



Tentamen - Programmering

DVA117

Akademien för innovation, design och teknik

Torsdag 2015-08-20

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

Skrivtid: 08.10 – 11.30
Hjälpmedel: Inga
Lärare: Caroline Uppsäll, 021-101456
Robert Suurna, 021-151790

Preliminära betygsgränser

Betyg 3: 12p
Betyg 4: 17p
Betyg 5: 21p
Max: 23p

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 tentamen innan du börjar skriva för att veta hur du ska disponera din tid.

Lycka till!

 MÄLARDALENS HÖGSKOLA ESKILSTUNA VÄSTERÅS	Kod: Code	Kurskod: Course code	Bladnr: Page nr
Uppgift nr: Task nr		Kursnamn: Course title	

Använd den här tabellen för att svara på delfrågorna i **Uppgift 1**. För varje delfråga (A-G) ska du markera det svarsalternativ som du tror är rätt svar med ett kryss (X) i motsvarande ruta.

Uppgift 1	Svar 1)	Svar 2)	Svar 3)	Svar 4)
A)				
B)				
C)				
D)				
E)				
F)				
G)				

Observera! Glöm inte att bifoga den här sidan tillsammans med övriga svarsblad när du lämnar in tentamen.

Uppgift 1 [7p]

Varje delfråga är värd 1p. Ett av svarsalternativen för varje delfråga är rätt. Använd gärna den bifogade svarstabellen eller svara genom att ange bokstaven för delfrågan och den siffra som motsvarar det rätta svarsalternativet (Exempel: H2, K3, osv...).

A. Vad är sant för följande kod?

```
struct person{  
    char name[10];  
    int age;  
};
```

1. Det är en definition
2. Det är en deklaration och en initiering
3. Det är en definition och en initiering
4. Syntax Error

B. följande logiska uttryck om x=40, y=2, z=4, a=1, b=6?

```
res= x - a > b || z + a < b + x || z <= y;
```

1. res=0
2. res=1
3. res=5
4. Odefinierat

C. På ett 32 bitars system, vad är den mest sannolika storleken i *byte* för en integer (*int*) ?

1. 2
2. 4
3. 16
4. 32

D. Vilken av följande kan inte användas i en switch-case-sats?

1. Character
2. Integer
3. Float
4. Enum

E. Vilken operator används för att få minnesadressen till en variabel?

1. *
2. &
3. &&
4. ||

F. Vilken typ returnerar minnesallokeringsfunktionerna malloc() och calloc()

1. void *
2. Pekare av samma typ som typen av det allokerade minnet
3. void **
4. int *

G. Vilket påstående är rätt för följande program?

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

```
int main(){
    int i=10;
    int *j=&i;
    return 0;
}
```

1. j och i är pekare till integer
2. i är en pekare till en integer och sparar adressen för j
3. j är en pekare till en integer och sparar adressen för i
4. j är en pekare till en pekare till en integer och sparar adressen av i

Uppgift 2 [6p]

För följande program (efter frågorna)

- A. Beskriv vad funktionen *consonant* gör.
- B. Beskriv vad funktionen *func* gör.
- C. Beskriv vad funktionen *main* gör.
- D. Vad kommer skrivas ut på skärmen när man skriver in texten "examina"
- E. Vad kommer samtidigt skrivas till filen?

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

```
int consonant(char ch){
    switch (ch) {
        case 'a': return 0;
        case 'e': return 0;
        case 'i': return 0;
        case 'o': return 0;
        case 'u': return 0;
        case 'y': return 0;
        case ' ': return 0;
        default : return 1;
    }
}
```

```
int func(FILE *fp, char ch) {
    fprintf(fp, "%c",ch);

    return 0;
}
```

```
int main()
{
    FILE *fp;
    char in[30];
    int i;

    if((fp = fopen("result.txt","w")) == NULL){
        printf("File could not be opened\n");
        return 0;
    }
}
```

```

printf("Input: ");
gets(in);

for(i=0; i<30 && in[i]!='\0'; i++)
    if(consonant(*(in+i)))
        printf("%co%c ",in[i],*(in+i));
    else{
        printf("%c",in[i]);
        func(fp,in[i]);
    }

fclose(fp);
return 0;
}

```

Uppgift 3 [10p]

Följ specifikationerna i deluppgifterna nedan – en fungerande lösning som ej är uppbyggd enligt specifikationen ger poängavdrag.

- A. Skapa en datatyp `City` som håller information om en stad. Den information som ska finnas om staden är namn, antal invånare samt koordinater. För att lagra koordinaterna för staden så ska följande datatyp (`Coordinates`) användas.

```

typedef struct
{
    int longitude;
    int latitude;
}Coordinates;

```

- B. Skapa en dynamiskt allokerad array med plats för `SIZE` städer. Glöm inte felhantering. Du kan anta att `SIZE` är ett definierat makro.
- C. Skriv en funktion (`addCity`) vars uppgift är att fylla en stad med information (namn, antal invånare, longitud och latitud). Funktionen ska som enda parameter ta en pekare till den stad (det element i den dynamiska arrayen) som ska editeras. *Funktionen får inte ta hela den dynamiska arrayen som parameter.* Ingenting ska returneras.

Visa också hur koden som fyller hela den dynamiska arrayen (från uppgift b) med städer ser ut.

- D. Skriv en funktion (`largestCity`) vars uppgift är att hitta den stad i arrayen som har flest invånare. Funktionen ska som parameter ta en pekare till arrayen av städer samt returnera antalet invånare i den största staden.

Exam - Programming

DVA117

School of innovation, design and technology

Thursday 2015-08-20

Write time: 08.10 – 11.30

Aids: No

Teachers: Caroline Uppsäll, 021-101456
Robert Suurna, 021-151790

Preliminary grading limits

Grade 3: 12p

Grade 4: 17p

Grade 5: 21p

Max: 23p

Generally

- All code should be written in standard ANSI C.
- Write each question on a new paper and do only use one side of the paper. Sub questions (a, b, c...) can be written on the same 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!
- Hint: To know how to allocate your time, read through the entire exam before you start writing.

Good luck!

 MÄLARDALENS HÖGSKOLA ESKILSTUNA VÄSTERÅS	Kod: Code	Kurskod: Course code	Bladnr: Page nr
Uppgift nr: Task nr		Kursnamn: Course title	

Use this reply table to answer the sub-questions of Question 1. For each sub-question (A-G), select the answer that you think is the correct one and put a cross (X) in the corresponding box.

Question 1	Answer 1)	Answer 2)	Answer 3)	Answer 4)
A)				
B)				
C)				
D)				
E)				
F)				
G)				

Please note! Do not forget to attach this page along with other answer sheets when you hand in the exam.

Question 1 [7p]

Each sub-question is worth 1p. One of the answers for every sub-question is correct. Please use the attached reply table or answer by entering the letter of the sub-question and the number corresponding to the correct answer (Example: H2, K3, etc...).

A. What is true for the following code?

```
struct person{  
    char name[10];  
    int age;  
};
```

1. It is a definition
2. It is a declaration and an initiation
3. It is a definition and an initiation
4. Syntax Error

B. What will be the result of the following logical expressions if x=40, y=2, z=4, a=1, b=6?

```
res= x - a > b || z + a < b + x || z <= y;
```

1. res=0
2. res=1
3. res=5
4. Undefined

C. On a 32 bit system, what is the most likely size in bytes of an integer (`int`) ?

1. 2
2. 4
3. 16
4. 32

D. Which of the following cannot be checked in a switch-case statement?

1. Character
2. Integer
3. Float
4. Enum

E. What operator is used to get the address value for a variable?

1. *
2. &
3. &&
4. ||

H. What is the return type of the memory allocation functions `malloc()` or `calloc()`?

1. `void *`
2. Pekare av samma typ som typen av det allokerade minnet
3. `void **`
4. `int *`

F. Which of the statements is correct about the program?

G. `#include<stdio.h>`

```
int main(){
    int i=10;
    int *j=&i;
    return 0;
}
```

1. j and i are pointers to an integer
2. i is a pointer to an integer and stores address of j
3. j is a pointer to an integer and stores address of i
4. j is a pointer to a pointer to an integer and stores address of i

Question 2 [6p]

See the following program (after the questions)

- A. Describe the functionality of the function *constant*
- B. Describe the functionality of the function *func*.
- C. Describe what functionality of the function *main*
- D. What will be printed on the screen when you input the text "examina"
- E. What will simultaneously be written to the file?

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

int consonant(char ch){
    switch (ch) {
        case 'a': return 0;
        case 'e': return 0;
        case 'i': return 0;
        case 'o': return 0;
        case 'u': return 0;
        case 'y': return 0;
        case ' ': return 0;
        default : return 1;
    }
}

int func(FILE *fp, char ch) {
    fprintf(fp, "%c",ch);

    return 0;
}

int main()
{
    FILE *fp;
    char in[30];
    int i;

    if((fp = fopen("result.txt","w")) == NULL){
        printf("File could not be opened\n");
        return 0;
    }
}
```

```

printf("Input: ");
gets(in);

for(i=0; i<30 && in[i]!='\0'; i++)
    if(consonant(*(in+i)))
        printf("%co%c ",in[i],*(in+i));
    else{
        printf("%c",in[i]);
        func(fp,in[i]);
    }

fclose(fp);
return 0;
}

```

Question 3 [10p]

Follow the specifications of the assignments bellow - a workable solution that is not structured according to the given specification will result in point deduction.

- A. Create a data type `City` that holds information about a city. The required information is name, number of residents and coordinates. To store the coordinates for the city the following data type (`Coordinates`) must be used.

```

typedef struct
{
    int longitude;
    int latitude;
}Coordinates;

```

- B. Create a dynamically allocated array with room for `SIZE` cities. Don't forget error handling. You can assume that `SIZE` is a defined macro.
- C. Write a function (`addCity`) whose task is to fill a city with its information (name, number of residents, longitude and latitude). The function must take a pointer to the city to be edited (the element in the dynamic array) as its parameter, *no other parameters to the function is allowed, the function is not allowed to take the entire array as parameter*. Nothing should be returned from the function.

You should also write the code that fills (calls the function `addCity`) the entire dynamic array (from task b) with information about cities.

- D. Write a function (`largestCity`) whose task is to find the city in the dynamic array that has the highest number of residents. The function should take a pointer to the array of cities as its parameter and return the number of residents in the largest city.