

OPINTOJAKSON OHJELMOINTI (T0004) HARJOITUSTEN RATKAISUJA

Seuraavassa on esitetty ratkaisuja ohjelmoinnin kurssilla syksyllä 2003 esillä olleisiin laboratorioharjoituksiin ja kotitehtäviin. Kaikesta noudatetusta huolellisuudesta huolimatta, ratkaisuihin on saattanut jäädä epätarkkuuksia tai suoranaisia virheitäkin, joista toivotaan palautetta kurssin vetäjälle.

Ohjelmointitehtävien ratkaisut on testattu, mutta tunnetusti ohjelman virheettömyydestä voi varmistua vain todistamalla ohjelman oikeaksi. Yhtään tässä esitetyistä ohjelmista ei ole todistettu oikeaksi.

Huom! Ohjelmakoodin tekstitulostuksissa esiintyvät \204 ja \224 ovat ä:n ja ö:n oktaalikoodeja DOS:n käyttämässä merkkijärjestelmässä.

1 HARJOITUS

1.1

Esitä seuraavan algoritmin **tila** joka rivin jälkeen. Algoritmilta annetaan syötteet 3, -1 ja 4. (Huom: tämä **ei ole** C-kieltä!)

```
a := 1; b := -2; c := 3;
lue(d);
a := (b-d)*a;
b := a+(b+(c+(d+1)*c)*b)*a;
lue(a); lue(c);
kirjoita(a-c);
c := c+c*c;
```

Ratkaisu Algoritmin tila = algoritmien käyttämien muuttujien arvojen yhdistelmä. Tulostaminen ei muuta algoritmin tilaa.

rivi	Algoritmin tila			
	a	b	c	d
a := 1; b := -2; c := 3;	1	-2	3	?
lue(d);	1	-2	3	3
a := (b-d)*a;	-5	-2	3	3
b := a+(b+(c+(d+1)*c)*b)*a;	-5	155	3	3
lue(a); lue(c);	-1	155	4	4
kirjoita(a-c);	-1	155	4	4
c := c+c*c;	-1	155	20	4

1.2

Millaisen "reaalimaailman" ongelman seuraava algoritmi ratkaisee? (Huom: tämäkään **ei ole** C-kieltä!)

```
a := 0;
lue(b);
c := b;
while (c>0) {
    lue(d);
    a := a + d;
    c := c - 1;
}
```

```

}
if (b=0)
    kirjoita("heissulivei")
else
    kirjoita(a/b);

```

Kirjoita algoritmi uudelleen siten, että muuttujilla on nimet, joiden avulla algoritmin tehtävän voi ymmärtää helpommin! Täydennä algoritmia myös tulostusoperaatioin, jotka kertovat, mistä on kysymys. Ketä ensimmäinen parannus palvelee? Entä toinen?

Ratkaisu Jos $b:n$ arvoksi annetaan nollaa suurempi kokonaisluku, algoritmi laskee lukukäskyllä lue(d) luettavien lukujen keskiarvon ja tulostaa sen.

Stilisoitu versio:

```

summa := 0;
kirjoita("Tämä ohjelma laskee antamiesi lukujen keskiarvon.
        Syötä aluksi lukujen määrä!");
lue(lukujen_maara);
luettavia_lukuja_jaljella := lukujen_maara;
while (luettavia_lukuja_jaljella > 0) {
    kirjoita("Anna luku");
    lue(luku);
    summa := summa + luku;
    luettavia_lukuja_jaljella := luettavia_lukuja_jaljella -
        1;
}
if (lukujen_maara=0) {
    kirjoita("Lukuja oli 0 kappaletta.");
    kirjoita("Keskiarvoa ei voi laskea!");
}
else {
    kirjoita("Antamiesi lukujen keskiarvo on ");
    kirjoita(summa/lukujen_maara);
}

```

Edellinen muutos palvelee ylläpitoa, toinen käyttäjää.

1.3

Laadi algoritmi, joka lukee käyttäjän antamia lukuja niin kauan, kunnes sille syötetään lopetusmerkiksi 9999. Ohjelma laskee lukemansa luvut yhteen (ei lopetusmerkkiä) ja tulostaa summan.

Ratkaisu pseudokoodina:

```

luku:=0;
summa:=0;
while(luku!=9999)
{
    lue(luku);
    if(luku != 9999)
        summa:=luku + summa;
}
kirjoita(summa);

```

vaihtoehtoinen ratkaisu:

```

summa:=0;
lue(luku);
while(luku!=9999)
{
    summa:=luku + summa;
    lue(luku);
    if(luku != 9999)
        continue;
}
kirjoita(summa);

```

1.4

Henkilö voi olla köyhä (tulokoodi=3), keskituloinen (tulokoodi=2) tai rikas (tulokoodi=1). Keskituloisen mielestä eräs tuote on halpa, jos sen hinta on alle 500, kohtuuhintainen, jos hinta on alle 800 ja kallis, jos hinta on 800 tai yli. Yli 1200 hinta on mahdoton. Köyhän mielestä rajat ovat 200 euroa pienemmät, rikkaan mielestä 500 suuremmat. Mikään hinta ei ole rikkaalle mahdoton. Laadi algoritmi, joka luettuaan tulokoodin ja tuotteen hinnan ilmoittaa henkilön mielipiteen hinnasta.

Ratkaisu pseudokoodina:

```
tulokoodi:=0; hinta:=0; arvio:=0;
halpa:=500; kohtuu:=800; kallis:=1200;

lue(tulokoodi); // 1=rikas 2=keskituloinen 3=köyhä

if(tulokoodi=3)
    muutos = -200;
if(tulokoodi=1)
    muutos = +500;
halpa := halpa +muutos;
kohtuu := kohtuu +muutos;
kallis := kallis +muutos;

lue(hinta);
if(hinta<halpa)
    kirjoita("halpa");
if(hinta>halpa && hinta<kohtuu)
    kirjoita("kohtuuhintainen");
if(hinta>kohtuu && hinta < kallis)
    kirjoita("kallis");
if(hinta>kallis && tulokoodi != 1)
    kirjoita("mahdoton");
if(tulokoodi = 1 && hinta>kallis)
    kirjoita("kallis");
```

1.5

Tee algoritmi, joka toistaa seuraavaa: algoritmi pyytää käyttäjältä aikuisen henkilön ikää, kunnes syötetty ikä on järkevä. Iän on oltava välillä 18 -120. Kun ikä on kunnossa, kysytään henkilön sukupuoli (M tai N). Myös sukupuoli tarkistetaan silmukassa. Ikien syöttäminen lopetetaan, kun käyttäjä syöttää iäksi nollan. Algoritmi tulostaa miesten ja naisten lukumäärät ja ikien keskiarvot.

Ratkaisu pseudokoodina:

```
/* alustukset */
nLkm=0;          /* naisten lukumäärä */
mLkm=0;          /* miesten lukumäärä */
nSumma=0;        /* naisten ikien summa */
mSumma=0;        /* miesten ikien summa */
ika=1;
sukupuoli='E';

while (ika <> 0)
```

```
{
    while (ika <18 or ika >120)
        lue(ika);
    if (ika<>0)
    {
        while(sukupuoli<> 'N' and sukupuoli <> 'M')
            lue(sukupuoli);
        if (sukupuoli == 'N')
        {
            nLkm    := nLkm + 1;
            nSumma := nSumma + ika;
        }
        else
        {
            mLkm    := mLkm + 1;
            mSumma := mSumma + ika;
        }
    }
}
kirjoita("miesten lukumäärä ja ikien keskiarvo",
        mLkm, mSumma/mLkm);
kirjoita("naisten lukumäärä ja ikien keskiarvo",
        nLkm, nSumma/nLkm);
```

1.6

KOTITEHTÄVÄ SEURAAVAKSI KERRAKSI: Laadi algoritmi, joka lukee kymmenen kappaletta lukuja, joiden pitäisi olla kahtakymmentä suurempia. Luvut, jotka ovat pienempiä tai yhtä suuria kuin 20, hylätään. Algoritmi tulostaa syötetyistä hyväksytyistä luvuista pienimmän ja toiseksi pienimmän. Saat olettaa, että käyttäjän syöttämien lukujen joukossa on ainakin kaksi kelvollista.

Ratkaisu pseudokoodina:

```
laskuri:=10;
pienin:=0;
tokaPienin:=0;
while(laskuri>0)
{
    lue(luku);
    if(luku>20)
    {
        if(luku<pienin || pienin==0)
        {
            tokaPienin=pienin;
            pienin=luku;
        }
        if(luku>pienin && (luku<tokaPienin) || tokaPienin==0))
            tokaPienin:=luku;
    }
    laskuri:= laskuri - 1;
}
kirjoita(pienin);
kirjoita(tokaPienin);
```

2 HARJOITUS

2.1

Suunnittele algoritmi, joka pyytää käyttäjältä syötteeksi lukuja. Jokaisen lukukerran jälkeen kysytään käyttäjältä, haluaako tämä jatkaa lukujen syöttämistä. Algoritmi tulostaa lukemisen jälkeen suurimman ja pienimmän luvun arvon.

Ratkaisu pseudokoodina:

```
jatketaan:='K';
lue(luku);
pienin:=luku;
suurin:=luku;
lue(jatketaan);
while(jatketaan='K')
{
    lue(luku);
    if(luku>suurin) suurin:=luku;
    if(luku<pienin) pienin:=luku;
    lue(jatketaan);
}
kirjoita(suurin);
kirjoita(pienin);
```

2.2

Eräissä vaaleissa oli neljä ehdokasta 2,3, 4 ja 5. Laadi algoritmi, joka lukee ehdokkaiden numeroita, kunnes syötetään luku -1. Jos äänestyslippu on tyhjä tai hylätty, syötetään 0. Algoritmi tulostaa annettujen äänien kokonaismäärän, kunkin ehdokkaan saaman äänimäärän ja ~~äänestysprosentin vaaleissa~~ hyväksytyjen äänestyslippujen prosenttiosuuden

Ratkaisu pseudokoodina:

```
e2:=0;e3:=0;e4:=0;e5:=0; /*nollataan äänimäärälaskurit */
lkm:=0;
luku:=0;
while(luku<>-1)
{
    lue(luku);
    if (luku==0) {lkm:=lkm+1;}
    if (luku==2) {e2:=e2+1; lkm:=lkm+1;}
    if (luku==3) {e3:=e3+1; lkm:=lkm+1;}
    if (luku==4) {e4:=e4+1; lkm:=lkm+1;}
    if (luku==5) {e5:=e5+1; lkm:=lkm+1;}
}
hyv=e2+ e3+e4+e5;
kirjoita("Ääniä annettiin kaikkiaan", lkm);
kirjoita("Ehdokkaan 2 äänimäärä", e2);
kirjoita("Ehdokkaan 3 äänimäärä", e3);
kirjoita("Ehdokkaan 4 äänimäärä", e4);
kirjoita("Ehdokkaan 5 äänimäärä", e5);
kirjoita("Hyväksytyjä ääniä annettiin ",100*hyv/lkm);
```

2.3

C-koodissa esiintyvät seuraavat kokonaislukuvakiot: 1999 0x7D0 03716. Mikä luvuista on suurin? Mikä kokonaisluvusta 0x40 0077 62 062 0x3F on pienin?

Ratkaisu: Muunnetaan luvut samaan lukujärjestelmään, esim. desimaalisiksi: 0x7D0= 2000 ja 03716=1998, joten 0x7D0 on suurin. Vastaavasti todetaan, että 062 on luvuista pienin (0x40=64, 0077=63 62=62 062=50 ja 0x3F=63).

2.4

Laita seuraavat C-kielen vakiot suuruusjärjestykseen:

102.00 10e01 1010.E-01 1.0400E+02
10500e-2 0.000106E6 0.00103e5

Ratkaisu: 10e01 < 102.00 < 0.00103e5 < 1.0400E+02 < 10500e-2 < 0.000106E6 < 1010.E-01

2.5

Mitkä alla olevista ovat laillisia C-kielen vakioita? Mikä on laillisten vakioiden tietotyyppi?

-0.002	"666.66"	'9'	666.66	02123	-0.001e3
"0x10A"	'666.66'	0xAA	-01	'+'	01110
66.0-E01	"2F"	'2f'	2f	2F	2.F
2L	2.L	E1	.1	1.	1.1e.1

Ratkaisu: Laillisia **vakioita** ovat: -0.002 (float), '9' (char), 666.66 (float), 02123 (int, oktaaliesitys), -0.001e3 (float), 0xAA (int, heksadesimaaliesitys), '+' (char), 01110 (int, oktaaliesitys), 2.F (float), 2L (long), 2.L (long), .1 (float) ja 1. (float).

2.6

Laadi algoritmi, joka kysyy käyttäjältä lukujen lukumäärän ja sen jälkeen lukumäärän mukaisen määrän reaalitylukuja. Algoritmi tulostaa suurimman ja pienimmän luvun sekä laskee lukujen keskiarvon ja keskihajonnan.

Keskihajonta voidaan laskea kaavalla:

$$s = \sqrt{\frac{1}{n-1} \left(\sum_{i=1}^n x_i^2 - \frac{1}{n} \left(\sum_{i=1}^n x_i \right)^2 \right)}$$

Ratkaisu pseudokoodina:

```
summa:= 0;
summa2:=0;
kirjoita("Algoritmi laskee antamiesi lukujen keskiarvon ja
keskihajonnan");
kirjoita("Anna ensin lukujen lukumäärä");
```

```

lue(lkm);
laskuri:= lkm;
while (laskuri > 0) {
    lue(luku);
    summa:= summa + luku;
    summa2:=summa2 + luku*luku;
    laskuri:= laskuri - 1;
}
keskiarvo := summa/lkm;
keskihajonta:= sqrt((summa2-summa*summa/lkm)/(lkm-1));
/* sqrt = neliöjuuri */
kirjoita("Keskiarvo ", keskiarvo);
kirjoita("Keskihajonta ", keskihajonta);

```

3 HARJOITUS

3.1

Mikä on seuraavien lausekkeiden arvo?

- a) $(1+2*3)$ b) $10\%3*3 - (1+2)$ c) $((1+2)*3)$ d) $(5 == 5)$ e) $(x=5)$

Ratkaisu : a) 7 b) 0 c) 9 d) 1 e) 5

3.2

Olkoon $x=4$, $y=6$ ja $z=4$. Selvitä seuraavien lausekkeiden totuusarvo.

- a) $(x == 4)$ b) $(x != y - z)$ c) $(z=1)$ d) (y)

Ratkaisu : a) 1 eli tosi b) 0 eli epätosi $(4 != 2)$ c) 1 eli tosi d) 6 eli tosi

3.3

- Kirjoita ohjelma, jolla tulostat kuvaruudulle oman nimesi.
- Muuta tulostuvaa tekstiä siten, että nimen lisäksi tulostat myös ikäsi.
- Muuta ohjelmaa niin, että ikää varten otat käyttöön muuttujan, jolle annat alkuarvoksi oman ikäsi.
- Tulosta osoitteesi nimirivin alapuolelle.
- Tallenna ohjelma nimellä NIMI.C.
- Käännä ohjelmastasi suorituskelpoinen versio.
- Avaa DOS-ikkuna ja käynnistä ohjelmasi EXE-versio.

Ratkaisu :

```

#include <stdio>

int main(void)
{
    int age = 21;
    printf("Nimi on Ilpo Insin\224\224ri, ik\204 on %d vuotta,\n", age);
    printf("Osoite on: Vanha maantie 6\n");
    return 0;
}

```

3.4

a) Kirjoita ohjelma, jossa on kaksi kokonaislukumuuttujaa, joista toiselle annat alkuarvon. Tulosta muuttujien arvot. Mitä huomaat?

b) Lisää ohjelmaasi kahden merkkimuuttujan määrittely. Anna toiselle merkille alkuarvo. Tulosta merkit kuvaruudulle (print-funktiossa käytetään määrettä %c). Mitä huomaat?

c) Mitä tapahtuu, kun tulostat kokonaislukumuuttujan %c-tulostusmääreellä?

Ratkaisu :

```
#include <stdio.h>
```

```
int main()
{
    int luku=66, luku2;
    char merkki='A', merkki2;

    printf("Luvut %d %d\n", luku, luku2);
    printf("Merkit %c %c\n", merkki, merkki2);
    printf("Luku merkin,, %c\n", luku);
    return 0;
}
```

a- ja b-kohdissa havaitaan, että alkuarvottomalle muuttujalle tulostuu jokin arvo, jota ei voi ennalta ennustaa. Havainto tulee selvemmäksi, kun avataan useita DOS-ikkunoita ja ajetaan ohjelmaa niissä kaikissa.

c-kohdassa tulostuu se ASCII-merkki, jonka koodin desimaalinen vastine ko. luku on (modulo 256), tässä siis B.

3.5

Kirjoita ohjelma, joka tulostaa vuoden pituuden

a) tunteina b) minuutteina c) sekunteina.

Ratkaisu :

```
#include <stdio.h>
```

```
int main(void)
{
    int days = 365;
    long hours, mins, secs;
    hours = days * 24;
    mins = hours * 60;
    secs = mins * 60;

    printf("Vuosi on:\n");
    printf("%d p\204iv\204\204\n", days);
    printf("%d tuntia\n", hours);
    printf("%d minuuttia\n", mins);
    printf("%d sekuntia\n", secs);

    return 0;
}
```


3.6

Kirjoita ohjelma, joka muuttaa sekunteina annetun aikamäärän minuuteiksi ja sekunneiksi. Voit olettaa, että sekuntimäärä on kokonaisluku.

Ratkaisu :

```
#include <stdio>

int main(void)
{
    long luku, mins, secs;
    printf("Anna aika sekunteina:\n");
    scanf("%d", &luku);
    mins = luku/60;
    secs = luku%60;
    printf("\nSe on %d minuuttia ja %d sekuntia\n", mins, secs);
    return 0;
}
```

3.7

Laadi ohjelma, joka lukee käyttäjän antamia lukuja niin kauan, kunnes sille syötetään luku 0. Ohjelma laskee lukemansa luvut yhteen ja tulostaa summan.

Ratkaisu :

```
#include <stdio.h>
int main()
{
    int luku=1, summa=0;
    while(luku!=0)
    {
        printf("Anna luku (0 lopettaa)\n");
        scanf("%d",&luku);
        summa=summa+luku;
    }
    printf("Antamiesi lukujen summa on %d\n",summa);
}
```

3.8

Tee ohjelma, joka lukee hinnan. Jos hinta on alle 200, kirjoitetaan ONPAS HALPA. Jos hinta on 200 tai yli, kirjoitetaan KOHTUUHINTAINEN.

Ratkaisu :

```
#include <stdio.h>
int main()
{
    int hinta=0;
    printf("Anna hinta\n");
    scanf("%d",&hint);
    if(hinta<200)
        printf("ONPAS HALPA\n");
    else
        printf("KOHTUUHINTAINEN\n");
}
```

4 HARJOITUS

4.1

Kirjoita ohjelma, joka kysyy käyttäjältä ympyrän säteen ja tulostaa ympyrän piirin ja pinta-alan.

Ratkaisu :

```
#include <math.h> /* piin arvo vakiossa M_PI */
#include <stdio.h>

int main(void)
{
    float sade, piiri;
    float ala;
    /*input*/
    printf("Anna ympyr\204n s\204de: ");
    scanf("%f",&sade);

    /* laskenta */
    piiri = M_PI*2*sade;
    ala = M_PI*sade*sade;

    /*output*/
    printf("Ympyr\204n piiri on %f\n", piiri);
    printf("Ympyr\204n ala on %f", ala);
    return 0;
}
```

4.2

Kirjoita ohjelma, joka kysyy käyttäjältä tämän iän ja sukupuolen. Jos käyttäjä on mies ja ikä on alle 20, tulostetaan "Olet aivan liian nuori."; jos ikä on yli 20, mutta alle 40, tulostetaan "Olet jo keski-ikäinen." ja jos ikä on yli 40, tulostetaan "Olet itse asiassa vanha!". Jos kyseessä on nainen, tulostetaan alle 50-vuotiaalle "Oletpa Sinä nuori!" ja yli 50-vuotiaalle "Keski-ikä lähestyy, vaikkei sitä kyllä Sinusta huomaa!".

Ratkaisu :

```
#include <stdio.h>

int ika;
char puskuri[128];
char sukupuoli;

int main(void)
{
    printf ("Annapa ik\204si: ");
    gets(puskuri);
    sscanf (puskuri,"%d",&ika);
    printf ("Anna sukupuolesi: ");
    gets(puskuri);
    sscanf (puskuri,"%c",&sukupuoli);

    if (sukupuoli=='m' || sukupuoli=='M')
    {
```

```
        if (ika<0)
        {
            printf ("Ei onnistu: et ole viel\204 syntynyt\204\204n!");
        }
    else if (ika<20)
    {
        printf ("Olet aivan liian nuori!");
    }
    else if (ika<=40)
    {
        printf ("Olet jo keski-ikainen!");
    }
    else
    {
        printf ("Olet itseasiassa jo vanha!");
    }
}
else if (sukupuoli=='n' || sukupuoli=='N')
{
    if (ika<=50)
    {
        printf ("Oletpa sin\204 nuori!");
    }
    else
    {
        printf ("Keski-ik\204 l\204hestyy, vaikkei sit\204 sinusta
huomaakaan!");
    }
}
else printf("Tuntematon sukupuoli");
return 0;
}
```

4.3

Mitä arvoja muuttujat saavat seuraavia lauseita suoritettaessa?

```
int i = 2, j, k = -4;
double p, r = 1.23;
j = i + 10*(k+i);
k = j/3;
p = j/3;
i = k*i/2;
p = p/r/2;
i++; ++i;
--j; j--;
p++; ++p;
k += 10;
r -= 1000;
```

Ratkaisu :

	i	j	k	p	r
int i = 2, j, k = -4;	2	?	-4	-	-
double p, r = 1.23;	2	?	-4	?	1.23
j = i + 10*(k+i);	2	-18	-4	?	1.23
k = j/3;	2	-18	-6	?	1.23
p = j/3;	2	-18	-6	-6.000000	1.23
i = k*i/2;	-6	-18	-6	-6.000000	1.23
p = p/r/2;	-6	-18	-6	-2.439024	1.23
i++; ++i;	-4	-18	-6	-2.439024	1.23
--j; j--;	-4	-20	-6	-2.439024	1.23
p++; ++p;	-4	-20	-6	0.439024	1.23
k += 10;	-4	-20	4	0.439024	1.23
r -= 1000;	-4	-20	4	0.439024	-998.770000

4.4

Kirjoita ohjelma, joka kysyy käyttäjältä, haluaako tämä pelata kiinalaista niukkuuspeliä vai amerikkalaista ylivoimapeliä. Tämän jälkeen käyttäjää pyydetään antamaan jokin luku.

Jos käyttäjä halusi pelata kiinalaista peliä tulostetaan ilmoitus "Minun lukuni on " ja käyttäjän antamaa lukua yhtä suurempi luku sekä ilmoitus "Voitin niukasti!"

Jos taas pelataan amerikkalaista peliä, ohjelman valitsema luku on 1000 kertaa käyttäjän luku. Tuloksesta ilmoitetaan samoin käyttäjälle, mutta nyt viimeinen ilmoitus on "Voitin ylivoimaisesti!"

Ratkaisu :

```
#include <stdio.h>

int main(void)
{
    int luku=0, peli=0;

    while(peli!=1&&pelii!=2){
        printf("Pelataanko kiinalaista niukkuuspeli\204(1)\n");
        printf("vai amerikkalaista ylivoimapeli\204(2)?\n");
        scanf("%d",&peli);
    }
    while (luku<=0) {
        printf("Anna jokin nollaa suurempi kokonaisluku!\n");
        scanf("%d",&luku);
    }
    if(peli==1)
        printf("Minun lukuni on %d, voitin niukasti!\n", luku+1);
    else
        printf("Minun lukuni on %d, voitin ylivoimaisesti!\n", luku*1000);

    return 0;
}
```

4.5

Olkoon määritelty **int n, m, i; ja double r, q, z;** Kirjoita C-kielinen looginen lauseke, joka ilmaisee, että

- a) n on parillinen
- b) n ja m ovat erimerkkiset
- c) $3.1 < r < 3.2$
- d) i on suurempi kuin n, mutta pienempi kuin m
- e) joko i on 7 ja m on negatiivinen tai q on suurempi kuin r:n ja p:n erotus
- f) n:n ja m:n summa on jaollinen i:llä
- g) jompikumpi luvuista r ja q on suurempi kuin z, mutta eivät molemmat

Ratkaisu :

- a) $n \% 2 == 0$
- b) $n * m < 0$
- c) $r > 3.1 \ \&\& \ r < 3.2$
- d) $i > n \ \&\& \ i < m$
- e) $(i == 7 \ \&\& \ m < 0 \ || \ q > r - p) \ \&\& \ !((i == 7 \ \&\& \ m < 0 \ \&\& \ q > r - p)$
- f) $(n + m) \% i == 0$
- g) $(r > z \ || \ q > z) \ \&\& \ (r <= z \ || \ q <= z)$

4.6

KOTITEHTÄVÄ 1: Kirjoita keskustelevala ohjelma, joka laskee käyttäjän onnennumeron saatuaan syötteeksi (kokonaislukuina) käyttäjän pituuden, painon, iän, kengännumeron ja sukupuolen ('N', 'M' tai 'E' – ei osaa sanoa). Sukupuolitieto koodataan uuteen muuttujaan niin, että koodia 'N' vastaa 2, koodia 'M' vastaa 1 ja koodia 'E' vastaa nolla. onnennumero on jakojäännös, joka saadaan, kun jaettavana on uudelleenkoodatulla sukupuolella kerrottu pituuden, painon, iän ja kengännumeron summa, ja jakajana on luku 37.

Ratkaisu :

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

int main(void)
{
    char r[128];
    int onnennumero, sukupuoli;
    int pituus=0;
    int paino=0;
    int ika=0;
    int kengannumero=0;
    char sp_m='A';

    clrscr();
    printf("T\204m\204 ohjelma laskee onnennumerosi, ");
    printf("\nkun annat sille joitain tietoja.\n\n");

    while(sp_m!='M' && sp_m!='m' && sp_m!='N'
        && sp_m!='n' && sp_m!='E' && sp_m!='e')
    {
        printf("Anna sukupuolesi (M/N/E)!\t");
        gets(r);
        sscanf(r, "%c", &sp_m);
    }

    if(sp_m=='M' || sp_m=='m') sukupuoli=1;
    else if(sp_m=='N' || sp_m=='n') sukupuoli=2;
    else sukupuoli=0;

    while (paino <=0)
    {
```

```

        printf("Mik\204 on painosi?\t\t");
        gets(r);
        sscanf(r, "%d", &paino);
    }

    while (pituus <=0)
    {
        printf("Pituutesi?\t\t\t");
        gets(r);
        sscanf(r, "%d", &pituus);
    }

    while (ika <=0)
    {
        printf("Ik\204si?\t\t\t\t");
        gets(r);
        sscanf(r, "%d", &ika);
    }

    while (kengannumero <=0)
    {
        printf("Anna viel\204 keng\204nnumerosi:\t");
        gets(r);
        sscanf(r, "%d", &kengannumero);
    }
    clrscr();
    onnenumero=(sukupuoli*(pituus+paino+kengannumero+ika))%37;
    printf("\n\n\t\ta\ta\taOnnennumerosi on %d !\n", onnenumero);
    return 0;
}

```

4.7

KOTITEHTÄVÄ 2: Kirjoita ohjelma, joka kysyy käyttäjältä kokonaisluvun *n* ja lukee sen jälkeen puskurin käyttäjän antaman tekstin. Ohjelma salakirjoittaa tekstin ns. Caesar-salakirjoitusta käyttäen, jossa kukin puskurin (merkkitaulukon) merkki on korvattu merkillä, joka saadaan siirtymällä aakkostossa *n* merkkiä eteenpäin. Jos aakkoset loppuvat kesken, jatketaan aakkosten alusta. Voit olettaa, että tekstissä on vain isoja kirjaimia (muista tämä myös testatessasi!).

Jos siis esim. *n*=3, korvataan A D:llä, B E:llä, ..., Z Ö:llä, Å A:lla, Ä B:llä ja Ö:llä. OHJE: Ota käyttöön aputaulukko, jossa on aakkoston merkit peräkkäin oikeassa järjestyksessä.

Ratkaisu :

Huom! , Ž ja ™ ovat Å:na, Ä:nä ja Ö:nä DOS:ssa näkyvien merkkien ilmiäiset Windowsissa.

```

#include <stdio.h>

int main(void)
{
    char teksti[128];
    char apu[]="ABCDEFGHIJKLMNOPQRSTUVWXYZ Ž™";
    int i, n, shift, koko=29;
    char c;

```

```
printf("Kerro n: ");
gets(teksti);
sscanf(teksti, "%d", &shift);

printf("Anna salattava teksti (vain ISOJA kirjaimia!): ");
gets(teksti);

i = 0;
while ( (c = teksti[i]) != '\0')
{
    n = 0;
    while ( apu[n] != c ) { n++; }
    teksti[i] = apu[(n+shift)%koko];

    i++;
}
printf("Salattuna: %s\n", teksti);
return 0;
}
```

5 HARJOITUS

5.1

- Kirjoita funktio, joka tulostaa kuvaruudulle rivin, jossa on 10 tähtimerkkiä.
- Kirjoita funktio, joka tulostaa kuvaruudulle rivin, jossa on funktiolle parametrina annettu määrä tähtiä.
- Kirjoita funktio, jolle annetaan parametreina kuvion muoto ja rivien määrä, ja joka tulostaa seuraavanlaisia kuvioita:

```
*           *****           *****           *
**          *****          *****          **
***         ***             ***             ***
****        **              **              ****
*****       *               *               *****
```

Testaa funktioita kirjoittamalla keskustelevala pääohjelma, joka kutsuu funktioita.

Ratkaisu :

```
#include <stdio.h>
int tayta(char a, int n);
int kuvio(int t, int r);

int main(void)
{
    int tyyppi=0, maara=0;

    while (tyyppi<1 || tyyppi>4)
    {
        printf("Anna kuvion tyyppi (1,2,3 tai 4)!");
        scanf("%d",&tyyppi);
    }
    while (maara <= 0 || maara >80)
    {
        printf("Anna rivien määrä!");
```

```

        scanf("%d",&maara);
    }
    printf("\n\n\n");
    kuvio(tyyppi, maara);
}

int kuvio(int tyyppi, int maara)
{
    int i;
    for (i=1; i<=maara ; i++)
    {
        tayta(' ',(tyyppi == 3)*(i-1)+(tyyppi == 4)*(maara-i));
        tayta('*', (tyyppi == 1 || tyyppi == 4)*i+(tyyppi == 2|| tyyppi
== 3)*(maara-i+1));
        printf("\n");
    }
    return 0;
}

int tayta(char a, int n)
{
    int i=1;
    while (i<=n)
    {
        printf("%c",a);
        i++;
    }
    return 0;
}

```

5.2

Kirjoita ohjelma, joka saa syötteenään luokalle pidetyn kokeen arvosanoja ja tallettaa ne taulukkoon sopivalla tavalla. Arvosanoista lasketaan keskiarvo ja keskihajonta, jotka tulostetaan, samoin kuin arvosanojen jakauma muodossa:

```

0: ***
1: *****
2: *****
3: *****
4: *****
5: *****

```

Käytä hyväksesi tehtävän 1 funktioita.

Ratkaisu :

```

#include <stdio.h>
#include <math.h>

void tahti(int maara);

int main(void)
{
    char rivi[128];
    int i;
    int lkm[6]={0};
    int arvosana;

```



```

int ok;
double s=0.0, s2=0.0;
int n=0;

printf("Ohjelma laskee arvosanojen jakauman sek\204");
printf("keskiarvon ja keskihajonnan\n");
printf("antamistasi yksitt\204isist\204 arvosanoista (0..5).\n");
printf("Anna arvosanat, ep\204kelpo arvosana lopettaa lukemisen.\n");

do {
    gets(rivi);
    sscanf(rivi,"%d",&arvosana);
    if (ok = (arvosana>=0 && arvosana <=6)) lkm[arvosana]++;
} while (ok);

printf ("Arvosanajakauma:\n");
printf ("=====\n\n\n");
for(i=0;i<6;i++)
{
    printf("%d : ",i);
    tahti(lkm[i]);
    s +=i*lkm[i];
    s2+=i*i*lkm[i];
    n +=lkm[i];
}
printf("\n\nArvosanojen keskiarvo:    %f\n",s/n);
printf("          keskihajonta: %f\n",sqrt((s2-s*s/n)/(n-1)));

return 0;
}

void tahti(int maara)
{
    int indeksi=0;
    while(indeksi<maara)
    {
        printf("*");
        indeksi++;
    }
    printf("\n");
}

```

5.3

Kirjoita funktio, joka palauttaa parametriensa arvoina saamistaan kolmesta luvusta pienemmän. Kirjoita toinen funktio, joka laskee vastaavalla tavalla maksimin.

Ratkaisu :

```

#include <stdio.h>
int pienin(int x1,int x2,int x3);
int suurin(int x1,int x2,int x3);
int main(void)
{
    char r[128];
    int x1,x2,x3;

```

```

    printf("Anna kolme lukua blankolla erotettuna.\n");
    gets(r);
    sscanf(r, "%i %i %i", &x1, &x2, &x3);
    printf("Suurin luvuista on %d,\n", suurin(x1, x2, x3));
    printf("Pienin luvuista on %d.\n", pienin(x1, x2, x3));
    return 0;
}

int suurin(int x1, int x2, int x3)
{
    int pal_arvo=x1;
    if(x2 > pal_arvo)
        pal_arvo=x2;
    if(x3>pal_arvo)
        pal_arvo=x3;
    return pal_arvo;
}

int pienin(int x1, int x2, int x3)
{
    int pal_arvo=x1;
    if(x2<pal_arvo)
        pal_arvo=x2;
    if(x3<pal_arvo)
        pal_arvo=x3;
    return pal_arvo;
}

```

5.4

Kirjoita ohjelma, joka tulostaa kertotaulun käyttäjän antamaan kokonaislukuun saakka. Jos siis annetaan syötteeksi luku 4, tulostetaan taulukko:

	1	2	3	4
1	1	2	3	4
2	2	4	6	8
3	3	6	9	12
4	4	8	12	16

Ratkaisu :

```

#include <stdio.h>

int main(void)
{
    char puskuri[128];
    int i, j;
    int luku;
    printf("\n\n ohjelma laskee ja tulostaa\n");
    printf("kertotaulun antamaasi lukuun asti.\n\n");
    printf("Anna luku: ");
    gets(puskuri);
    sscanf(puskuri, "%d", &luku);

    printf("\n\n x |");
    for (i=1; i <= luku; i++) printf("%4d", i);
    printf("\n---|");
}

```

```

        for (i=1; i <= luku; i++) printf("----");
        printf("\n");
        for (i=1; i <= luku; i++)
        {
            printf("%4d|",i);
            for (j=1; j <= luku; j++)
            {
                printf("%4d", i*j);
            }

            printf("\n");
        }

        return 0;
    }

```

5.5

Tutki, mitä seuraava C-ohjelma tekee. Missä suhteessa se on hyvien C-ohjelmointitapojen vastainen?

```

#include <stdio.h>
#define MAX 2800
long int a = 10000, b, c = MAX, d, e, f[MAX+1],g;
int main(void)
{
    for (;b-c;) f[b++] = a/5;
    for (;d=0,g=c*2;c-=14, printf("%.4d",e+d/a),e=d%a)
        for(b=c;d+=f[b]*a, f[b]=d%--g, d/=g--, --b;d*=b);
    return 0;
}

```

Ratkaisu :

5.6

KOTITEHTÄVÄ : Tee ohjelma, joka käyttäjän antamat kolme lukua ja tutkii, voivatko ne olla kolmion sivujen pituudet (kolmiossa kahden sivun summa on aina pidempi kuin kolmas sivu, erityisesti kahden lyhimmän sivun summan on oltava suurempi kuin pisin). Jos voivat, tutkitaan lisäksi, onko kolmio tasakylkinen (kaksi sivua yhtä pitkiä), tasasivuinen (kaikki sivut yhtä pitkiä) tai suorakulmainen (pisimmän sivun neliö on kahden muun neliöiden summa). Päättele ja tulosta tieto kustakin ominaisuudesta omassa funktiossaan.

Ratkaisu :

```

#include <stdio.h>
#include <math.h>
#define EPSILON 0.0000001

double pienin(double x1,double x2,double x3);
double keski(double x1,double x2,double x3);
double suurin(double x1,double x2,double x3);
int onko_kolmio(double a, double b, double c);
int onko_tasakylkinen(double a, double b, double c);
int onko_tasasivuinen(double a, double b, double c);
int onko_suorakulmainen(double a, double b, double c);

```

```
int onko_yhtasuuri(double a, double b);

int main(void)
{
    double a=0,b=0,c=0;      /* kolmion sivut */
    double x1,x2,x3; /* sivut suuruusjärjestyksessä */
    int tasas=0;
    printf("Anna kolmion sivut\n");
    while (a<=0) scanf("%lf",&a);
    while (b<=0)scanf("%lf",&b);
    while (c<=0)scanf("%lf",&c);

    x1=pienin(a,b,c);
    x2=keski(a,b,c);
    x3=suurin(a,b,c);

    if (onko_kolmio(x1,x2,x3))
    {
        if (onko_tasakylkinen(x1,x2,x3))
            tasas=onko_tasasivuinen(x1,x2,x3);
        if (!tasas)
            onko_suorakulmainen(x1,x2,x3);
    }
    return 0;
}

int onko_tasakylkinen(double a, double b, double c)
{ /* funktioon tultaessa a <= b <= c */
    int ret_val;
    if(onko_yhtasuuri(a,b) ||onko_yhtasuuri(b,c))
    {
        printf("Kolmio on tasakylkinen.\n");
        ret_val=1;
    }
    else ret_val=0;
    return ret_val;
}

int onko_tasasivuinen(double a, double b, double c)
{ /* funktioon tultaessa a <= b <= c */
    int ret_val;
    if(onko_yhtasuuri(a,c))
    {
        printf("Kolmio on tasasivuinen.\n");
        ret_val=1;
    }
    else ret_val=0;
    return ret_val;
}

int onko_suorakulmainen(double a, double b, double c)
{ /* funktioon tultaessa a <= b <= c */
    int ret_val;
    if(onko_yhtasuuri(a*a+b*b,c*c))
    {
        printf("Kolmio on suorakulmainen.\n");
        ret_val=1;
    }
}
```

```
        else ret_val=0;
        return ret_val;
    }
    int onko_kolmio(double a, double b, double c)
    { /* funktioon tultaessa a <= b <= c */
        int ret_val;
        if(a+b>c)
        {
            printf("Kyseess\204 on kolmio.\n");
            ret_val=1;
        }
        else
        {
            printf("Annetut sivut eiv\204t voi muodostaa kolmiota.");
            ret_val=0;
        }
        return ret_val;
    }

    int onko_yhtasuuri(double a, double b)
    {
        int ret_val;
        if (fabs(a-b) < EPSILON) ret_val=1;
        else ret_val=0;
        return ret_val;
    }

    double suurin(double x1,double x2,double x3)
    {
        double pal_arvo=x1;
        if(x2>pal_arvo)
            pal_arvo=x2;
        if(x3>pal_arvo)
            pal_arvo=x3;
        return pal_arvo;
    }

    double pienin(double x1,double x2,double x3)
    {
        double pal_arvo=x1;
        if(x2<pal_arvo)
            pal_arvo=x2;
        if(x3<pal_arvo)
            pal_arvo=x3;
        return pal_arvo;
    }

    double keski(double x1,double x2,double x3)
    {
        double pal_arvo;
        double suur=suurin(x1,x2,x3);
        double pien=pienin(x1,x2,x3);

        pal_arvo=x3;
        if (x1 > pien && x1< suur) pal_arvo=x1;
```

```
    if (x2 > pien && x2< suur) pal_arvo=x2;
    return pal_arvo;
}
```

6 HARJOITUS

6.1

Tee ohjelma, joka lukee käyttäjän antamia kokonaislukuja niin kauan kuin käyttäjä antaa jonkin syöttämänsä luvun uudestaan. Voit olettaa, että luvut ovat välillä 0..100.

Ratkaisu :

```
#include <stdio.h>
int main(void){
    int luku, lopeta=0, taulukko[101]={0};

    while(lopeta!=1)
    {
        printf("Anna kokonaisluku!\n");
        scanf("%d",&luku);
        if(taulukko[luku]!=1)
            taulukko[luku]=1;
        else
            lopeta=1;
    }
    printf("Lukeminen lopetettu!\n");
    printf("\n\nKontrollitulostus/luetut luvut:\n");
    for (luku=0; luku<101; luku++) if (taulukko[luku]) printf("%3d ", luku);

    return 0;
}
```

6.2

Tee funktio, joka vaihtaa ("muuttuja-") parametrina saamiensa kolmen kokonaisluvun arvot niin, että ensimmäisenä parametrina olevaan muuttujaan tulee arvoista pienin, toiseen keskimäinen ja kolmanteen arvoista suurin.

Ratkaisu :

```
#include <stdio.h>

void vaihda(int *a, int *b, int *c);

int main(void)
{
    int luku1, luku2, luku3;

    printf("Anna kolme lukua.\n");
    scanf ("%i", &luku1);
    scanf ("%i", &luku2);
    scanf ("%i", &luku3);

    printf("Luvut alkuper\204isess\204 j\204rjestyksess\204:\n");
    printf("%d, %d, %d\n", luku1, luku2, luku3);

    vaihda(&luku1, &luku2, &luku3);
    printf("Luvut suuruusj\204rjestyksess\204:\n");
}
```

```
        printf("%d, %d, %d\n", luku1, luku2, luku3);

        return 0;
    }

    void vaihda(int *a, int *b, int *c)
    {
        int min, max, med;

        if ((*a >= *b) && (*b >= *c)) {max = *a; med=*b;min=*c;}
        else if ((*a >= *c) && (*c >= *b)) {max = *a; med=*c;min=*b;}
        else if ((*b >= *a) && (*a >= *c)) {max = *b; med=*a;min=*c;}
        else if ((*b >= *c) && (*c >= *a)) {max = *b; med=*c;min=*a;}
        else if ((*c >= *a) && (*a >= *b)) {max = *c; med=*a;min=*b;}
        else /* ((*c >= *b) && (*b >= *a)) */ {max = *c; med=*b;min=*a;};
        *a = min;
        *b = med;
        *c = max;
    }
```

6.3

Tee funktio, joka vaihtaa parametrina saamansa kokonaislukutaulukon luvut päinvastaiseen järjestykseen. Toisena parametrina välitetään taulukon alkioden lukumäärä.

Ratkaisu :

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
void kaannos(int *taulukko,int n);
```

```
int main(void)
```

```
{
```

```
    int i,k;
```

```
    char luku[128];
```

```
    int taulukko[128];
```

```
    clrscr();
```

```
    printf("\nSyota lukujen maara (oltava alle 128): ");
```

```
    gets(luku);
```

```
    sscanf(luku,"%d",&i);
```

```
    for(k=0;k<i;k++) /* Kerataan syotetyt luvut taulukkoon */
```

```
    {
```

```
        printf("\nSyota luku nro. %d taulukkoon: ",k+1);
```

```
        gets(luku);
```

```
        sscanf(luku,"%d",&taulukko[k]);
```

```
    }
```

```
    for(k=0;k<i;k++) /* Naytetaan syotetyt taulukon arvot */
```

```
    {
```

```
        printf("\nTaulukon alkuperainen luku nro. %d on:
%d\n",k+1,taulukko[k]);
```

```
    }
```

```
    kaannos(taulukko,i);
```

```
        for(k=0;k<i;k++)      /* Naytetaan syötetyt taulukon arvot */
        {
            printf("\nMuutetun taulukon luku nro. %d on:
%d\n",k+1,taulukko[k]);
        }

        return 0;
}

void kaannos(int *taulukko,int n)
{
    int apu;
    int i;

    for (i=0;i<n/2;i++)
    {
        apu = taulukko[n-i-1];
        taulukko[n-i-1] = taulukko[i];
        taulukko[i] = apu;
    }
}
```

7 HARJOITUS

7.1

Kirjoita ohjelma, joka tulostaa kokonaislukuarvoja vastaavat ASCII-merkit.

Ratkaisu :

```
#include <stdio.h>

int main(void){
    int merkki = 0;
    while(merkki<=255){
        printf(" %3d %3o %2X %c\n",merkki,merkki,merkki,merkki);
        merkki++;
    }
    return 0;
}
```

7.2

Kirjoita funktio, joka saa parametrinaan kokonaislukuarvon ja palauttaa vastaavan ASCII-merkin sekä funktio, joka saa syötteenään merkin ja palauttaa vastaavan kokonaislukuarvon.

7.3

Kirjoita edellisen tehtävän funktioiden avulla ohjelma, jonka avulla käyttäjä voi tutkia mikä merkki vastaa annettua numeerista koodia ja mikä numeerinen koodi vastaa annettua merkkiä.

Ratkaisu (7.2 ja 7.3) :

```
#include <stdio.h>

int CharToCode(unsigned char c);
char CodeToChar(int code);

int main(void)
{
    char puskuri[128];
    char c;
    int luku;
    int valinta=3;

    while (valinta){
        printf("\n\nValitse seuraavista:\n");
        printf("\t1: Haetaan antamaasi lukuarvoa vastaava merkki.\n");
        printf("\t2: Haetaan antamaasi merkki\204 vastaava lukuarvo.\n");
        printf("\t0: Lopetus.\n\n");

        printf("Anna valintasi.\n");
        fgets(puskuri, 128, stdin);
        sscanf(puskuri, "%d", &valinta);

        switch(valinta){

            case 1:    printf("Anna luku: ");
                      fgets(puskuri, 128, stdin);
                      sscanf(puskuri, "%d", &luku);

                      c = CodeToChar(luku);

                      printf("Antamasi luku vastaa merkki\204: %c\n", c);

                      break;

            case 2:    printf("Anna merkki: ");
                      fgets(puskuri, 128, stdin);
                      sscanf(puskuri, "%c", &c);

                      luku = CharToCode(c);

                      printf("Antamasi merkin lukuarvo on: %d\n", luku);

                      break;

        }
    }
    return 0;
}
```

```
int CharToCode(unsigned char c)
{
    int code;
    code = c;
    return code;
}
```

```
char CodeToChar(int code)
{
    char c;
    c = code;
    return c;
}
```

7.4

Kirjoita funktio, joka etsii merkkijonosta tiettyä merkkiä.

Funktiolle annetaan parametreina merkkijono ja merkki. Funktio palauttaa arvonaan osoittimen merkkijonon ensimmäiseen sellaiseen merkkiin, jota etsitään. Ellei etsittävää merkkiä löydy, funktio palauttaa arvon NULL.

Kirjoita funktiolle myös käyttöympäristö eli pääohjelma, josta kutsut sitä.

Ratkaisu:

```
#include <stdio.h>
```

```
int main(void);
char *etsi(char *mista, char mita);
```

```
int main(void)
{
    char puskuri[128];
    char c;
    char *p;
```

```
    printf("Anna etsitt\204v\204 merkki: ");
    fgets(puskuri, 128, stdin);
    sscanf(puskuri, "%c", &c);
```

```
    printf("Anna merkkijono, josta merkki\204 etsit\204\204n:\n");
    fgets(puskuri, 128, stdin);
```

```
    if ((p=etsi(puskuri, c)) != NULL){
        printf("Merkkijono sis\204lsi merkin %c ", c);
        printf("kohdassa %d.\n",p-puskuri);
    }
    else printf("Merkkijonossa %s ei ole merkki\204 %c.\n",puskuri,c);
```

```
    return 0;
}
```

```
char *etsi(char *q, char merkki)
{
    char *p = NULL;

    while (*q != '\0'){
        if (*q == merkki) { p = q; break; }
        q++;
    }

    return p;
}
```

7.5

Kirjoita funktio, joka voi saada parametrikseen ison kirjaimen A-Z ja joka palauttaa arvonaan vastaavan pienen kirjaimen. Ellei parametri ole mikään kirjaimista A-Z, funktio palauttaa arvonaan parametrin muuttumattomana. Kirjoita myös testiohjelma.

Funktion kutsu on muotoa: `pieni= ch_pieneksi(iso);`

Kirjainten A-Z numeeriset ASCII-vastineet ovat 65-90 järjestyksessä yhden välein. Pienten kirjainten a-z numeeriset vastineet ovat 97-122.

Ratkaisu:

```
#include <stdio.h>

char ch_pieneksi(char c);

int main(void)
{
    char puskuri[128];
    char c;

    printf("Anna pieneksi muutettava kirjain: ");
    fgets(puskuri, 128, stdin);
    sscanf(puskuri, "%c", &c);

    printf("%c on pienen\204 kirjaimena: %c\n", c, ch_pieneksi(c));

    return 0;
}

char ch_pieneksi(char c)
{
    if ((c >= 65) && (c <= 90)) c += 32;
    return c;
}
```

7.6

Kirjoita ohjelma, joka tulostaa sille syötetyn merkkijonon pituuden. Toteuta ohjelma kahdella tavalla:

- käytä *strlen()*-funktia
- laske taulukossa olevan merkkijonon pituus (loppumerkinä `'\0'`). funktio, joka voi saada parametrikseen ison kirjaimen A-Z ja joka palauttaa arvonaan vastaavan pienen

Ratkaisu:

```
#include <stdio.h>

int strlen_oma(char* jono);

int main (void)
{
    int pituus1=0, pituus2=0;
    char jono[128];

    printf("Anna merkkijono\n");
    fgets(jono, 128, stdin);
    pituus1= strlen(jono);
    pituus2= strlen_oma(jono);
    printf("Pituus on funktion strlen mukaan %d ja itse laskettuna %d.\n",
        pituus1, pituus2);
    return 0;
}

int strlen_oma(char* jono){
    int pituus=0, i;
    for(i=0; jono[i]!='\0'; i++)
        pituus++;
    return pituus;
}
```

7.7

KOTITEHTÄVÄ: C:ssä merkkijonojen vertailu ei onnistu yhtäsuuruusvertailulla (*if (mjon1 == mjon2)*), koska merkkijonon nimi on merkkijonon osoite.

Tämän vuoksi vertailu on tehtävä funktiolla, joka saa parametreikseen kaksi merkkipointteria.

Funktiossa käydään jonot läpi merkki merkiltä, kunnes voidaan todeta, että toinen jonoista on suurempi tai että merkkijonot ovat samat.

- Kirjoita ohjelma, joka tutkii kahden syötetyn merkkijonon aakkosjärjestyksen. Käytä *strcmp()*-funktia.
- Kirjoita oma versiosi *strcmp()*-funktioista ja testaa se.

Ratkaisu:

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

int vertaa(char *a, char *b);
```

```
int main(void){
char rivi1[128],rivi2[128];

printf("Anna merkkijono 1!\n" );
fgets(rivi1,127,stdin);
printf("Anna merkkijono 2!\n" );
fgets(rivi2,127,stdin);

printf("Oman vertailun tulos on %d\n", vertaa(rivi1,rivi2));
printf("strcmp-vertailun tulos on %d\n", strcmp(rivi1,rivi2));

return 0;
}

int vertaa(char *p, char *q){
while (*p != '\0' && *p == *q){
p++;
q++;
}

return *p-*q;
}
```

8 HARJOITUS

8.1

Kirjoita ohjelma, joka tutkii satunnaislukugeneraattorin tuottamien satunnaislukujen hyvyyttä. Generaattorin pitäisi tuottaa kaikkia lukuja samalla todennäköisyydellä. Ohjelma pyytää käyttäjältä generoitavien satunnaislukujen määrän ja kokonaislukualueen (esim. 1 .. n), jolle lukuja generoidaan. Saatu jakauma talletetaan taulukkoon ja tulostetaan. Lisäksi ilmoitetaan suurimman ja pienimmän esiintymiskerran erotus.

Ratkaisu:

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

#define MAX_ALUE 100

int main(void)
{
int jakauma[MAX_ALUE] = {0};
int alaraja, ylaraja, lkm, i;
int suurin, pienin, sat;
char puskuri[128];

printf("\nMille v\204lille lukuja generoidaan?\n");
printf("Anna alaraja:");
fgets(puskuri,127,stdin);
sscanf(puskuri,"%d",&alaraja);

printf("Anna yl\204raja:");
```

```
fgets(puskuri,127,stdin);
sscanf(puskuri,"%d",&ylaraja);

printf("\nAnna arvottavien lukujen m\204\204r\204 ");
scanf("%d",&lkm);
srand(time(NULL));

/* Arvotaan satunnaisluvut */

for(i=0;i<lkm;i++)
{
    sat=rand()%(ylaraja-alaraja+1)+alaraja;
    jakauma[sat]++;
}

/* Etsitään maksimi ja minimi ja tulostetaan jakauma */
suurin=ylaraja;
pienin=alaraja;
for(i=alaraja;i<=ylaraja;i++)
{
    if(jakauma[i]<jakauma[pienin]) pienin=i;
    if(jakauma[i]>jakauma[suurin]) suurin=i;
    printf("%d: %d \n",i,jakauma[i]);
}
printf("\nOdotettu frekvenssi: %1.2f, suurin: %d, pienin: %d, erotus: %d",
       (float) lkm/(ylaraja-alaraja+1),jakauma[suurin],
       jakauma[pienin],jakauma[suurin]-jakauma[pienin]);
return 0;
}
```

8.2

Tee erilliset funktiot, jotka palauttavat parametrina välitetystä merkkijonosta tiedon, kuinka monta a) merkkiä, b) ei-tyhjää merkkiä, c) vokaalia, d) isoa kirjainta, e) numeroa, f) skandinaavista kirjainta ja g) eri kirjainta merkkijono sisältää.

Tee sitten pääohjelma, joka analysoi em. funktioilla näppäimistöltä annetusta merkkijonosta em. tiedot.

Ratkaisu:

```
#include <ctype.h>
#include <stdio.h>
#include <stdlib.h>

int laskeMerkit(unsigned char* jono);
int laskeEiTyhjat(unsigned char* jono);
int laskeVokaalit(unsigned char* jono);
int laskeIsot(unsigned char* jono);
int laskeNumerot(unsigned char* jono);
int laskeSkandit(unsigned char* jono);
int laskeEriKirjaimet(unsigned char* jono);

int main(void){
    unsigned char merkkiJono[128];
    int merkit,eiTyhjat,vokaalit,isot,numerot,skandit,eriKirjaimet;

    printf("Anna merkkijono\n");
    fgets(merkkiJono,127,stdin);
    merkit=laskeMerkit(merkkiJono);
```

```
eiTyhjat = laskeEiTyhjat(merkkiJono);
vokaalit = laskeVokaalit(merkkiJono);
isot = laskeIsot(merkkiJono);
numerot = laskeNumerot(merkkiJono);
skandit = laskeSkandit(merkkiJono);
eriKirjaimet = laskeEriKirjaimet(merkkiJono);

printf("Jonossa oli:\n");
printf("\t %d merkki\204\n",merkit);
printf("\t %d ei-tyhj\204\204 merkki\204\n",eiTyhjat);
printf("\t %d vokaalia\n",vokaalit);
printf("\t %d isoa kirjainta\n",isot);
printf("\t %d numeroa\n",numerot);
printf("\t %d skandinaavista kirjainta\n",skandit);
printf("\t %d eri kirjainta\n",eriKirjaimet);
return 0;

}

int laskeMerkit(unsigned char* jono){
    int lkm=0;
    while(jono[lkm]!='\0'){
        lkm++;
    }
    return lkm;
}

int laskeEiTyhjat(unsigned char* jono)
{
    int i=0,lkm=0;
    while(jono[i]!='\0'){
        if(jono[i]!=32) lkm++;
        i++;
    }
    return lkm;
}

int laskeVokaalit(unsigned char* jono)
{
    int i=0,lkm=0;
    while(jono[i]!='\0'){
        if(jono[i]=='a' || jono[i]=='e' || jono[i]=='i' || jono[i]=='o' ||
           jono[i]=='u' || jono[i]=='y' ||
           jono[i]==134 || jono[i]== 132 || jono[i]== 148 || /* å, ä ja ö */
           jono[i]=='A' || jono[i]=='E' || jono[i]=='I' || jono[i]=='O' ||
           jono[i]=='U' || jono[i]=='Y' ||
           jono[i]==143 || jono[i]==142 || jono[i]==153 ) /* Å, Ä ja Ö */
            lkm++;
        i++;
    }
    return lkm;
}

int laskeIsot(unsigned char* jono)
{
    int i=0,lkm=0;
    while(jono[i]!='\0'){
        if((jono[i]>64&& jono[i]<91) || jono[i]==143 || jono[i]==142 || jono[i]==153)
```

```
        lkm++;
    i++;
}
return lkm;
}

int laskeNumerot(unsigned char* jono)
{
    int i=0, lkm=0;
    while(jono[i]!='\0'){
        if(jono[i]>47&& jono[i]<58)
            lkm++;
        i++;
    }
    return lkm;
}

int laskeSkandit(unsigned char* jono)
{
    int i=0, lkm=0;
    while(jono[i]!='\0'){
        if(jono[i]==134 || jono[i]== 132 || jono[i]== 148 || /* å, ä ja ö */
           jono[i]==143 || jono[i]==142 || jono[i]==153 )
            lkm++;
        i++;
    }
    return lkm;
}

int laskeEriKirjaimet(unsigned char* jono){
    int taulu[255]={0}, lkm=0, i=0;
    while(jono[i]!='\0'){
        if(taulu[jono[i]]==0 && (isalpha(jono[i]) ||
                                jono[i]==134 || jono[i]== 132 || jono[i]== 148 || /* å, ä ja ö */
                                jono[i]==143 || jono[i]==142 || jono[i]==153))
            taulu[jono[i]]=1;
        i++;
    }
    i=0;
    while(i<256){
        if(taulu[i]==1)
            lkm++;
        i++;
    }
    return lkm;
}
```


8.3

Tee funktio, joka poistaa merkkijonosta kaikki turhat blankot eli poistaa kaikki alussa ja lopussa olevat sekä muuttaa keskellä olevat useamman blankon jonot yhdeksi blankoksi.

Ratkaisu:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MJP 1000 /* merkkijonon maksimipituus */

char* poistaLiiatBlankot(char* jono);

int main (void){
    char jono[MJP];
    printf("Anna tiivistett\204v\204 merkkijono\n");
    fgets(jono,MJP-1,stdin);
    printf("\nAlkuper\204inen merkkijono:\n");
    printf(jono);

    poistaLiiatBlankot(jono);

    printf("\nTiivistetty merkkijono\n");
    printf(jono);

    return 0;
}

char *poistaLiiatBlankot(char* jono)
{
    /* koko jono käydään läpi ja kirjoitetaan */
    /* uudestaan vanhan paikalle */
    char *p=jono; /* p osoittaa kirjoituskohdan */
    char *q=p; /* q osoittaa, mistä kirjoitetaan */

    while (*q == ' ') q++; /* selataan pois alussa olevat blankot */

    while (*q != '\0'){
        *p=*q;
        p++;
        if (*q == ' ') { /* selataan pois yhtä lukuisimmat blankot */
            q++;
            while (*q == ' ') q++;
        }
        else q++;
    }
    if (*(p-1)==' ') p++;
    *p = '\0';
    return jono;
}
```

8.4

Tee ohjelma, joka laskee näppäimistöltä annetun mielivaltaisen pitkän kokonaisluvun numeroiden summan. Lue syöte merkkijonona ja hylkää se, jos syötteessä esiintyy muuta kuin numeroita tai merkki '+', joka ilmoittaa, että luku jatkuu seuraavalle riville.

Ratkaisu:

```
#include <ctype.h>
#include <stdio.h>
#include <stdlib.h>

int main(void){
    int summa=0;
    char rivi[128];
    char *p;
    int apu;
    int jatka=1;
    int virhe=0;
    printf("Tämä ohjelma laskee antamasi mielivaltaisen pitkän kokonaisluvun numeroiden summan. Voit kirjoittaa riville\n");
    printf("enintään yksi merkki, mutta voit jatkaa riviä\n");
    printf("kirjoittamalla riville viimeiseksi merkiksi +-merkin. \n\n");
    printf("Anna numerojono!\n");

    while (jatka){
        jatka=0;
        fgets(rivi,127,stdin);
        p=rivi;
        while (isdigit(*p)){
            sscanf(p++,"%i",&apu);
            summa+=apu;
        }
        if (*p=='+') jatka=1;
        else if (*p!='\0' && *p!='\n') virhe=1;
    }
    if (!virhe) printf("Antamasi kokonaisluvun numeroiden summa oli %d\n",summa);
    else printf("Antamassasi luvussa oli virheellisiä merkkejä\n");

    return 0;
}
```

8.5

Kirjoita funktio, joka lisää merkkijonossa osoitettuun paikkaan toisen merkkijonon. Funktion prototyyppi on muotoa

char * insert(char *mihin, const char *lisa, int kohta);

Osoitin **mihin** kertoo sen jonon alun, johon lisätään, **lisa** osoittaa lisättävään merkkijonoon ja **kohta** kertoo kohdemerkkijonon indeksin, josta alkaen lisättävän tekstin tulee alkaa.

Ratkaisu:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define KOKO 1000
```

```
char *insert(char *mihin, const char *lisa, int kohta);

int main(void)
{ char mihin[KOKO],mista[KOKO];
  int paikka=0;
  printf ("Anna merkkijono\n");
  gets(mihin);
  printf ("Anna lisättävä merkkijono\n");
  gets(mista);
  printf("Anna paikka, johon lisätään.");
  scanf("%d",&paikka);
  if (strlen(mihin)+strlen(mista) > KOKO || paikka+strlen(mista)>KOKO) {
    printf ("Tila ei riitä, käytä vähän lyhyempiä merkkijonoja\n"); }
  else {
    insert(mihin,mista, paikka);
    printf(mihin);
  }
  return 0;
}

char * insert(char *mihin, const char *mista, int kohta){
char  apu[KOKO];
  strcpy(apu,mihin+kohta);
  mihin[kohta]='\0';
  strcat(mihin,mista);
  strcat(mihin,apu);
  return mihin;
}
```

8.6

KOTITEHTÄVÄ Oletetaan, että merkkijonomuuttujassa on nimitiedot niin, että ensin on kaikki etunimet ja sitten sukunimi, kaikki pienillä kirjaimilla. Kirjoita funktio, joka muuttaa alkukirjaimet isoiksi. Funktion prototyyppi on muotoa

char *alku_kirjaimet_isoiksi(char *s)

missä **s** osoittaa alkuperäisen merkkijonon alkuun ja funktion palauttama arvo osoittaa muunnetun merkkijonon alkuun.

Esimerkkejä muunnoksista:

kaarlo viljami möttönen -> Kaarlo Viljami Möttönen

gunhild pula-aho -> Gunhild Pula-Aho

minni hirvelö kukka-maaria örkensgård-mähniö -> Minni Hirvelö Kukka-Maaria Örkensgård-Mähniö

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

#define MJONON_MAX_PITUUS 128

char *alku_kirjaimet_isoiksi(char *s);
char isoksi(char a);

int main(void)
{ char nimi[MJONON_MAX_PITUUS];
```

```

        gets(nimi);
        alku_kirjaimet_isoiksi(nimi);
        printf(nimi);
        return 0;
    }

char *alku_kirjaimet_isoiksi(char *s)
{
    int i;
    int pit=strlen(s);
    if (s[0]!=' ') s[0] = isoksi(s[0]);
    for( i=1;i<pit;i++)
        if (s[i-1]==' ' || s[i-1]=='-') s[i]=isoksi(s[i]);

    return s;
}

char isoksi(char a){
    char ret_val=a;
    switch(a) {
        case 134: ret_val= 143; break; /* å->Å */
        case 132: ret_val= 142; break; /* ä->Ä */
        case 148: ret_val= 153; break; /* ö->Ö */
        default: ret_val=toupper(a);
    }
    return ret_val;
}

```

9 HARJOITUS

9.1

Määrittele tietuerakenne, jossa on kaksi kokonaislukukenttää. Tee sitten 50 alkion tietueaulukko näistä tietueista ja täytä kentät satunnaisluvuilla. Tulosta sitten taulukosta sen tietueen tiedot, jossa kenttien summa on suurin.

Ratkaisu:

```

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

typedef struct {
    int luku1, luku2;
} Lukupari;

int main(void){
    Lukupari taulu[50], suurin;
    int i=0;

    srand(time(NULL));

    while(i<50){
        taulu[i].luku1=rand();
        taulu[i].luku2=rand();
        i++;
    }
}

```

```
    suurin=taulu[0];
    for(i=1;i<50;i++){
        if((suurin.luku1+suurin.luku2)<(taulu[i].luku1+taulu[i].luku2))
            suurin=taulu[i];
    }

    printf("Suurin pari oli %d %d\n",suurin.luku1,suurin.luku2);
    return 0;
}
```

9.2

Kirjarekisteriä varten halutaan kirjasta tallettaa seuraavat tiedot:

- kirjan nimi
- tekijä
- kustantaja
- painovuosi
- sivumäärä
- hinta
- onko kuvitusta
- ISBN-numero

Tee sopiva C-kielinen tietuemäärittely. Valitse kenttien tietuetyypit huolella. Yritä arvioida, millaisia operaatioita niihin tullaan kohdistamaan.

9.3

Tee ohjelma, joka lukee edellisen tehtävän tietuerakenteen mukaista tietoa ja siirtää tiedot taulukkoon, jonka alkiotyyppinä on ko. tietuetyyppi. Lopuksi taulukon sisältö tulostetaan.

Ratkaisu (9.2 ja 9.3):

```
#include <stdio.h>
#include <stdlib.h>
#define MKT_KOKO 100
typedef struct{
    char nimi[80];
    char tekija[80];
    char kustantaja[80];
    char painovuosi[4];
    int sivumaara;
    float hinta;
    int kuvitusta;
    char ISBN[13];
} Kirja;

int main (void){
    Kirja taulu[MKT_KOKO];
    int lopetus=0,i=0,maara=0;
    char apu[128];

    while(lopetus!=1){
        printf("Anna kirjan nimi\n");
        gets(taulu[i].nimi);
```

```

printf("Anna kirjan tekij\204\n");
gets(taulu[i].tekija);

printf("Anna kirjan kustanja\n");
gets(taulu[i].kustantaja);

printf("Anna kirjan painovuosi(yyyy)\n");
gets(taulu[i].painovuosi);

printf("Anna kirjan sivum\204\204r\204\n");
fgets(apu,127,stdin);
taulu[i].sivumaara=atoi(apu);

printf("Anna kirjan hinta\n");
fgets(apu,127,stdin);
taulu[i].hinta=atof(apu);

printf("Onko kirjassa kuvitus (1-on 0-ei)\n");
fgets(apu,127,stdin);
taulu[i].kuvitusta=atoi(apu);

printf("Anna kirjan ISBN-numero\n");
gets(taulu[i].ISBN);

printf("Lopetetaanko (0 jatkaa)\n");
fgets(apu,127,stdin);
lopetus=atoi(apu);
maara++;
i++;
}

for(i=0;i<maara;i++){

printf("\n\nKirja:%s \nTekij\204:%s \nKustantaja:%s \n\tpainovuosi:%s\n",
    taulu[i].nimi,taulu[i].tekija,taulu[i].kustantaja,taulu[i].painovuosi);

printf("\tsivum\204\204r\204:%d \n\thinta:%.2f \n\tkuvitusta:%d \n\ISBN:%s",
    taulu[i].sivumaara,taulu[i].hinta,taulu[i].kuvitusta,taulu[i].ISBN);
}

return 0;
}

```

9.4

KOTITEHTÄVÄ: Varastonvalvontajärjestelmää varten talletetaan jokaisesta tuotteesta seuraavat tiedot: tuotenumero, tuotteen nimi, valmistajan nimi ja osoite, toimittajan nimi ja osoite, tuotteen yksikköhinnat (hankintahinta ja ulosmyyntihinta) ja lukumäärä varastossa.

Suunnittele sopiva tietuemäärittely. Varaston tila voidaan kuvata tätä tietuetyyppiä olevalla taulukolla. Tee ohjelma, joka laskee ja tulostaa varaston arvon. Lisäksi tuotetaan pyydettyä lista tuotteista, joita on varastossa alle 10 kpl, ja näiden tuotteiden toimittajat osoitetietoineen.

Datan tietorakenteeseen voi muodostaa syöttämällä käsin tai sopivasti arvoltaan rajoitetuilla satunnaisluvuilla.

Ratkaisu:

```
#include <stdio.h>
```

```
#include <stdlib.h>
#define MAX_TUOTTEIDEN_LKM 2

typedef struct{
    char tuotetunnus[10];
    char nimi[80];
    char valmistaja[80];
    char valmOsoite[80];
    char toimittaja[80];
    char toimOsoite[80];
    float hankHinta;
    float myyntiHinta;
    int lkmVarastossa;
} Tuote;

int lisaaTuote(Tuote *taulu, int i);
void laskeVarastonArvo(Tuote *s, int lkm);
void vahanVarastossa(Tuote *s,int lkm,int halytysRaja);

int main(void)
{
    int lkm=0;
    Tuote varasto[MAX_TUOTTEIDEN_LKM];
    int valinta;
    char puskuri[128];

    do{
        printf("\n1. Lis\204\204 tuotteita");
        printf("\n2. Tulosta varaston arvo");
        printf("\n3. Hae tuotteet, joita varastossa alle tilausrajan");
        printf("\n0. Lopetus");
        printf("\n\nAnna valintasi:");
        gets(puskuri);
        sscanf(puskuri,"%i",&valinta);

        switch (valinta){
            case 1:
                if (lkm < MAX_TUOTTEIDEN_LKM)
                    lkm=lisaaTuote(varasto,lkm);
                else {
                    printf("\n\n *** Varasto on t\204nn\204!");
                    printf("\n *** Uusia tuotteita ei voi sy\224tt\204\204.\n\n");
                }
                break;
            case 2:
                laskeVarastonArvo(varasto, lkm);
                break;
            case 3:
                vahanVarastossa(varasto,lkm,10);
                break;
        }
    } while (valinta!=0);

    return 0;
}

void laskeVarastonArvo(Tuote *s, int lkm){
```

```
float arvoH=0, arvoU=0;
int i;

for(i=0;i<lkM;i++){
    arvoH += s[i].hankHinta*s[i].lkmVarastossa;
    arvoU += s[i].myyntiHinta*s[i].lkmVarastossa;
}
printf("\n\nVaraston arvo on\n\thankintahinnan mukaan: %2.2f EUR\n", arvoH);
printf("\tmyyntihinnan mukaan: %2.2f EUR\n\n", arvoU);
return ;
}

int lisaaTuote(Tuote *taulu, int i) {
    char apu[10];
    printf("Anna tuotteen tunnus\n");
    gets(taulu[i].tuotetunnus);

    printf("Anna tuotteen nimi\n");
    gets(taulu[i].nimi);

    printf("Anna tuotteen valmistajan nimi\n");
    gets(taulu[i].valmistaja);
    printf("Anna tuotteen valmistajan osoite\n");
    gets(taulu[i].valmOsoite);

    printf("Anna tuotteen toimittajan nimi\n");
    gets(taulu[i].toimittaja);
    printf("Anna tuotteen toimittajan osoite\n");
    gets(taulu[i].toimOsoite);

    printf("Anna tuotteen hankintahinta\n");
    gets(apu);
    taulu[i].hankHinta=atof(apu);
    printf("Anna tuotteen myyntihinta\n");
    gets(apu);
    taulu[i].myyntiHinta=atof(apu);

    printf("Anna tuotteen varastossa oleva m\204\204r\204\n");
    gets(apu);
    taulu[i].lkmVarastossa=atoi(apu);

    return ++i;
}

void vahanVarastossa(Tuote *s,int lkm,int halytysRaja){
int i;
system("cls");
printf("\n\nSeuraavia tuotteita on alle h\204lytysrajan %d\n",halytysRaja);
for(i=0;i<lkM;i++){
if(s[i].lkmVarastossa < halytysRaja)
    printf("\nTuotteen tunnus:\t%s \tnimeke:%s",
        s[i].tuotetunnus,s[i].nimi);
    printf("\nToimittaja:\t%s \t osoite:%s \n\n\n",
        s[i].toimittaja,s[i].toimOsoite);
}
}
```


}

10 HARJOITUS

10.1

Kirjoita ohjelma, joka tutkii onko syötteenä annettu merkkijono palindromi ts. sama luettuna alusta loppuun ja alusta loppuun, kun välilyönnit jätetään huomiotta. Ohjelma tulostaa saamansa syötteen ja takaperin käännetyt merkkijonon sekä tiedon siitä, onko kyseessä palindromi. Testiaineistona voi käyttää seuraavia : ”saippuakauppias”, ”innostunut sonni”, ”nisumaa oli isäsi ilo aamuisin”.

Ratkaisu:

```
# #include <stdio.h>
#include <ctype.h>
#include <string.h>
#define MAX_RIVIN_PITUUS 128

void poistaBlankot(char *s);
int onkoPalindromi(char *s);
char isoksi(char a);

int main(void){
    int i;
    char buf[MAX_RIVIN_PITUUS+1];
    char apu[MAX_RIVIN_PITUUS+1];

    printf("Anna palindromiksi ep\204ilty merkkijono: \n");
    fgets(buf, MAX_RIVIN_PITUUS, stdin);
    fflush(stdin);
    if (buf[strlen(buf)-1] == '\n' ) buf[strlen(buf)-1] = '\0'; /* muutetaan
                                                                    rivinvaihto
                                                                    loppumerkiksi
                                                                    */

    strcpy(apu, buf);
    poistaBlankot(apu);

    printf("\n\nAntamasi merkkijono\n\n\t%s \n\n", buf);
    if(onkoPalindromi(apu))
        printf("on ");
    else printf("ei ole ");
    printf("palindromi.\n");

    printf("\n\nMerkkijono on k\204\204nnettyn\204:\n\n\t");
    for (i=strlen(buf); i>=0; i--) printf("%c", buf[i]);
    printf("\n");

    return 0;
}

void poistaBlankot(char *s){
    int i, j;
    for(i=0; s[i]!='\0'; i++)
    {
        if( isspace(s[i]) )
        {
```

```

        for(j=i; s[j]!='\0'; j++)
            s[j] = s[j+1];
    }
}

int onkoPalindromi (char *s){

    int i, j;
    int vastaus = 1;

    for(i=0, j=strlen(s)-1; i<strlen(s); i++, j--)
    {
        if(isoksi(s[i]) != isoksi(s[j]))
        {
            vastaus = 0;
            break;
        }
    }
    return vastaus;
}

char isoksi(char a){
    char ret_val=a;
    switch(ret_val) {
        case 134: ret_val= 143;
                 break; /* å->Ä */

        case 132: ret_val= 142;
                 break; /* ä->Ä */

        case 148: ret_val= 153;
                 break; /* ö->Ö */

        default: ret_val=toupper(a);
    }
    return ret_val;
}

```

10.2

Tee ohjelma, joka laskee tekstitiedostoon tallennetuista henkilöiden tiedoista keskiarvot, keskihajonnat sekä minimi ja maksimit. Tiedot tulostetaan kuvaruudulle.

Syötetiedostossa on kullakin rivillä henkilön nimi, ikä, pituus ja paino, esim.

Leopold Lässärä	35	163	97
Minni Mäkkyläpuro	27	185	62
Åke-Öjvind Äkämänder	21	178	78
Talla Puikina	24	172	72
..			

Syötetiedoston voi muodostaa käsin vaikkapa Notepadilla. Tietueiden (rivien) formaatin voit suunnitella itse.

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
#define MAX_LINE_LENGTH 256

int main(int argc, char *argv[] ){

    FILE *f1;
    char p[MAX_LINE_LENGTH+1];
    int x[3];
    int min_x[3]={1000000,1000000,1000000};
    int max_x[3]={-1,-1,-1};
    double sum[3]={0};
    double sum2[3]={0};
    double k_arvo[3];
    double k_haj[3];
    int lkm=0;
    int i;

    if (argc !=2) return (-1);

    if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Sy\224tt\224tiedostoa ei voi avata ...\n"); return -1; }

    while ( (fgets(p, MAX_LINE_LENGTH, f1)) != NULL ){
        lkm++;
        sscanf(p,"%s %s %d %d %d",&x[0],&x[1],&x[2]);
        for (i=0;i<3;i++){
            sum[i] +=x[i];
            sum2[i]+=x[i]*x[i];

            if (lkm==1){
                min_x[i]=x[i];
                max_x[i]=x[i];
            }
            else {
                if (min_x[i] > x[i]) min_x[i]=x[i];
                if (max_x[i] < x[i]) max_x[i]=x[i];
            }
        }
    }
    for (i=0;i<3;i++){
        k_arvo[i] = sum[i]/lkm;
        k_haj[i] = sqrt((sum2[i]-sum[i]*sum[i]/lkm)/(lkm-1));
    }
    system("cls");
    printf("Aineiston %s tunnuslukea:\n\n", argv[1]);
    printf("\t\ttik\204\tpituus\tpaino\n");
    printf("keskiarvo\t%1.2f\t%1.2f\t%1.2f\n",k_arvo[0],k_arvo[1],k_arvo[2]);
    printf("keskihajonta\t%1.2f\t%1.2f\t%1.2f\n",k_haj[0],k_haj[1],k_haj[2]);
    printf("minimi\t%d\t%d\t%d\n",min_x[0],min_x[1],min_x[2]);
    printf("maksimi\t\t%d\t%d\t%d\n",max_x[0],max_x[1],max_x[2]);

    fclose(f1);
```

```
    return 0;
}
```

10.3

Kirjoita ohjelma, joka lukee edellisen tehtävän tiedostoa, laskee kullekin henkilölle painoindeksin ja kirjoittaa alkuperäiset tiedot ja painoindeksin uuteen tiedostoon. Painoindeksi saadaan jakamalla paino (kg) pituuden (m) neliöllä.

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAX_LINE_LENGTH 256

int main(int argc, char *argv[] ){

    FILE *f1,*f2;
    char p[MAX_LINE_LENGTH+1];
    int x[3];
    double p_indeksi;
    char nimi1[MAX_LINE_LENGTH/2];
    char nimi2[MAX_LINE_LENGTH/2];
    int lkm=0;

    if (argc !=3) return (-1);

    if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Sy\224tt\224tiedostoa ei voi avata ...\n"); return -1; }

    if ((f2 = fopen(argv[2], "w")) == NULL)
    { printf("Tulostiedostoa ei voi avata..\n"); return -1; }

    while ( (fgets(p, MAX_LINE_LENGTH, f1)) != NULL ){
        lkm++;
        sscanf(p,"%s %s %d %d %d",nimi1,nimi2,&x[0],&x[1],&x[2]);
        p_indeksi=10000.0*x[2]/x[1]/x[1];
        fprintf(f2,"%s %s %d %d %d %1.2f\n",nimi1,nimi2,x[0],x[1],x[2],p_indeksi);
    }

    printf("\nK\204siteltiin %d tietuetta ",lkm);

    fclose(f1);
    fclose(f2);

    return 0;
}
```

10.4

KOTITEHTÄVÄ: Tee erilliset funktiot, jotka palauttavat parametrina välitetystä merkkijonosta tiedon, kuinka monta a) merkkiä, b) ei-tyhjää merkkiä, c) vokaalia, d) isoa kirjainta, e) numeroa, f) skandinaavista kirjainta ja g) eri kirjainta merkkijono sisältää. Kirjoita myös pääohjelma, joka analysoi näillä funktioilla annetusta tekstitiedostosta em. tiedot.

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

#define STR_PITUUS 256
char loydetyt[64];

int count_chars(char *);
int count_nonblanks(char *);
int count_vowels(char *);
int count_upper(char *);
int count_digits(char *);
int count_scandinavian(char *);
int count_diff_chars(char *);

typedef struct {
    int n_chars;
    int n_nonblank;
    int n_vowels;
    int n_upper;
    int n_digits;
    int n_scand;
    int n_diff_chars;
} Summary;

int main(int argc, char * argv[])
{
    Summary tl={0,0,0,0,0,0,0};
    char rivi[STR_PITUUS];
    FILE *tdsto;
    tdsto=fopen(argv[1],"r");

    while(fgets(rivi,STR_PITUUS,tdsto))
    {
        tl.n_chars+=count_chars(rivi);
        tl.n_nonblank+=count_nonblanks(rivi);
        tl.n_vowels+=count_vowels(rivi);
        tl.n_upper+=count_upper(rivi);
        tl.n_digits+=count_digits(rivi);
        tl.n_scand+=count_scandinavian(rivi);
        tl.n_diff_chars+=count_diff_chars(rivi);
    }
    printf("\n\nAnalyysi tiedostosta: %s \n",argv[1]);
    printf("=====\n");
    printf("\nMerkkej\204: \t\t%8d",tl.n_chars);
    printf("\nEi-tyhji\204 merkkej\204: \t%8d",tl.n_nonblank);
    printf("\nVokaaleja: \t\t%8d",tl.n_vowels);
    printf("\nIsoja kirjaimia: \t%8d",tl.n_upper);
    printf("\nNumeroita: \t\t%8d",tl.n_digits);
    printf("\nSkandinaavisia merkkej\204:%8d",tl.n_scand);
    printf("\nEri kirjaimia: \t\t%8d",tl.n_diff_chars);
    return 0;
}

int count_chars(char *str)
{
    char *i;
```

```
    int tulos=0;
    for(i=str;*i;i++) if(isprint(*i)) tulos++;
    return tulos;
}

int count_nonblanks(char *str)
{
    char *i;
    int tulos=0;
    for(i=str;*i;i++) if(isgraph(*i)) tulos++;
    return tulos;
}

int count_vowels(char *str)
{
    char *i;
    int tulos=0;
    for(i=str;*i;i++) if(strchr("aeiouyääöAEIOUYÄÄÖ Ž™",*i)) tulos++;
    return tulos;
}

int count_upper(char *str)
{
    char *i;
    int tulos=0;
    for(i=str;*i;i++) if(strchr("ABCDEFGHJKLMNOPQRSTUVWXYZÄÄÖ Ž™",*i)) tulos++;
    return tulos;
}

int count_digits(char *str)
{
    char *i;
    int tulos=0;
    for(i=str;*i;i++) if(isdigit(*i)) tulos++;
    return tulos;
}

int count_scandinavian(char *str)
{
    char *i;
    int tulos=0;
    for(i=str;*i;i++) if(strchr("ääöÄÄÖ Ž™†„",*i)) tulos++;
    return tulos;
}

int count_diff_chars(char *str)
{
    int pointer=0;
    char *i;
    for(i=str;*i;i++)
        if((isalpha(*i)||strchr("ääöÄÄÖ Ž™†„",*i))&&!strchr(loydetyt,*i))
        {
            loydetyt[pointer]=*i;
            pointer++;
            loydetyt[pointer]='\0';
        }
}
```

```
    return pointer;
}
```

11 HARJOITUS

Kaikki tämänkertaiset tehtävät käsitellään kotitehtävinä seuraavalla kerralla ja niistä voi jokaisesta saada yhden lisäpisteen.

11.1

Lasten käyttämät kontinkieli ja vede-kieli perustuvat siihen, että salakielistettävän tekstin sanan ja koodisanan (kontti tai vede) kaksi ensimmäistä kirjainta vaihdetaan keskenään ja sanat lausutaan (ja kirjoitetaan) yhteen. Esim. ”Mitä kuuluu?” on kontinkielellä ”Kotämintti kouluukuntti” ja vede-kielellä ”Midevetä kudeveuluu?”.

Kirjoita ohjelma, joka kääntää annetun tekstitiedoston tällaiselle salakielelle haluttua avainsanaa käyttäen. Tulos kirjoitetaan uuteen tiedostoon.

Ohje: kannattanee kirjoittaa funktiot, joista toinen eristää syötteestä sanoja ja toinen, joka saa argumentteinaan osoittimen muunnettavaan sanaan ja ”koodiavaimeen” ja palauttaa arvonaan osoittimen muunnettuun sanaan (myös parametrina annettu sana muutetaan).

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
char *vaihdAAlut(char *sana, char *koodi, char etuVaiTaka);

int main(int argc, char *argv[] ){

    FILE *f1, *f2;
    char puskuri[128];
    char koodisana[128];
    char apu[130];
    char *p;
    char *q;
    char etuVaiTaka;
    int rVaihto=0;

    if (argc != 3){
        system("cls");
        printf("\n\nOhjelman kutsu: %s infile outfile \n\n",argv[0]);
        return(-1);
    }

    if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Syöttötiedostoa ei voi avata ...\n"); return -1; }

    if ((f2 = fopen(argv[2], "w")) == NULL)
    { printf("Tulostiedostoa ei voi avata..\n"); return -1; }

    do{
        printf("Anna koodauksen avainsana: ");
```

```

    fgets(puskuri, 127, stdin);
    sscanf(puskuri, "%s", &koodisana);
} while (strlen(koodisana) < 4);

do {
    printf("Operoidaanko edest\204 (standardi-vede) \nvai takaa (standardi-
kontti) (e/t): ");
    fgets(puskuri, 127, stdin);
    sscanf(puskuri, "%c", &etuVaiTaka);
} while (etuVaiTaka != 'e' && etuVaiTaka != 't');

while ( (fgets(puskuri, 127, f1)) != NULL )
{
    p = puskuuri;
    if((q=strtok(p, " ,.")) != NULL)
    do {
        strcpy(apu,q);
        q=apu;
        if ( *(q+strlen(q)-1) =='\n'){
            *(q+strlen(q)-1)=' ';
            rVaihto=1;
        }
        vaihdaAlut(q, koodisana, etuVaiTaka);
        fprintf(f2,"%s ",q);
        if (rVaihto){
            fprintf(f2,"\n");
            rVaihto=0;
        }
    } while ((q=strtok(NULL, " ,.")) != NULL) ;
}

fclose(f1);
fclose(f2);

return 0;
}

char *vaihdaAlut(char *sana, char *koodi, char etuVaiTaka){
    char apu[130]={'\0'};
    char apu2[130]={'\0'};
    switch (etuVaiTaka){
        default:
        case 'e': apu[0]=sana[0];
                  apu[1]=sana[1];
                  apu[2]='\0';
                  strcpy(apu2, sana);
                  strcpy(sana, strcat(strncat(strcat(apu, koodi+2),
                  koodi, 2), apu2+2));

                  break;
        case 't': apu[0]=koodi[0];
                  apu[1]=koodi[1];
                  apu[2]='\0';
                  strcpy(apu2, sana);
                  strcpy(sana, strcat(strncat(strcat(apu, apu2+2), apu2, 2), koodi+2));
                  break;
    }
    return sana;
}

```


}

11.2

Tee ohjelma, joka poistaa tekstitiedostosta täsmälleen samanlaisina esiintyvien peräkkäisten rivien kopiot. Jos siis peräkkäin on kaksi tai useampia täsmälleen samanlaisia rivejä, uuteen tiedostoon niitä tulee vain yksi kappale.

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define MAX_LINE_LENGTH 256
int main(int argc, char *argv[] ){

    FILE *f1, *f2;
    char p1[MAX_LINE_LENGTH+1];
    char p2[MAX_LINE_LENGTH+1]={0};

    char *p=p1;

    if (argc !=2) return (-1);

    if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Syöttötiedostoa ei voi avata ...\n"); return -1; }

    if ((f2 = fopen("temp.txt", "w")) == NULL)
    { printf("Tulostiedostoa ei voi avata..\n"); return -1; }

    while ( (fgets(p, MAX_LINE_LENGTH, f1)) != NULL ){
        if (strcmp(p1,p2)) fputs(p,f2);
        if (p==p1) p=p2; else p=p1;
    }

    fclose(f1);
    fclose(f2);

    remove(argv[1]);
    rename("temp.txt",argv[1]);

    return 0;
}
```

11.3

Kirjoita sanakirjaohjelma, joka tunnistaa vähintään 20 atk-termiä englanniksi ja suomeksi.

Ohjelmalle annetaan komentorivillä sana englanniksi tai suomeksi. Ohjelma tulostaa annetun sanan vastineen toisella kielellä.

Ohjelman käyttöesimerkkejä:

```
C:\>käännä file
file (english) -> tiedosto (suomi)
C:\>käännä bitti
bitti (suomi) -> bit (english)
```

Toteutuksesta: Toteuta sanakirja alustettuna 2-ulotteisena merkkijonotaulukkona. Taulukon rivillä on kaksi sanaa (eli merkkijonoa), toinen suomeksi ja toinen englanniksi.

Ratkaisu:

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

/* Määritellään sanakirja globaaliksi taulukoksi */

char sanakirja[][2][30]={
    {"byte", "tavu"},
    {"computer", "tietokone"},
    {"directory", "hakemisto"},
    {"disk", "levy"},
    {"file", "tiedosto"},
    {"folder", "kansio"},
    {"folder", "hakemisto"},
    {"keyboard", "n\u00e4pp\u00e4rimist\u00e4"},
    {"memory", "muisti"},
    {"monitor", "n\u00e4tin"},
    {"mouse", "hiiri"},
    {"mouse", "osoitinrasia"},
    {"network", "verkko"},
    {"printer", "kirjoitin"},
    {"program", "ohjelma"},
    {"storage", "muisti"},
    {"", ""}
};

/* Esitellään funktiot */

int avustus(char *nimi);
int kaanna(char *sana);

/* Pääohjelma */

int main(int argc, char *argv[])
{
    if(argc==2) kaanna(argv[1]);
    else avustus(argv[0]);
    return 0;
}

int kaanna(char *sana)
{
    int i=0;
    int match=0;
    while (*sanakirja[i][0] != NULL) {
        if(! strcmp(sana, sanakirja[i][0])) { /* etsitään sanalle suomenkielistä vastinetta */
            printf("\n%s (english) -> %s (suomi)\n", sanakirja[i][0], sanakirja[i][1]);
            match=1;
        }

        if(strcmp(sana, sanakirja[i][1])==0) { /* etsitään sanalle englanninkielistä
```

```

        vastinetta */
        printf("\n%s (suomi) -> %s (english)\n", sanakirja[i][1], sanakirja[i][0]);
        match=1;
    }
    i++;
}
/*Jos sanaa ei löytynyt*/
if (! match) printf("\nSanaa '%s' ei löydy sanakirjasta\n", sana);
return 0;
}

int avustus(char *nimi )
{
    printf("\nOhjelma kääntää annetun");
    printf(" sanan suomeksi tai englanniksi.\n\n");
    printf("Syntaksi: '%s sana'\n", nimi);

    return 0;
}

```

11.4

Eräiden Intelin prosessorien osoiteväylien leveydet ja sananpituudet ovat seuraavan taulukon mukaiset:

Prossessorin tyyppi	osoiteväylän leveys(bittia)	sanapituus (bittia)
8080	16	8
8088/8086	20	8/16
80188/80186	20	8/16
80286	24	16
386	32	32
486	32	32
Pentium	32	32

Kirjoita C-ohjelma, jolle voidaan tehdä seuraavanlaisia kyselyjä:

```

C:\>intel /a=32
(näyttää prosessorit, joiden osoitteet ovat 32 b)
a=32: 386, 486, Pentium

```

```

C:\>intel /p=8080
(näyttää 8080:n ominaisuudet)
8080: w=8b, a=16b

```

```

C:\>intel /w=16
(näyttää prosessorit, joiden sanapituus on 16 b)
w=16: 8086, 80186, 80286

```

Suunnittele tietorakenne niin, että tietoja on tarvittaessa helppo päivittää.

Ratkaisu:

```
# include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

int main(int argc, char *argv[])
{
    if (argc != 2)
    {
        printf("\n\nOhjelman k\204ytt\224: %s /komento=arvo\n",argv[0]);
        printf("\t /a - etsi osoitev\204yl\204n leveyden mukaan\n");
        printf("\t /p - etsi prosessoryyppin mukaan\n");
        printf("\t /w - etsi sananpituuden mukaan\n");
        return -1;
    }

    char *p[][3] = {{ "8080", "16", "8"},
                    { "8088", "20", "8"},
                    { "8086", "20", "16"},
                    { "80188", "20", "8"},
                    { "80186", "20", "16"},
                    { "80286", "24", "16"},
                    { "386", "32", "32"},
                    { "486", "32", "32"},
                    { "Pentium", "32", "32"},
                    { "", "", "" }
    };

    char param;
    char arvo[20];
    int i=0;
    int pilkku=0;

    sscanf(argv[1], "%c=", &param);
    sscanf(argv[1]+3, "%s", arvo);

    switch (param) {
        case 'a': printf("\na=%s:", arvo);
            while (*p[i][0] != NULL){
                if (!strcmp(p[i][1], arvo)){
                    if (pilkku) printf(",");
                    printf(" %s", p[i][0]);
                    pilkku=1;
                }
                i++;
            }
            printf("\n");
            break;

        case 'p': printf("\n%s:", arvo);
            while (*p[i][0] != NULL){
                if (!strcmp(p[i][0], arvo))
                    printf(" w=%sb, a=%sb\n", p[i][2], p[i][1]);
                i++;
            }
    }
}
```

```

        printf("\n");
        break;

    case 'w': printf("\nw=%s:", arvo);
                while (*p[i][0] != NULL) {
                    if (!strcmp(p[i][2], arvo)) {
                        if (pilkku) printf(",");
                        printf(" %s", p[i][0]);
                        pilkku=1;
                    }
                    i++;
                }
        printf("\n");
        break;

    default: printf("\nTuntematon komento!\n");
        return -1;
}
return 0;
}

```

12 HARJOITUS

Kaikki tämänkertaiset tehtävät käsitellään kotitehtävinä seuraavalla kerralla ja niistä voi jokaisesta saada yhden lisäpisteen.

12.1

Laadi ohjelma, joka etsii annetusta tekstitiedostosta kaikki sanan **int** esiintymät, korvaa ne tekstillä ** int ** ja tulostaa muunnetun tekstin uuteen tiedostoon. Tiedostojen nimet annetaan ohjelmalle komentoriviparametreina.

Ratkaisu:

```

#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#define ROW_MAX_LEN 500

int main(int argc, char *argv[] ){

    FILE *f1, *f2;
    char pus kuri[ROW_MAX_LEN+1];          /* lukupuskuri */
    char apu[ROW_MAX_LEN+ROW_MAX_LEN]={'\0'}; /* tulostuspuskuri */
    char vanha[]="int";
    char uusi[]="<b> <font color=\"#0000FF\"> int </font> </b>";
    char *p;
    char *q;

    if (argc != 3){
        system("cls");
        printf("\n\nusage: %s infile outfile \n\n", argv[0]);
        return(-1);
    }
}

```

```

if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Syöttötiedostoa ei voi avata ...\n"); return -1; }

if ((f2 = fopen(argv[2], "w")) == NULL)
    { printf("Tulostiedostoa ei voi avata..\n"); return -1; }

while ( (fgets(puskuri, ROW_MAX_LEN, f1)) != NULL ){
    p = puskuuri;
    memset(apu, '\0', 2*ROW_MAX_LEN);
    q=apu;
    while(*p != NULL){
        if ((!strncmp(p, vanha, strlen(vanha)) /* korvattava merkkijono löytyi */
            && ((p == puskuuri)|| /* varmistetaan vielä, ettei löydös */
                (!isalpha(*(p-1)) /* ole laillisen tunnuksen osa */
                 && !isdigit(*(p-1)) /* eli edeltävänä tai seuraavana */
                 && *(p-1)!='_')) /* merkkinä ei ole kirjain, numero */
                && !isalpha(*(p+strlen(vanha))) /* eikä alaviiva */
                && !isdigit(*(p+strlen(vanha)))
                && *(p+strlen(vanha))!='_')){
            strcpy(q, uusi);
            q+=strlen(uusi); /* kopioidaan uusi teksti vanhan */
            p+=strlen(vanha); /* sijasta tulostuspuskuriin */
        }
        else { /* merkkijonoa ei löytynyt tästä kohdasta */
            *q = *p; /* kopioidaan luettu merkki tulostuspuskuriin */
            q++; /* ja siirrytään seuraavaan merkkiin */
            p++;
        }
    }
    fputs(apu, f2);
}

fclose(f1);
fclose(f2);

return 0;
}

```

12.2

Tee ohjelma, joka lukee parametrina annetun tiedoston ja tulostaa sen heksadesimaalisena näytölle.

Esim.	Vaka vanha Väinämöinen laski laulellen vesiä.									
=>	56	61	6B	61	20	76	61	6E	68	61
	20	56	84	69	6E	84	6D	94	69	6E
	65	6E	0A	20	6C	61	73	6B	69	20
	6C	61	75	6C	65	6C	6C	65	6E	20
	76	65	73	69	84					

Ratkaisu:

```

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

```

```
#include <ctype.h>
#define ROW_MAX_LEN 500
#define N_OF_CHARS_ON_ROW 16

int main(int argc, char *argv[] ){

FILE *f1;
unsigned char p;
int i_rivi=1;

    if (argc != 2){
        system("cls");
        printf("\n\K\204tt\224: %s infile  \n\n",argv[0]);
        return(-1);
    }

    if ((f1 = fopen(argv[1], "r")) == NULL)
    { printf("Syöttötiedostoa ei voi avata ...\n"); return -1; }

    while (!feof(f1)){
        p=fgetc(f1);
        printf("%02X ",p);
        i_rivi++;
        if (i_rivi == N_OF_CHARS_ON_ROW){
            i_rivi=1;
            printf("\n");
        }
    }

    fclose(f1);

    return 0;
}
```

12.3

Tee luennoilla esitettyä korttipakkaesimerkkiä hyväksi käyttäen ohjelma, joka valitsee umpimähkään käyttäjän ilmoittaman määrän kortteja, tulostaa kortit käyttäjälle ja laskee korttien silmälukujen summan.

Ratkaisu:

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

/* TIETOTYYPIT *****/

typedef char Maa[7];
typedef char Arvo[3];

typedef struct {
    Maa maa;
    Arvo arvo;
}Kortti;
```

```
/* FUNKTIOIDEN PROTOTYYPIIT *****/

void tayta_pakka(Kortti *);
void sekoita(Kortti *);
int arvo_numeerisena(Arvo);
void anna(int, Kortti *);

/* GLOBAALIT MUUTTUJAT *****/

Arvo arvot[13]= {"A","2","3","4","5","6","7","8",
                "9","10","J","Q","K"};

Maa maat[4]={"Risti","Ruutu","Hertta","Pata"};

/* PÄÄOHJELMA *****/

int main(void)
{
    Kortti pakka[52];

    tayta_pakka(pakka);
    sekoita(pakka);
    anna(5,pakka);
    return 0;
}

/* FUNKTIOIDEN MÄÄRITTELYT*****/

void tayta_pakka(Kortti *pakka)
{
    int i;

    for (i=0; i<=51; i++)
    {
        strcpy(pakka[i].arvo,arvot[i%13]);
        strcpy(pakka[i].maa,maat[i/13]);
    }
}

void sekoita (Kortti *pakka)
{
    int i,x;
    Kortti apu;

    srand(time(NULL));
    for (i=0; i<=51; i++)
    {
        x=rand()%52;
        apu=pakka[i];
        pakka[i]=pakka[x];
        pakka[x]=apu;
    }
}
```



```
void anna (int lkm,Kortti *pakka)
{
    int summa=0;
    int i;

    system("cls");
    printf ("\n\nK,,tesi on:\n");
    for (i=0; i<lkm;i++)
    {
        printf ("%6s %-2s  ",pakka[i].maa, pakka[i].arvo);
        summa+=arvo_numeerisena(pakka[i].arvo);
    }
    printf("\n\nSilm,,lukujen summa on %d.\n",summa);
}

int arvo_numeerisena(Arvo arvo){
    int i_arvo=0;
    int i;
    for(i=0;i<13;i++){
        if (!strcmp(arvo,arvot[i])) i_arvo=i+1;
    }
    if (i_arvo==0) i_arvo = -1000000;
    return i_arvo;
}
```

12.4

Kyselytutkimuksen yhteydessä syntyneeseen tiedostoon on talletettu henkilöiden vastaukset kahteen kysymykseen, joihin vastaukset on annettu asteikolla 1..5. Jos vastaus puuttuu, on arvona 0. Esim.

Leopold Lässärä	5	4
Asarias Ahtopaine	3	2
Manasse Jäppinen	1	1
Talla Puikina	0	4
Petteri Paisuntasäiliö	2	5

...

Tee ohjelma, joka laskee kysymyksiin annettujen vastausten yksiulotteiset jakaumat eli kummankin yksittäisen kysymyksen vastausvaihtoehtoihin annettujen vastausten lukumäärät sekä vastausten ristiintaulukoinnin eli kaksiulotteisen taulukon, jonka rivit vastaavat kysymyksen 1 vastausvaihtoehtoja ja sarakkeet kysymyksen 2 vastausvaihtoehtoja, soluissa on kyseisen vastausyhdistelmän esiintymisfrekvenssi

Ratkaisu:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#define ROW_MAX_LEN 500

void tulosta_suora_jakauma(char *otsikko, int *luvut, int lkm);
void tulosta_ristiintaulukointi(char *otsikko, int *luvut, int lkm);

int main(int argc, char *argv[] ){

    FILE *f1;
    char puskuuri[ROW_MAX_LEN+1];          /* lukupuskuuri */
```

```

int jakauma[6][6]={0};
int reunajakauma[6]={0};
int vast1, vast2;
int i,j;

if (argc != 2){
    system("cls");
    printf("\n\nOhjelman kutsu: %s infile outfile \n\n",argv[0]);
    return(-1);
}

if ((f1 = fopen(argv[1], "r")) == NULL)
{ printf("Syöttötiedostoa ei voi avata ...\n"); return -1; }

while ( (fgets(puskuri, ROW_MAX_LEN, f1)) != NULL ){
    sscanf(puskuri,"%*s %*s %d %d",&vast1,&vast2); /* %*s ohittaa merkkijonon */
    if (vast1<0 || vast1>5) vast1=0; /* muutetaan virheellinen puuttuvaksi */
    if (vast2<0 || vast2>5) vast2=0; /* muutetaan virheellinen puuttuvaksi */
    jakauma[vast1][vast2]++;
}

fclose(f1);

/* Jakaumien tulostus */
/* 1-ulotteiset jakaumat saadaan ristiintaulukoinnista summaamalla */

for (i=0;i<6;i++) {
    reunajakauma[i]=0;
    for (j=0;j<6;j++) reunajakauma[i]+=jakauma[i][j]; /* rivisumma */
}
tulosta_suora_jakauma("\n\nVastauksen 1 jakauma\n===== \n\n",
    reunajakauma,5);

for (i=0;i<6;i++) {
    reunajakauma[i]=0;
    for (j=0;j<6;j++) reunajakauma[i]+=jakauma[j][i]; /* sarakesumma */
}
tulosta_suora_jakauma("\n\nVastauksen 2 jakauma\n===== \n\n",
    reunajakauma,5);

tulosta_ristiintaulukointi(
"\n\nVastausten 1 ja 2
ristiintaulukointi\n===== \n\n"
, jakauma,5);

return 0;
}

void tulosta_suora_jakauma(char *otsikko, int *luvut, int lkm){
int i;
printf(otsikko);
for (i=0;i<lkm+1;i++) printf("\t%5d",i);
printf("\n");
for (i=0;i<lkm+1;i++) printf("\t-----");
printf("\n");
for (i=0;i<lkm+1;i++)
    printf("\t%5d", luvut[i]);
}

```

```
    printf("\n");
}

void tulosta_ristiintaulukointi(char *otsikko, int *luvut, int lkm){
    int i,j;
    printf(otsikko);
    for (i=0;i<lkm+1;i++) printf("\t%5d",i);
    printf("\n");
    for (i=0;i<lkm+1;i++) printf("\t-----");
    printf("\n");
    for (i=0;i<lkm+1;i++) {
        printf("%5d|",i);
        for (j=0;j<6;j++) printf("\t%5d", luvut[i*lkm+j]);
        printf("\n");
    }
}
```

12.5

Suunnittele tietuerakenne ja kirjoita ohjelma, jonka avulla voidaan tallettaa tiedostoon Internet-linkkejä samaan tapaan kuin selainohjelmien suosikkilinkit toiminnossa (Favorites-valikko MS Explorer-selaimessa).

Internet-linkistä talletetaan vähintään:

- käyttäjän antama linkin nimi
- linkin internet-osoite (URL=Uniform Resource Locator)
- käyttäjän antama kuvaus linkistä, max 256 merkkiä
- ryhmä, jonka alle linkki halutaan tallettaa (voi myös puuttua, koodaus?)

Ratkaisu:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#define MAX_RIVIN_PITUUS 255

typedef struct
{
    char nimi[128];
    char url[256];
    char kuvaus[256];
    int ryhmä;
} Suosikki;

char puskuri[MAX_RIVIN_PITUUS+1];

void poistaRivinVaihtoMerkki(char *p);
void kyseleLinkinTiedot(Suosikki *linkki);

int main(int argc, char *argv[])
{
    int jatka = 0;
    char vastaus;
    FILE *f;
    Suosikki linkki;

    if (argc !=2) {
```

```
printf("Ohjelman kutsu: %s linkkitiedosto\n",argv[0]);
return (-1);
}

if ((f = fopen(argv[1], "w")) == NULL)
{ printf("Linkkitiedostoa ei voi avata..\n"); return -1; }

do{

    kyseleLinkinTiedot(&linkki);

    fprintf(f, "%s | %s | %s | %d\n",
            linkki.nimi, linkki.url, linkki.kuvaus, linkki.ryhma);

    printf("\n\nLis\204\204tk\224 viel\204 linkkej\204? (k/e)\n");
    fgets(puskuri, MAX_RIVIN_PITUUS, stdin);
    sscanf(puskuri, "%c", &vastaus);
    if (vastaus == 'k' || vastaus == 'K') jatka = 1; else jatka = 0;

} while (jatka == 1);

printf("\n");
return 0;
}

void kyseleLinkinTiedot(Suosikki *linkki){

    printf("\n\nAnna linkin tiedot\n");
    printf("\tnimi? ");
    fgets(linkki->nimi, MAX_RIVIN_PITUUS/2, stdin);
    poistaRivinVaihtoMerkki(linkki->nimi);

    printf("\tURL? ");
    fgets(linkki->url, MAX_RIVIN_PITUUS, stdin);
    poistaRivinVaihtoMerkki(linkki->url);

    printf("\tkuvaus? ");
    fgets(linkki->kuvaus, MAX_RIVIN_PITUUS, stdin);
    poistaRivinVaihtoMerkki(linkki->kuvaus);

    printf("\tryhm\204tunnus (-1 : ei ryhm\204\204)? ");
    fgets(puskuri, MAX_RIVIN_PITUUS, stdin);
    sscanf(puskuri, "%d", &linkki->ryhma);

}

void poistaRivinVaihtoMerkki(char *p)
{
    if (p != NULL)
    {
        while (*p != '\0')
        {
            if (*p == '\n') { *p = '\0'; break;}
            p++;
        }
    }
}
```

```

    }
}
}

```

12.6

Kirjoita ohjelma, joka lukee näppäimistöltä kaksi 2x2 neliömatriisia ja tulostaa niiden summan, erotuksen ja tulon.

Esim.

A=

a	b
c	d

B =

x	y
z	v

A+B =

a+x	b+y
c+z	d+v

AB =

ax+bz	ay+bv
cx+dz	cy+dv

A-B =

a-x	b-y
c-z	d-v

Ratkaisu:

```

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
#define DIM 2

```

```

/* FUNKTIOIDEN PROTOTYYPIIT *****/

```

```

/* funktioissa on käytetty kolmea eri tapaa välittää kaksikulotteinen taulukko parametrina:

```

1. yleisenä osoitintaulukkona
2. osoitintaulukkona, jonka alkiot osoittavat
int x[DIM] -tyyppisiin taulukoihin
3. osoittimena pitkään yksikulotteiseen taulukkoon, jossa matriisin
rivit ovat peräkkäin

```
*/

int lue_matriisi(int **a, int dim);    /* tapa 1 */
int lue_matriisi2(int a[][DIM]);      /* tapa 2 */
int lue_matriisi3(int *a,int dim_dim); /* tapa 3 */

int tulosta_matriisi(char *otsikko, int **a, int dim); /* tapa 1 */

int laske_summa(int summa[][DIM],int a[][DIM], int b[][DIM]); /* tapa 2 */
int laske_erotus(int *summa, int *a, int *b,int dim);          /* tapa 3 */
int laske_tulo(int *summa[], int *a[], int *b[], int dim);      /* tapa 1 */

char puskuri[128];

int main(void)
{
    int a[DIM][DIM];
    int b[DIM][DIM];
    int tulos[DIM][DIM];
    int *p_a[DIM];
    int *p_b[DIM];
    int *p_tulos[DIM];
    int i;

    for(i=0;i<DIM;i++){
        p_a[i]=a[i];
        p_b[i]=b[i];
        p_tulos[i]=tulos[i];
    }

    lue_matriisi2(a);
    tulosta_matriisi("\n\nLuettu matriisi A tavalla 2:\n\n", p_a,DIM);

    /* allaoleva matriisin uudelleen lukeminen on turha, mutta se */
    /* osoittaa, miten kolmatta lukutapaa voidaan käyttää      */

    lue_matriisi3(&a[0][0],DIM);
    tulosta_matriisi("\n\nLuettu matriisi A tavalla 3:\n\n", p_a,DIM);

    lue_matriisi(p_b,DIM);
    tulosta_matriisi("\n\nLuettu matriisi B tavalla 1:\n\n",p_b,DIM);

    laske_summa(tulos,a,b);
    tulosta_matriisi("\n\nLuettujen matriisien summa on:\n\n", p_tulos,DIM);

    laske_erotus(tulos,a,b,DIM);
    tulosta_matriisi("\n\nLuettujen matriisien erotus on:\n\n", p_tulos,DIM);

    laske_tulo(p_tulos,p_a,p_b,DIM);
    tulosta_matriisi("\n\nLuettujen matriisien tulo on:\n\n", p_tulos,DIM);

    return 0;
}

int laske_summa(int summa[][DIM],int a[][DIM], int b[][DIM]){
    int i,j;
```

```
        for (i=0;i<DIM;i++)
            for (j=0;j<DIM;j++)
                summa[i][j]=a[i][j]+b[i][j];

    return 0;
}

int laske_erotus(int *summa, int *a, int *b,int dim){
    int i;
    for (i=0;i<dim*dim;i++)
        summa[i]=a[i]-b[i];
    return 0;
}

int laske_tulo(int *summa[],int *a[], int *b[], int dim){
    int i,j,k;
    int s;
    for (i=0;i<dim;i++) {
        for (j=0;j<dim;j++){
            s=0;
            for (k=0;k<dim;k++)
                s+=a[i][k]*b[k][j];
            summa[i][j]=s;
        }
    }
    return 0;
}

int lue_matriisi(int *a[], int dim){

    int i,j;
    printf ("Anna %dx%d matriisin \n",dim,dim);
    printf ("alkiot yksitellen riveitt\204in \n");
    for (i=0;i<dim;i++){
        for (j=0;j<dim;j++){
            fgets(puskuri,127,stdin);
            sscanf(puskuri,"%d",&a[i][j]);
        }
    }
    return 0;
}

int lue_matriisi2(int a[][DIM]){
    int i,j;
    printf ("Anna %dx%d matriisin \n",DIM,DIM);
    printf ("alkiot yksitellen riveitt\204in \n");
    for (i=0;i<DIM;i++){
        for (j=0;j<DIM;j++){
            fgets(puskuri,127,stdin);
            sscanf(puskuri,"%d",&a[i][j]);
        }
    }
    return 0;
}

int lue_matriisi3(int *a,int dim){
    int i;
```

```
int ylaraja=dim*dim;
    printf ("Anna %dx%d matriisin \n",dim,dim);
    printf ("alkiot yksitellen riveitt\204in \n");
    for (i=0;i<ylaraja;i++){
        fgets(puskuri,127,stdin);
        sscanf(puskuri,"%d ",(a+i));

    }
    return 0;
}

int tulosta_matriisi(char *otsikko, int *a[], int dim){
    int i,j;
    printf (otsikko);
    printf("\n");

    for (i=0;i<dim;i++){
        for (j=0;j<dim;j++){
            printf("%5d ",a[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```