf("%d %d", &number1, &number2);

    printf("Before funk: %d and %d", number1, number2);
    //funktionsanrop
    printf("After funk: %d and %d", number1, number2);

    return 0;
}
```

Uppgift 10 [3p]

När vi bygger vårt program sker tre saker innan vi får ut en exekverbar fil, en av dem är kompilering (som faktiskt består av två saker). Förklara vilka dessa tre saker är och vad de gör.

Uppgift 11 [6p]

I nedanstående program finns totalt 12 fel, både syntax, semantik och logik. Vissa fel kan ligga på samma rad och vissa existerar på flera rader. Existerar felet på flera rader räknas det fortfarande som ett fel.

Ange för varje fel du hittar vilken/vilka rad/rader det finns på samt hur satsen ska se ut. 0,5p per korrekt hittat och rättat fel.

```
/*Programmet ska allokerar minne för 10 filmer och sedan fylla alla filmer
med information om titel och produktionsår*/

1      #include <stdio.h>
2      #include <stdlib.h>
3
4      struct movie
5      {
6          char title[STRSIZE];
7          int productionYear;
8      };
9
10     typedef struct
11     {
12         movie *list;
13         int numberOf;
14     }movieList;
15
16     int main(void)
17     {
18         movieList *myMovies;
19         myMovies->numberOf = 10;
20         myMovies = (movieList*)calloc(myMovies->numberOf, sizeof(movieList));
21
22         if(myMovies->list == NULL)
23         {
24             printf("Enter movie information\n-----\n");
25             addInformation(&myMovies);
26             free();
27         }
28         return 0;
29     }
30
31     void addInformation(movieList *myMovies)
32     {
33         for(i = 0; i <= myMovies->numberOf; i++)
34         {
35             printf("Title: ");
36             fgets(myMovies[i]->list.title, STRSIZE, stdin);
37             printf("ProductionYear: ");
38             scanf("%d", myMovies[i]->list.productionYear);
39         }
40     }
```

Exam - Programming

DVA117

School of Innovation, design and technology

Monday 2018-03-19

Writing time: 14.10 – 19.30

Aids: Any non-electronic material

Examiner: Caroline Uppsäll, 0704616110
(Can be reached by telephone if you ask the exam guard)

Number of questions: 11

Preliminary grading limits

Grade 3: 14p

Grade 4: 21p

Grade 5: 25p

Max: 28p

Generally

- All code should be written in standard ANSI C.
- Write clearly what task/sup-task your answers consider.
- Do only use one side of the paper.
- Do not refer between answers.
- If you are unsure of a meaning of a question, write down your assumption.
- *Unreadable/incomprehensible answers will not be marked.*
- Comment your code!
- It is not allowed to use goto-statements or global variables
- Hint: To know how to allocate your time, read through the entire exam before you start writing.

Good luck!

/Caroline

Question 1 [1p]

What is the difference of the two variables below (x1 and x2)? Explain.

```
int x1;  
float x2;
```

Point is given based on how detailed the answer is.

Question 2 [1p]

What is the difference of using if-else if-else and switch-case to create selection in a program – more than that the syntax looks different.

Question 3 [1p]

```
int a = 10, b = 25, c = 15, d = 5;  
  
((b%2 == 0) || (a>=d-c)) && ((d<c%10) || ((1==1) && !(a!=b+c+d)))
```

What is the result of the logical expression (true or false)?

Question 4 [4p]

In the small program below a while-loop is used to create a loop behaviour. Rewrite the loop behaviour (the code surrounded by `/******LOOP******/`) in two different ways. Once by using a for-loop and once by using a do-while-loop. You can assume that the user doesn't enter negative numbers to start and decrementStep and that the input to decrementStep isn't 0. 0 is a possible input to start.

```
#include <stdio.h>  
int main(void)  
{  
    int start, decrementStep, i;  
    scanf("%d", &start);  
    scanf("%d", &decrementStep);  
  
    /******LOOP*****/  
    i = start;  
  
    while(i > 0)  
    {  
        //do something, for example printf("%d ", i);  
        i = i - decrementStep;  
    }  
    /******LOOP*****/  
}
```

Question 5 [2p]

Assume that you have a binary file that you open with a program applicable for opening text files (for example notepad or TextEdit). The information in the file will be perceived as it is corrupted. Explain why (use no more than 10 sentences)

Question 6 [1p]

What is an algorithm? Explain using a maximum of 5 sentences.

Question 7 [4p]

Create the function `fillArray`.

The Array are to be filled with random integer numbers between `minLimit` and `maxLimit` (including these)

If `minLimit` is 5 and `maxLimit` is 10 then 5, 6, 7, 8, 9 and 10 are possible random numbers in the interval.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define SIZE 10

int main(void)
{
    int arr[SIZE];
    int minLimit, maxLimit;
    printf("Enter limits for randomized integers: ");
    scanf("%d %d", &minLimit, &maxLimit);

    fillArray(arr, minLimit, maxLimit);

    return 0;
}
```

Question 8 [2p]

Explain (with words and potentiellay pictures) how you would (in a c program) find out how many sentences a string contains of. A sentence can end with a dot (.), question mark (?) or exclamation mark (!)

Question 9 [3p]

Write a function that swaps the content of two integers. You also need to show what the function call looks like. The return type of the function should be void.

If the input to the program below is: 10 and 20, the output should be:

Before funk: 10 and 20

After funk: 20 and 10

```
#include <stdio.h>

int main(void)
{
    int number1, number2;
    printf("Enter two numbers: ");
    scanf("%d %d", &number1, &number2);

    printf("Before funk: %d and %d", number1, number2);
    //function call
    printf("After funk: %d and %d", number1, number2);

    return 0;
}
```

Question 10 [3p]

When a C program is building, three things happens before an executable file is created, one of them is the compiler (which actually consists of two things). Explain all three thing together with what they do (their task).

Question 11 [6p]

In the program below there is a total of 12 errors, both syntax, semantics and logic. Some of the errors can be on the same row and some of the errors exists on multiple rows. If an error exists on multiple rows it still counts as one error.

For each of the errors you find – write down the row/rowa and how it should look to be correct.

Each correct error found (row) and corrected gives 0.5p

*/*The program should allocate memory for 10 movies and then fill all the movies with information about title and production year */*

```
1      #include <stdio.h>
2      #include <stdlib.h>
3
4      struct movie
5      {
6          char title[STRSIZE];
7          int productionYear;
8      };
9
10     typedef struct
11     {
12         movie *list;
13         int numberOf;
14     }movieList;
15
16     int main(void)
17     {
18         movieList *myMovies;
19         myMovies->numberOf = 10;
20         myMovies = (movieList*)calloc(myMovies->numberOf, sizeof(movieList));
21
22         if(myMovies->list == NULL)
23         {
24             printf("Enter movie information\n-----\n");
25             addInformation(&myMovies);
26             free();
27         }
28         return 0;
29     }
30
31     void addInformation(movieList *myMovies)
32     {
33         for(i = 0; i <= myMovies->numberOf; i++)
34         {
35             printf("Title: ");
36             fgets(myMovies[i]->list.title, STRSIZE, stdin);
37             printf("ProductionYear: ");
38             scanf("%d", myMovies[i]->list.productionYear);
39         }
40     }
```