

Lösningsförslag till datorövning 4, Programmering C.

```
/* Uppg4a.c */
#include <stdio.h>

void main()
{
    int index, vektor[4];

    /* Loopa framlänges, läs tal*/
    for ( index = 0; index < 4; index++ )
    {
        printf( "Ge vektor[%d]: ", index );
        scanf( "%d", &vektor[index] );
    }

    /* Loopa baklänges, skriv tal */
    for ( index = 3; index >= 0; index-- )
    {
        printf( "%5d", vektor[index] );
    }
    printf( "\n" );
}

/* Uppg4b.c */
#include <stdio.h>

void main()
{
    int index;
    float vektor[5], min, max, summa;

    /* Loopa läs tal*/
    for ( index = 0; index < 5; index++ )
    {
        printf( "Ge vektor[%d]: ", index);
        scanf( "%f", &vektor[index] );
    }

    /* Sätt startvärden för variabler */
    min = vektor[0];
    max = vektor[0];
    summa = vektor[0];

    /* Loopa igenom resten av vektorn */
    for ( index = 1; index < 5; index++ )
    {
        summa = summa + vektor[index];

        if ( min > vektor[index] )
        {
            min = vektor[index];
        }
        else if ( max < vektor[index] )
        {
            max = vektor[index];
        }
    }
    printf( "Största = %.2f\nMinsta = %.2f\nSumma = %.2f\n", max, min, summa);
}
```

```

/* Uppg4c.c */

#include <stdio.h>
#include <stdlib.h> /* för att slumpa */
#include <time.h>    /* för att slumpa */

void main()
{
    int index, min_index, right, temp;
    int vektor[100];

    randomize();

    /* Slumpa 100 tresiffriga tal,
       skriv 10 tal per rad */
    for ( index = 0; index < 100; index++ )
    {
        vektor[index] = 100 + random(900);
        printf( "%5d", vektor[index] );
        if ( (index + 1) % 10 == 0 )
        {
            /* Ordna ny rad */
            printf( "\n" );
        }
    }

    /* Sortera vektorn */
    for ( index = 0; index < 99; index++ )
    {
        /* Sök det minsta bland de till höger om index */
        min_index = index;
        for ( right = index + 1; right < 100; right++ )
        {
            if ( vektor[right] < vektor[min_index] )
            {
                min_index = right;
            }
        }

        /* Byt så att minsta hamnar vid index. OM det behövs */
        if ( min_index > index )
        {
            temp = vektor[min_index];
            vektor[min_index] = vektor[index];
            vektor[index] = temp;
        }
    }

    /* Skriv ut vektorn, 10 tal per rad */
    printf( "\n\nSorterad vektor:\n" );
    for ( index = 0; index < 100; index++ )
    {
        printf( "%5d", vektor[index] );
        if ( (index + 1) % 10 == 0 )
        {
            /* Ordna ny rad */
            printf( "\n" );
        }
    }
}

```

```

/* Uppg4d.c */

#include <stdio.h>

void main()

```

```

{
    char buf[80];
    int index;

    printf( "Skriv sträng med ord: " );
    gets( buf );

    /* Loopa till strängslut */
    for ( index = 0; buf[index] != '\0'; index++ )
    {
        if ( buf[index] == ' ' )
        {
            putchar( '\n' );
        }
        else
        {
            putchar( buf[index] );
        }
    }
}

```

/* Uppg4e.c */

```

#include <stdio.h>
#include <string.h> /* strlen-funktionen */

void main()
{
    char buf[80];
    int index;

    printf( "Skriv en sträng: " );
    gets( buf );

    /* Loopa igenom hela strängen baklänges */
    for ( index = strlen(buf) - 1; index >= 0; index-- )
    {
        putchar( buf[index] );
    }
}

```

/* Uppg4f.c */

```

#include <stdio.h>

void main()
{
    struct medlem_s
    {
        int nr;
        char namn[40];
        int telnr;
    } medlem;

    printf( "Skriv namn: " );
    gets( medlem.namn );

    printf( "Skriv nr och telnr: " );
    scanf( "%d%d", &medlem.nr, &medlem.telnr );

    /* Skriv ut det inlästa */
    printf("Medlemsnummer: %d\nNamn: %s\nTelefonnummer: %d\n",
        medlem.nr, medlem.namn, medlem.telnr );
}

```

```

/* Uppg4g.c */
#include <stdio.h>

void main()
{
    struct rtal
    {
        int talj;
        int namn;
    } tal1, tal2, summa;
    char tecken;

    printf( "Skriv bråktal a / b: " );
    scanf( "%d%c%d", &tal1.talj, &tecken, &tal1.namn );

    printf( "Skriv annat bråktal a/b: " );
    scanf( "%d%c%d", &tal2.talj, &tecken, &tal2.namn );

    /* Beräkna summabräk, skriv ut */
    summa.talj = tal1.talj * tal2.namn + tal2.talj * tal1.namn;
    summa.namn = tal1.namn * tal2.namn;

    printf( "%d/%d + %d/%d = %d/%d\n", tal1.talj, tal1.namn,
            tal2.talj, tal2.namn, summa.talj, summa.namn );
}

```

```

/* Uppg4h.c */
#include <stdio.h>

void main()
{
    float diametrar[] = {1.80, 2.15, 2.50, 2.80, 3.15,
                        3.45, 3.80, 4.15, 4.50, 4.80,
                        5.45, 6.15, 6.80, 7.50, 8.15};

    int index;
    float diameter;

    printf ( "Skriv en diameter: " );
    scanf( "%f", &diameter );

    /* Sök index med närmast större diameter */
    index = 0;
    while ( (index < 15) && (diameter >= diametrar[index]) )
    {
        index = index + 1;
    }

    if ( index == 15 )
    {
        puts( "Det fanns ingen skruv med större diameter" );
    }
    else
    {
        printf("%.2f är närmast större skruvdiameter.\n", diametrar[index]);
    }
}

```

```

/* Uppg4i.c */
#include <stdio.h>

void main()
{
    struct medlem_s

```

```

{
    int nr;
    char namn[40];
    int telnr;
} medlemmar[10];

int antal, medlemsnr, index;

/* while-loop som upprepat läser poster,
   avslutas då 10 poster lästs in ELLER
   om medlemsnr 0 läses in */
antal = 0;
printf( "Skriv nr (0 avslutar): " );
scanf( "%d", &medlemmar[antal].nr );
while ( ( antal < 10 ) && ( medlemmar[antal].nr != 0 ) )
{
    printf( "Skriv telnr: " );
    scanf( "%l", &medlemmar[antal].telnr );
    getchar(); /* rensa undan ENTER-tecknet */

    printf( "Skriv namn: " );
    gets( medlemmar[antal].namn );

    /* Läs bara in nästa medlemsnr om det finns plats */
    antal = antal + 1;
    if ( antal < 10 )
    {
        printf( "Skriv nr (0 avslutar): " );
        scanf( "%d", &medlemmar[antal].nr );
    }
}

/* Läs in medlemsnummer, men bara om det finns poster */
if ( antal > 0 )
{
    printf( "\nGe ett medlemsnummer: " );
    scanf( "%d", &medlemsnr );
    putchar( '\n' );

    /* Sök posten med medlemsnumret */
    index = 0;
    while ( ( index < antal ) &&
            ( medlemmar[index].nr != medlemsnr ) )
    {
        index = index + 1;
    }

    /* Skriv postdata om hittad */
    if ( index < antal )
    {
        printf( " Nr: %-5d Namn: %-50s%d\n", medlemmar[index].nr,
                medlemmar[index].namn, medlemmar[index].telnr );
    }
    else
    {
        printf( " Det finns ingen medlem med nr: %d\n", medlemsnr );
    }
}
else
{
    puts( "Det blev inga poster inlagda." );
}
}

/* Uppg4j.c */

#include <stdio.h>
#include <string.h> /* strlen */

```

```

void main()
{
    char buf[10];
    int index, faktor, summa;

    printf( "Skriv sifror: " );
    gets( buf );

    /* Sökt loop-lösning */
    summa = 0;
    faktor = 1;
    for ( index = strlen(buf) - 1; index >= 0; index-- )
    {
        summa = summa + ( buf[index] - '0' )*faktor;
        faktor = faktor * 10;
    }

    puts( "Efter looplösning:" );
    printf( "2 * %s = %d\n", buf, 2*summa );

    /* Enklare, läs från strängen som från tangentbordet */
    sscanf( buf, "%d", &summa );
    puts( "\Efter sscanf:" );
    printf("2 * %s = %d\n", buf, 2*summa);
}

```

/* Uppg4k.c */

```

#include <stdio.h>
#include <stdlib.h> /* random */
#include <time.h> /* random */

void main()
{
    int nytt_nr, antal, index, right;
    int rad[7];

    /* Logiska variabler, utnyttjas för att
       hejda loop som försöker stoppa in nytt_nr i rad */
    int nr_fanns_redan, instoppat;

    randomize();

    /* Slumpa första */
    rad[0] = 1 + random(35);
    antal = 1;

    /* Slumpa resten, håll rad sorterad */
    while ( antal < 7 )
    {
        nytt_nr = 1 + random(35);

        /* Försök stoppa in nytt_nr */
        index = 0;
        nr_fanns_redan = 0;
        instoppat = 0;
        while ( !( instoppat || nr_fanns_redan ) )
        {
            if ( index == antal )
            {
                /* OK, nytt_nr ska ligga sist */
                rad[antal] = nytt_nr;
                instoppat = 1;
            }
            else
            {
                nr_fanns_redan = ( rad[index] == nytt_nr );
            }
        }
    }
}

```

```

    if ( !nr_fanns_redan )
    {
        if ( nytt_nr < rad[index] )
        {
            /* Ok stoppa in här */
            /* Flytta först allt öster om index 1 steg österut */
            for ( right = antal; right > index; right-- )
            {
                rad[right] = rad[right - 1];
            }
            rad[index] = nytt_nr;
            instoppat = 1;
        }
        else
        {
            /* Kolla nästa index */
            index = index + 1;
        }
    }
}

if ( instoppat )
{
    /* OK, fanns inte förut*/
    antal = antal + 1;
}

/* Skriv ut raden */
for ( index = 0; index < 7; index++ )
{
    printf( "%5d", rad[index] );
}
putchar( '\n' );
}

```