

## Programmeringsteknik med C och Matlab

---

**Skrivtid:** 9-13.

**Hjälpmedel:** Pennor.

**Antal uppgifter:** 7.

**Instruktioner. OBS! Läs igenom instruktionerna noggrant innan du börjar lösa uppgifterna.**

- Börja med att skriva ditt namn och personnummer på första bladet. Skriv sedan ditt kodnummer (från försättsbladet) **på varje blad**.
- Skriv i mån av plats dina lösningar direkt i tentamen.
- Om du behöver mer plats, skriv dina lösningar på extrablad. Lös bara en uppgift på varje extrablad. Se till att ditt **kodnummer** och **uppgiftens nummer** är tydligt markerade på varje extrablad du lämnar in.
- Se till att den C-kod som ingår i dina lösningar är vettigt indenterad.
- **Skriv tydligt.** Om vi inte kan läsa dina lösningar kan vi inte ge poäng för dem.
- Observera att tentamen är tryckt dubbelsidig. Läs alltså båda sidorna av varje blad.

**Betygsättning:** Totalt går det maximalt att uppnå 44 poäng.

- För betyget **3** (godkänt) krävs **22 poäng**.
- För betyget **4** krävs **29 poäng**.
- För betyget **5** krävs **36 poäng**.

Kodnummer: \_\_\_\_\_

## Uppgift 1

(5 Poäng)

Betrakta följande C-program:

```
#include <stdio.h>

int main(void){
    int a = 5;
    int b = a -3;
    char c = 'd';
    double d = 3.14;
    double e = 4;

    printf("%d\n", a); /* Utskrift 1 */

    printf("%d\n", a/b); /* Utskrift 2 */

    d -=a;

    printf("%lf\n", d); /* Utskrift 3 */

    printf("%.2lf\n", a/e); /* Utskrift 4 */

    printf("%c %d\n", c, b); /* Utskrift 5 */
    return 0;
}
```

Vad skriver programmet ut när det körs?

Utskrift 1:	5
Utskrift 2:	2
Utskrift 3:	-1.860000
Utskrift 4:	1.25
Utskrift 5:	d 2

Σ:

Kodnummer: \_\_\_\_\_

---

## Uppgift 2

(3 Poäng)

En triangels yta är dess bas gånger dess höjd genom två. Skriv en C-funktion som givet basen och höjden som inparametrar returnerar motsvarande triangels yta. Din funktion ska ha följande prototyp:

```
double triangleArea(double base, double height);
```

Lösningförslag:

```
double triangleArea(double base, double height){  
    return (base * height) / 2;  
}
```

$\Sigma$ :

Kodnummer: \_\_\_\_\_

---

### Uppgift 3

(6 Poäng)

a) Skriv en funktion med prototypen

```
double multiplyArray(double array[], int length);
```

som returnerar produkten av de `length` första elementen i `array`.

Lösningsförslag:

```
double multiplyArray(double array[], int length){
    int i;
    double result = 1; /* An empty product equals 1 */
    for(i = 0 ; i < length ; i++){
        result *= array[i];
    }
    return result;
}
```

b) Skriv en `main`-funktion som

- deklarerar en `double`-array av längd tre,
- fyller arrayen med tre flyttal (tex, 3.5, 4.7 och -2.3),
- anropar `multiplyArray` för att räkna ut produkten av de tre talen, och
- skriver ut det värde `multiplyArray` returnerar med tre decimalers precision.

Lösningsförslag:

```
int main(void){
    double array[] = {3.5, 4.7, -2.3};
    double product;
    product = multiplyArray(array, 3);
    printf("The product is %.3lf.\n", product);
    return 0;
}
```

Σ:

Kodnummer: \_\_\_\_\_  
\_\_\_\_\_

Kodnummer: \_\_\_\_\_

#### Uppgift 4

(9 Poäng)

Betrakta följande C-funktioner:

```
int vanilla(int chocolate, int strawberry){
    if(chocolate <= strawberry && strawberry > -2){
        return chocolate + strawberry;
    }else{
        return strawberry +1;
    }
}
```

```
int lily(int rose, int tulip, int gardenia){
    int buttercup=0, edelweiss = tulip;
    do{
        edelweiss -= gardenia;
        buttercup++;
    }while(buttercup < rose);
    return edelweiss;
}
```

```
int horse(int cow, int pig){
    int cat, dog, mouse;

    cat = cow % pig;
    dog = pig + cat;
    if((cow/cat) > cat){
        cat ++;
    }
    for(mouse = dog ; mouse >= cat ; mouse--){
        dog--;
    }
    return dog;
}
```

Vilket returvärde ger följande tre anrop?

vanilla(2,3);	5
lily(2,3,4);	-5
horse(15,4);	3

$\Sigma$ :

Kodnummer: \_\_\_\_\_

---

### Uppgift 5

(6 Poäng)

Du har fått anställning på det nystartade bokförlaget Aristoi, för att skriva program som hanterar förlagets utgivning av böcker. För att komma igång måste du till att börja med definiera lämpliga datatyper för att hantera *författare* och *böcker*.

De uppgifter som är intressanta för en författare är

- Förnamn (max 200 tecken),
- Efternamn (max 200 tecken), och
- Födelseår (ett heltal).

a) Definiera en datatyp i C som är lämplig för att hantera författardata.

Lösningsförslag:

```
typedef struct{
    char firstName[201];
    char lastName[201];
    int birthYear;
} authorData;
```

De uppgifter som är intressanta för en bok är

- Titel (max 1000 tecken),
- Författare, och
- Utgivningsnummer (ett positivt heltal).

b) Definiera en datatyp i C som är lämplig för att hantera bokdata.

Lösningsförslag:

```
typedef struct{
    char title[1001];
    authorData author;
    unsigned int publishingNumber;
} bookData;
```

Σ:

Kodnummer: \_\_\_\_\_

---

## Uppgift 6

(10 Poäng)

Givna är följande datatypsdefinitioner:

---

```
typedef enum {MOTORCYCLE, CAR, TRUCK} vehicleType;

typedef struct {
    vehicleType type;
    double weight;
    double length;
} vehicle;
```

---

a) Skriv en funktion

```
vehicle vehicleNew(vehicleType newType, double newWeight, double newLength);
```

som returnerar en struktur av typen `vehicle` med elementen `type`, `weight` och `length` satta till funktionens respektive inparametrar.

Lösningsförslag:

```
vehicle vehicleNew(vehicleType newType, double newWeight, double newLength){
    vehicle newVehicle = {newType, newWeight, newLength};
    return newVehicle;
}
```

b) Skriv en funktion

```
vehicle * allocVehicleArray(int length);
```

som allokerar en `vehicle`-array av längd `length` och returnerar en pekare till arrayen.

Lösningsförslag:

```
vehicle * allocVehicleArray(int length){
    vehicle * array = (vehicle *) calloc(length, sizeof(vehicle));
    return array;
}
```

c) Skriv en funktion

```
void countVehicles(vehicle * array, int length, int * mcs, int * cars, int * trucks);
```

som

- går igenom de `length` första elementent i `array`,
- sparar antalet motorcyklar den sett i variabeln som `mcs` pekar på,
- sparar antalet bilar den sett i variabeln som `cars` pekar på, och
- sparar antalet lastbilar den sett i variabeln som `trucks` pekar på.

Lösningsförslag:

```
void countVehicles(vehicle * array, int length, int * mcs, int * cars, int * trucks){
    int i;
    *mcs = 0;
    *cars = 0;
    *trucks = 0;
    for(i = 0; i < length ; i++){
        if(array[i].type == MOTORCYCLE){
            (*mcs)++;
        }else if(array[i].type == CAR){
            (*cars)++;
        }else{
            (*trucks)++;
        }
    }
}
```

Σ:



Kodnummer: \_\_\_\_\_  
\_\_\_\_\_

Kodnummer: \_\_\_\_\_

---

### Uppgift 7

(5 Poäng)

Bestäm vilka av följande utsagor som är sanna och vilka som är falska. För varje riktigt svar ges 1 poäng, för varje felaktigt svar -1 poäng. Om totalsumman för uppgiften blir negativ ges totalt 0 poäng.

- |  | sant                     | falskt                   |
|--|--------------------------|--------------------------|
| a) Alla C-program måste inkludera <code>stdio.h</code> . | <input type="checkbox"/> | <input type="checkbox"/> |

Falskt. `stdio.h` måste bara inkluderas av program som vill använda sig av funktionerna som finns deklarerade där.

- |   |                          |                          |
|---|--------------------------|--------------------------|
| b) En <i>kompilator</i> är ett program som översätter programkod skriven i ett högnivåspråk till maskinkod. | <input type="checkbox"/> | <input type="checkbox"/> |
|---|--------------------------|--------------------------|

Sant

- |  |                          |                          |
|--|--------------------------|--------------------------|
| c) Om <code>ap</code> är en <code>int</code> -pekare så refererar <code>*ap</code> till den <code>int</code> -variabel vars adress är lagrad i <code>ap</code> . | <input type="checkbox"/> | <input type="checkbox"/> |
|--|--------------------------|--------------------------|

Sant

- |   |                          |                          |
|---|--------------------------|--------------------------|
| d) Funktionen <code>malloc</code> returnerar antalet minnesceller den reserverat för programmets räkning. | <input type="checkbox"/> | <input type="checkbox"/> |
|---|--------------------------|--------------------------|

Falskt. `malloc` returnerar en pekare till minnet den reserverat.

- |  |                          |                          |
|--|--------------------------|--------------------------|
| e) Funktionen <code>scanf</code> returnerar antalet platshållare den lyckats matcha. | <input type="checkbox"/> | <input type="checkbox"/> |
|--|--------------------------|--------------------------|

Sant

$\Sigma$ :