



Antal blad /
Number of sheets

08 ✓

TENTAMEN / EXAMINATION

Anvisningar:

Skriv din anonymitetskod på varje blad.
Endast en uppgift får lösas på varje blad.
Var vänlig skriv tydligt!

Instructions:

Write your anonymous code on each sheet.
Answer only one question on each sheet.
Please write clearly!

Vänligen texta anonymitetskoden i textboxen enligt exempel nedan!
Please write the Anonymous Code clearly in the textbox like example below!

Bokstäver/Letters:

A-B-C-D-E-F-G-H-I-J-K-L-M-N-O
P-Q-R-S-T-U-V-W-X-Y-Z-Å-Ä-Ö

Siffror/Numbers:

0-1-2-3-4-5-6-7-8-9

Exempel:

A B C 1 7 0 - 0 1 7

ISGA04 Grundläggande programmering
Kurskod + Kurs / Course Code + Course:

Delkurs / Part course:

Anonymitetskod / Anonymous code =
Kurskod + kodnr / course code + code number
ISGA04 - 006 ✓

Tentamensdatum /
Examination date:
23/8-2016

Behandlade uppgifter / Solved problems

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
✓	✓	✓	✓	✓	✓									
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Ifylles av lärare / To be completed by the examiner

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
9	9	2	3	1	6									
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30

Poäng / Marks gained: 30

Betyg / Grade: V4

Max poäng / Total marks gained: 40

För Gk poäng / Marks gained to be passed: 20

Examin. lärare / Kursansvarig signatur / Signature of the examiner

Johan Tharberg
Namnförtydligande / Clarification of the signature



ISGA04 - ØØ6

1

1

a) return används av funktioner i C och är användbart vid anrop (av funktioner). En funktion kan till exempel returnera ett intressant värde (såsom en int). Tex: return värde; 1-

d) free() = används för att frigöra en redan tagen minnesplats för ett värde. 1
 för att tilldela minnesplats används malloc() 1

b) typedef = används för att typdefiniera en egen datatyp (som kan innehålla flera värden, som en struct). tex: 1

```
typedef struct postType
```

```
int värde;
```

```
char text[10];
```

```
} postType
```

c) Selektion används för bättre användarvänlighet och kan representeras på olika sätt. Tex:

```
if(condition){
    statement(s);
} else {
    statement(s);
}
```

} if-sats

```
switch(expression){
    case 1:(condition)
        statement(s);
        break;
    case 2:(condition)
        statement(s);
        break;
}
```

} switch-sats 1

e) rand() används för att slumpa tal. Tex

```
s1 = rand() % 3;
```

Slumpas tal mellan 1-3 och lagras i variabeln s1. ✓



f) && = används som jämförelseoperator vilket betyder att den kan tex användas i if-satser för att kolla om ett värde uppfyller flera krav.

 Uppgift nr /
 Question no:

1

 Poäng / Points
 awarded:

9

 Lärarens
 anteckning
 Examiner's remarks:

g) Binärsökning används för att hitta tal i en lista. Principen är att sökningen hela tiden delas halva vektorn för att kolla vad som är i mitten.

DEN KRÄVER DOCK EN REDAN SORTERAD VEKTOR!

ex:

```
int main() {
    int a[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int first = 0, middle, last = 9, soktal;
    printf("Ange söktal: ");
    scanf("%d", &soktal);
    while (last - first > 1) {
        middle = (first + last) / 2;
        if (soktal == a[middle]) {
            first = middle;
        } else {
            last = middle;
        }
    }
    if (soktal == a[middle]) {
        printf("Antal %d finns på plats %d", soktal, middle);
    } else {
        printf("Talet hittades inte!");
    }
}
```

2

h) strcmp() = används för att jämföra strängar
 (Parameter: str, str med sträng 2: tex: strcmp(str1, str2);

1

i) Används för att lägga input (från stdin) till en vald destination, läser in till vald destination.
 tex: gets(post.namn), destination

1

I headerfil egen.h:

#ifndef

#include <stdio.h>

#include <stdlib.h>

typedef struct postType{

char namn[40];

char bostort[40];

int fodelseAr;

} postType;

#endif

SLUT header-fil!Main fil:

#include <stdio.h>

#include <stdlib.h>

#include "egen.h"

int main(){

int i, j, slask;

postType post; vek+10];

FILE *fp;

 if((fp = fopen("IGLIS.DAT", "a+b")) = NULL){
 printf("Kunde inte hitta filen!");
 exit(0);
 }

}

fseek(fp, 0, SEEK_SET);

forts nästa sida →

← Antagande: Jag

uppfattar som

för- och -efternamn

ska finnas i samma

post

// antar 10 inkommer max?



```

i = 0;
fread(&post, sizeof(postTyp), 1, fp);
while(!feof(fp)){
    vek[i] = post;
    i++;
    fread(&post, sizeof(postTyp), 1, fp);
}
antal = fgetc(fp) / sizeof(postTyp);
fclose(fp);
for(i = 0; i < antal; i++) {
    for(j = antal - 1; j > i; j--) {
        if(vek[j-1].post.fodelseAr > vek[j].post.fodelseAr) {
            slask = vek[j-1];
            vek[j-1] = vek[j];
            vek[j] = slask;
        }
    }
}
for(i = 0; i < antal; i++) {
    printf("Äldsta igelkott: %d \n Ågare %s",
        vek[i].post.fodelseAr, vek[i].post.namn);
}
return 0;
}

```

slut MAIN

Antagande: Jag uppfattar att bostadsort INTE ska skrivas ut. Skulle den ha gått det bör det se ut så här:

```

printf("Äldsta igelkott: %d \n Ågare: %s \n Bostadsort: %s",
    vek[i].post.fodelseAr, vek[i].post.namn, vek[i].post.bostOrt);

```



a) Vektorer har precis som pekare en allokerad minnesplats i minnet. Platsen kan bestå av ett värde eller datasamling (array). 1-

 Uppgift nr /
 Question no:

3

 Poäng / Points
 awarded:

2

 Lärarens
 anteckning
 Examiner's remarks:

b) Statisk variabel är en variabel som kan nås av alla programdelar och funktioner som ligger i samma fil (oftast main). Den kan inte ändras sitt värde eftersom den är statisk. ex
 static int exempel;

Dynamisk variabel är en variabel som finns bara i en programdel (eller fil) och dess värde kan ändras. Används också som en temporär variabel att t.ex skicka in ett intressant värde till. ex
 int i;

Användbart i loopar/iterationer.
 En statisk kan alltså nå överallt och dynamisk är användbart temporärt!

c) fread(&post, sizeof(postType), 1, fp);

 Vart den ska
 skriva

 hur långt den
 ska skriva

 hur många
 gånger den
 ska skriva

 och vart finns
 det den ska skriva
 (till vilken fil)

Används för att skriva till fil. Returnerar false ifall fp inte finns. (sista parametern)
 Exempel finns ovan! 4
 Annars skriver den.



En länkad lista består av flera strukturer som innehåller poster som pekar på ett annat värde. Det måste vara MINST en post/variabel som pekar på ett annat värde ifall det är en länkad lista.

följande kod jämför en struct med en länkad lista:

struct

```
typedef struct postTyp {
    char exempel[10];
    int exempel2;
} postTyp;
```

länkad lista:

```
typedef struct postTyp {
    char exempel[10];
    int exempel2;
} struct postTyp *next;
```

Skilnaden: pekar på nästa struct.



Ange anonymitetskod / Write your anonymity code
(Vid icke anonym tentamen ange kurskod + namn + personnummer)
(For non-anonymous exams write the course code + name + civic registration number)

ISGA04 - 006

Löpande sidnr
Consecutive no:

7

Uppgift nr /
Question no:

5

Poäng / Points
awarded:

1

Lärarens
anteckning
Examiner's remarks:

Följande utskiften genereras (endast talen):

Utskift 1: 4

Utskift 2: 3

Utskift 3: 12

1

Utskift 4: 3

Utskift 5: 10

Utskiften är i den ordningen som programmet
exekveras så säklart.

(E3 nödvändigtvis efter vart alla poängs är!).



```

int minfunktion(int vekt[], int tal){
    int i = tal;
    vekt[i] = vekt[i];
    places = sizeof(vekt[i]);
    int j, slask;

```

```

    for(i=0; i < places; i++){
        for(j=places-1; j > i; j--){
            if(vekt[j-1] > vekt[j]){
                slask = vekt[j-1];
                vekt[j-1] = vekt[j];
                vekt[j] = slask;
            }

```

```

        }
        printf("%d", vekt[i]);
    }
}

```

```

for(i=0; i < places; i++){
    for(j=places-1; j > i; j--){
        if(vekt[j-1] < vekt[j]){
            slask = vekt[j-1];
            vekt[j-1] = vekt[j];
            vekt[j] = slask;
        }

```

```

    }
    printf("%d", vekt[i]);
}

```

```

} // slut min funktion!

```