



Antal blad /
Number of sheets

10 ✓

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:

Ø-1-2-3-4-5-6-7-8-9

Exempel:

A	B	C	1	7	Ø	-	Ø	1	7
---	---	---	---	---	---	---	---	---	---

DVGAØ8 Grundläggande programmering

Kurskod + Kurs / Course Code + Course:

Delkurs / Part course:

Anonymitetskod / Anonymous code =
Kurskod + kodnr / course code + code number

D	V	G	A	Ø	8	-	Ø	1	2
---	---	---	---	---	---	---	---	---	---

Tentamensdatum /
Examination date:

2017-01-18

Behandlade uppgifter / Solved problems

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

Poäng / Marks gained:

51,5
5

Betyg / Grade:

Examin. lärare / Kursansvarig signatur / Signature of the examiner

Max poäng / Total marks gained:

Namnförtydligande / Clarification of the signature

För Gk poäng / Marks gained to be passed:



a) Double är en datatyp som kan hålla värden av decimaltal. Ex. double tal = 10.2; ↑

b) Switch är en selektion-sats där användaren måste välja ett val för att programmet ska fortsätta.

Ex.

```
Switch(val)
{
```

```
    case 1: printf("Val 1 körs..."); break;
```

```
    case 2: printf("Val 2 körs..."); break; ↑
```

```
    default: printf("Ogiltlig inmatning");
```

```
}
```

c) Datorns processor kan bara läsa och förstå två tecken, ettor och nollor. När C-kod kompileras omvandlas koden till ettor och nollor, också kallat maskinkod. ↑

Ex. 0011 1010

d) En lokal variabel är en variabel som endast är känd i den funktion den blir deklarerad i. Om en programmerare försöker använda en lokal variabel utanför funktionen variabeln blev skapad i, så skulle programmet inte känna igen den. ↑

e) Return används när en funktion ska returnera någonting, det som ska returneras skrivs efter return

Ex. return a+a; ↑

Return kan även användas av en funktion som inte returnerar någonting, då skriver man endast return; ↑



När en funktion använder "return;" är det som att säga "gå tillbaka" till main.

f) En konstant är som en variabel med enda skillnaden att en konstant inte kan ändra värde efter den här blivit deklarerad. Namnet på konstanten ska enligt bra programmeringsvana bestå av stora bokstäver och konstanten brukar bli deklarerad innan main-funktionen. Ex. `#define TAL 10`

g) En funktion kan anropas från main med ett värde anrop. Det betyder att när funktionen anropas skickas det med specifika värden som funktionen ska behandla.

h) En funktion kan även anropas genom ett referensanrop. Istället för att skicka med datatypers värden så skickar man istället med deras adresser.



a) #include <stdio.h>

```
int main()
```

```
{
    int tal, large, antal = 0;
```

```
    printf("Ange ett tal: ");
```

```
    scanf("%d", &tal);
```

```
    large = tal;
```

```
    if (tal % 2 == 0) —
```

```
{
        // Det största talet är 0
        antal++;
    }
```

```
    while (tal != 0)
```

```
{
        printf("Ange ytterligare ett tal");
```

```
        scanf("%d", &tal);
```

```
        if (tal > large)
```

```
{
```

```
            large = tal;
```

```
}
```

```
        if (tal % 2 == 0) —
```

```
{
            antal
```

```
            antal++;
```

```
}
```

```
}
```

```
    printf("Det största talet är %d och
```

```
%d tal är positiva", large, antal);
```

```
    return 0;
```

```
}
```

3

b) `int tal = 0``while (tal < 5)``{``printf("Hej");``tal++;``}``tal = 0;``do``{``printf("Hej");``tal++;``} while (tal < 5);``for (tal = 0; tal < 5; tal++)``{``printf("Hej");``}`

2,5

15,5



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)

DVGA08-012

Löpande sidnr
Consecutive no:

5

a) Utskriften blir: 2.0000 4.0000

2

Uppgift nr /
Question no:
3

b) Utskriften blir: Sant, n är nu 1-

1

Poäng / Points
awarded:

c) Utskriften blir: 12

2

Lärarens
anteckning
Examiner's remarks:

d) Utskriften blir: 6 8 0 10

3

18



a) `#include <stdlib.h> #include <time.h>`
`void slumpa()`
`{`
`int vektor[100];`
`int i;`
`srand(time(NULL));`
`for(i=0; i<100; i++)`
`{`
`vektor[i] = rand() % 6 - 2;`
`}`
`printf("De olika talen slumpades`
`1 gång");`
`return;`
`}`

3

b) `#include <string.h>`
`int compare(char *s1, char *s2)`
`{`
`if(strcmp(s1, s2) == 0)`
`{`
`return 1;`
`}`
`else`
`{`
`return 0;`
`}`
`}`

3

1/6



```
1. strcpy(f1.titel, "The dark knight");  
   f1.year = 2008;
```

2

```
2. printf("%d", strlen(f1.titel));
```

2

```
3. strcpy(f1, f2);
```

0

```
4. strcpy(kopia, f2.titel);
```

2

```
5. FILE * fPointer;
```

```
   fPointer = fopen("filmer.dat", "rb");
```

```
   fread(fPointer, f2, sizeof(film), 1);
```

2

```
   fclose(fPointer);
```

18



```

int binarySearch (int *vektor, int antalTal, int key)
{
    int low = 0;
    int high = antalTal - 1;
    int mid = -1;
    int found = 0;
    while (low <= high && !found)
    {
        mid = (low + high) / 2;
        if (mid == key)
            found = 1;
        else if (mid < key)
            low = mid + 1;
        else
            high = mid - 1;
    }

    if (found)
        return mid;
    else
        return -1;
}

```

15



a) Utskriften blir: 21

3

b) Basfallet är villkoret $if(a == 6)$,
alltså när a är lika med 6.

Om man anropar rek med parametern 7
så kommer basfallet aldrig att nås, och
programmet kommer gå in i en evighetsloop
och till slut kraschar programmet när
minnet tar slut.

2

1/5



int sort är fel! I stället ska det stå
void sort, eftersom funktionen returnerar
ingenting.

for (j=1... är fel! I stället ska det
stå for (j=0...

if (num[j] < num[j+1]) är fel!

I stället ska det stå

if (num[j] > num[j+1])

tmp = num[j+1] är fel!

I stället ska det stå tmp = num[j]

for (i=0; i <= 100; i++) är fel!

I stället ska det stå for (i=0; i < 100; i++)