

1) (1p) Skriv den tilldelningssats i C som motsvarar formeln :

$$x = \frac{2y}{y+3} + 7/z$$

2) (1p) Vilka värden har följande uttryck?

- a) `5 / 2 % 6`
 b) `'M' < 'm' && ('3' - '0') == 3`
-

3) (1p) Datatypen

```
struct fastighet
{
    char bet[20]; // Fastighetsbeteckning ex Lugnet 5
    int tax;      // Taxeringsvärde i hela kr
    int pris;     // Utgångspris vid försäljning
};
```

avbildar fastigheter till försäljning hos en mäklare. Definiera en variabel av denna typ och initiera den med beteckning Lugnet 5, taxeringsvärde 1200000 kr och utgångspriset 1500000.

4) (1p) Vad blir slutvärdet på variablerna a resp. b då nedanstående kod körs?

```
for (a = 2, b = 10; a <= b; a++, b -= 2)
{
    if (a == b)
        a--;
    else
        b--;
}
```

5) (1p) Antag att du har en vektor, `fastvek`, bestående av 25 fastigheter enligt uppgift 3 ovan. Skriv den kod som nollställer vektorn. Fastighetsbeteckningen ska nollställas med en tom sträng.

6) (2p) Skriv en funktion, `skriv_fastighet`, som tar en fastighet enligt uppgift 3 ovan som parameter och skriver ut fastigheten. Funktionshuvud enligt:

```
void skriv_fastighet(struct fastighet f);
```

7) (2p) Skriv en funktion, `bensinmatare`, som tar mängden bensin i tanken som parameter och som returnerar mängden bensin avrundat till närmaste hela liter, om mängden är 10.0 eller mer, annars ska 0 returneras.

8) (2p) Skriv en funktion, `skriv_vek`, som tar en heltalsvektor och antalet element i vektorn som parametrar och som skriver ut vektorns element på skärmen med 10 tal per rad.

- 9) (2p) Skriv en funktion som tar ett filnamn som parameter och som bestämmer filens typ. Du kan anta att filnamnet alltid består av ett namn följt av en punkt och filtyp. Exempelvis har filnamnet Hello.obj filtypen obj. Funktionshuvud enligt :

```
void filtyp(char *filnamn, char *filtyp);
```

- 10)(2p)Skriv en funktion, las_fastighet, som tar en pekare till en struct enligt uppgift 3 ovan som parameter och som läser in fastighetens data. Funktionshuvud enligt :

```
void las_fastighet(struct fastighet *fp);
```

- 11)(5p)Skriv ett program som upprepat (avslut 0) läser in en fastighets taxeringsvärde och skriver ut försäljningspriset om taxeringsvärdet är 75 % av försäljningspriset. Som variabel ska du använda en struct enligt uppgift 3 ovan.
-

- 12)(5p)Skriv ett program som simulerar en bensinmätare på en bil. Programmet ska upprepat (avslut 0.0) fråga hur mycket bensin som finns kvar i tanken. Finns det 10.0 liter eller mer ska programmet skriva ut mängden bensin avrundat till närmaste hela liter. Är mängden bensin mindre än 10.0 liter ska istället en varningstext som uppmanar föraren att tanka bilen, skrivas ut. Du ska anropa funktionen bensinmatare i uppgift 7 ovan.
-

- 13)(5p)Skriv ett program som centrerar en vektor bestående av ett inläst antal (max 100) slumpade tvåsiffriga heltal. Att centrera en vektor innebär att man tar reda på spridningen (max – min) och subtraherar sedan alla tal i vektorn med spridningen. Du ska skriva ut vektorn genom att anropa funktionen skriv_vek i uppgift 8 ovan både före centreringen och efter. För full poäng krävs också att centreringen görs av en funktion som tar vektorn och antalet element i vektorn som parametrar.
-

- 14)(5p) Textfilen Filer.txt innehåller ett antal filnamn, en per rad enligt :

```
Hello.c
Namn.txt
Lotto.c
. . .
```

Skriv ett program som upprepat (avslut ENTER) läser in en filtyp och som skriver ut alla filnamn i Filer.txt, som har denna filtyp. Du ska anropa funktionen i uppgift 9 ovan för bestämning av filtyp.

- 15)(5p)En mäklarfirma har alla sina sålda fastigheter i binärfilen fastigheter.dat i form av structar med fastighetsbeteckning, taxeringsvärde och försäljningspris. Skriv ett program som läser alla fastigheter från filen och räknar ut medelvärdet av de antal

procent som taxeringsvärdet utgör av försäljningspriset. Använd structen i uppgift 3 ovan.

Lösningar till tentamen i Programmering , 050108.

- 1) `a = 2*y / (y+3) + 7/z;`
- 2) a) 2 b) 1
- 3) `struct fastighet f = {"Lugnet 5", 1200000, 1500000};`
- 4) `a = 4;`
`b = 2;`
- 5)

```
for (i = 0; i < 25; i++)
{
    fastvek[i].bet[0] = '\\0';
    fastvek[i].tax = 0;
    fastvek[i].pris = 0;
}
```
- 6)

```
void skriv_fastighet(struct fastighet f)
{
    printf("Beteckning : %s\n", f.bet);
    printf("Taxeringsvärde : %d\n", f.tax);
    printf("Beteckning : %d\n", f.pris);
}
```
- 7) Se uppgift 12.
- 8) Se uppgift 13.
- 9) Se uppgift 14
- 10)

```
void las_fastighet(struct fastighet *fp)
{
    printf("Ge fastighetsbeteckning : ");
    gets(fp->bet);
    printf("Ge taxeringsvärde : ");
    scanf("%d", &fp->tax);
    printf("Ge pris : ");
    scanf("%d", &fp->pris);
}
```
- 11)

```
/* Uppg11.c */
#include <stdio.h>
void main()
{
    struct fastighet
    {
        char bet[20];
        int tax;
        int pris;
    }f;
    printf("Ge taxeringsvärde (avslut 0): ");
    scanf("%d", &f.tax);
    while (f.tax != 0)
    {
        f.pris = (int) (f.tax / 0.75);
        printf("Normalt försäljningspris = %d\n", f.pris);
        printf("Ge taxeringsvärde (avslut 0): ");
        scanf("%d", &f.tax);
    }
}
```

```

12)  /* Uppg12.c */

#include <stdio.h>
int bensinmatare(float liter)
{
    if (liter >= 10.0)
    {
        return liter + 0.5;
    }
    return 0;
}

void main()
{
    float literbensin;
    int svar;

    printf("Hur mycket bensin finns i tanken (avslut 0): ");
    scanf("%f", &literbensin);
    while (literbensin != 0.0)
    {
        svar = bensinmatare(literbensin);
        if(svar == 0)
        {
            printf("Du måste tanka bilen!\n");
        }
        else
        {
            printf("Bensinmätaren visar = %d\n", svar);
        }
        printf("Hur mycket bensin finns i tanken (avslut 0): ");
        scanf("%f", &literbensin);
    }
}

```

```

13)  /* Uppg13.c */

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <time.h>

void skriv(int v[], int nr)
{
    int i;

    for (i = 0; i < nr; i++)
    {
        printf("%d ", v[i]);
        if ((i+1) % 10 == 0)
            printf("\n");
    }
}

void centrera(int v[], int nr)
{
    int min = v[0], max = v[0], i;

    for (i = 1; i < nr; i++)
    {
        if (v[i] > max)
        {
            max = v[i];
        }
    }
}

```

```

    }
    else if (v[i] < min)
    {
        min = v[i];
    }
}
for (i = 0; i < nr; i++)
{
    v[i] -= (max-min)/2;
}
}

void main()
{
    int vek[100], i, antal;

    srand((unsigned)time(NULL));
    printf("Ge antalet tal i vektorn (max 100) : ");
    scanf("%d", &antal);
    for (i = 0; i < antal; i++)
    {
        vek[i]= rand() % 90 + 10;
    }
    printf("\nSlumpad vektor : \n");
    skriv(vek, antal);
    centrera(vek, antal);
    printf("\n\nCentrerad vektor : \n");
    skriv(vek, antal);
    getch();
}

```

14) /* Uppg14.c */

```

#include <stdio.h>
#include <conio.h>
#include <string.h>

void filtyp(char *filnamn, char *filtyp)
{
    int i = 0, k = 0;

    while(filnamn[i] != '.')
    {
        i++;
    }
    i++;
    while(filnamn[i] != '\0')
    {
        filtyp[k] = filnamn[i];
        i++;
        k++;
    }
    filtyp[k]= '\0';
}

void main()
{
    FILE *tsin;
    char filnamn[20], ftyp[10], sftyp[10];

    tsin = fopen("Filnamn.txt", "rt");
    printf("Vilken filtyp söks? ");
}

```

```

gets(sftyp);
while(fscanf(tsin, "%s", filnamn) != EOF)
{
    filtyp(filnamn, ftyp);
    if (strcmp(sftyp, ftyp) == 0)
    {
        printf("%s\n", filnamn);
    }
}
fclose(tsin);
getch();
}

```

15)

```

#include <stdio.h>
#include <conio.h>

void main()
{
    FILE *bsin;
    struct fastighet
    {
        char bet[20];
        int tax;
        int pris;
    }f;
    float procsum = 0.0;
    int count = 0;

    bsin = fopen("Fastigheter.dat", "rb");
    fread(&f, sizeof(struct fastighet), 1, bsin);
    while(!feof(bsin))
    {
        procsum += (float)f.tax / f.pris * 100;
        count++;
        fread(&f, sizeof(struct fastighet), 1, bsin);
    }
    printf("Medelprocenten = %.1f", procsum / count);
    fclose(bsin);
    getch();
}

```