

TARTU ÜLIKOOL  
ARVUTITEADUSE INSTITUUT

Programmeerimise eksamiülesannete kogu

Helle Hein  
Jüri Kiho  
Reimo Palm  
Eno Tõnisson

Tartu 2007

Käesoleva õppevahendi väljaandmist on toetanud Eesti Infotehnoloogia Sihtasutus projekti „Tiigriülikool“ raames.

Kaanekujundaja Aita Linnas

## Sisukord

Saateks . . . . .	3
Järjendid . . . . .	5
Sõned ja failid . . . . .	22
Graafika . . . . .	37
Kirjalikud ülesanded . . . . .	55

ISBN 978-9949-11-691-1

Tartu Ülikooli Kirjastus  
<http://www.tyk.ut.ee/>  
Tellimus nr 323

# Saateks

Tartu Ülikoolis on programmeerimist õpitud ja õpetatud juba aastakümneid. Aegade jooksul on programmeerimise algkursus olnud küllaltki erinev, küll nime, mahu ja kasutatava programmeerimiskeele poolest, aga ka kursuslaste arvu, eriala ning taseme poolest. Viimastel aastatel on algkursus toimunud nimetuse *MTAT.03.100 Programmeerimine* all. See neljaainepunktiline kursus on olnud kohustuslik kõigile matemaatika-informaatikateaduskonna ja peaaegu kõigile füüsika-keemiateaduskonna esmakursuslastele. Igal aastal on osalejaid olnud ka mitmetelt teistelt erialadelt. Kursuslaste arv on aastati olnud erinev, umbes 300–500. Erinevatel aastatel on kasutatud mõnevõrra erinevat hindamissüsteemi ning jaotust loenguteks, tahvlipraktikumideks ja arvutipraktikumideks, aga alati on kursuse üheks oluliseks osaks olnud eksam.

Õppeaastatel 2003/4—2006/7 kasutati programmeerimise kursuse eksamitel 27 ülesandekomplekti, igaühes 3–4 (reeglina mitmeosalist) ülesannet. Ülesannete lahendamisel olid vajalikud programmeerimise baaselementide (tingimus, tsükkel jms), järjendite, sõnede, failide ning ekraanigraafika kasutamise põhioskused. Mõnedes ülesannetes oli otseselt nõutud programmeerimiskeele Java konstruktsioonide loomist.

Käesolev kogu sisaldab 85 eksamiülesannet, mis on jaotatud nelja peatükki. Kolmes esimeses asuvad ülesanded, mille lahenduseks on programm, neljandas peatükis aga on ülesanded, milles ei nõuta otseselt programmi, küll aga vastuseid küsimustele võimaliku programmi struktuuri, andmevahetuse jm kohta. Ülesannete tekste on stiililiselt ühtlustatud. Läbi ülesannete kordunud lõigud on viidud peatükkide sissejuhatustesse. Mõnevõrra on püütud vähendada tekstide otsesest seostatust programmeerimiskeelega Java, aga siiski on osa ülesandeid täpselt sellise sõnastusega ainult Javas lahendatavad. Sõnastust muutes saab ülesandeid kasutada ka teiste programmeerimiskeelte puhul.

Välja võib tuua rea käeoleva ülesannete kogu võimalikke kasutusviise. Näiteks oleksid ülesanded kasutatavad nimetatud programmeerimise kursuse täiustamisel järgnevatel aastatel, nüüd küll mitte enam eksamiülesannetena, vaid õppematerjalidena. Samuti saaks neid ülesandeid kasutada (vajadusel täiendades, muutes) muudes programmeerimise kursustes nii Tartu Ülikoolis kui ka teistes kõrgkoolides, aga ka üldharidus- ja kutseõppeasutustes. Rõhutada tulebki just täiendamise ja muutmise võimalikkust, sest praegusel kujul võivad ülesanded olla liiga seotud Tartu Ülikooli konkreetse kursuse eesmärkide ja eripäraga. Ülesandeid saab kasutada ka iseseisvaks harjutamiseks. Märkida tuleb küll, et osade ülesannete puhul on lahenduse õigsust lihtsam hinnata (ka lahendajal endal), osade puhul raskem.

Ülesanded on autorite looming, kuigi ilmselt võib olla ka mõjutusi nõ klassikalistest ülesannetest. Paljude ülesannete tekstid on pseudoelulised, kus sõnastuses võib näha autoritel tol ajahetkel meesmõlkunud teemasid (nt rektori valimised, ujula kell, tantsutund). Autorite ring on tegelikult laiem kui kaanel näha. Oma suurema või väiksema panuse on ülesannete valmimisse andnud paljud selle aine

õppejõud. Eriliselt tahaksime tänada Marina Issakovat, Heli Uibot, Vambola Le-pingut, Gunnar Nellist, Ahti Pederit, Margus Treumuthit, kes on mitmetel aastatel selle kursuse juhendamisel kaasa löönud. Aga suured tänud ka teistele juhendaja-tele ning kindlasti tudengitele, kes neid ülesandeid suurema või ka väiksema eduga lahendanud on.

Loodetavasti leiavad lugejad sellest kogumikust lahendamist ja/või arendamist väärivaid ülesandeid. Teated märgatud ebatäpsustest, samuti täpsustused ja soovid saada ülesannete tekstide faile on oodatud aadressil `eno.tonisson@ut.ee`.

# Järjendid

Käesolev peatükk sisaldab ülesandeid järjendite kasutamise kohta. Enamikel juhitudel piisab järjendi ühekordsest läbivaatamisest. Kõikide ülesannete korral on eeldatud, et lisaks nõutud meetoditele koostab lahendaja ka meetodi etteantud järjendi väljastamiseks ekraanile.

## Ülesanne 1 Tantsupaarid (2003/2004)

On antud kaks täisarvujärjendit — poiste pikkuste järjend ja tüdrukute pikkuste järjend (pikkused on antud sentimeetrites). Kumbki järjend ei pruugi olla pikkuste järgi järjestatud. Koostada programm, mis moodustab tantsupaarid (poiss ja tüdruk) järgmise eeskirja kohaselt.

- Kui poisse ja tüdrukeid on võrdselt, siis moodustatakse paarid nii, et kõige pikem poiss tantsib kõige pikema tüdrukuga, pikkuselt järgmised on omavahel paaris jne.
- Kui ühe soo esindajaid on rohkem kui teise soo esindajaid, siis jäetakse sellest soost sobiv arv kõige pikemaid ilma paariliseta.

Peameetodis väljastada lähteandmed (poiste ja tüdrukute pikkused esialgses järjestuses), tantsupaarid ning paariliseta jäänud poiste või tüdrukute pikkused (kui keegi jäi paariliseta).

### Väljundi näidis

Poiste pikkused:

180, 175, 200, 172, 169, 183, 188

Tüdrukute pikkused:

165, 167, 172, 169, 162

Tantsupaarid on:

(183, 172) (180, 169) (175, 167) (172, 165) (169, 162)

Paariliseta jäid poisid pikkustega

200, 188.

## Ülesanne 2 Paarsusühildamine (2003/2004)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Moodi leidmine*. Antud: täisarvujärjend. Tulemus: tagastatakse täisarv, mida leidub järjendis kõige rohkem. Kui selliseid arve on mitu, siis tagastatakse neist vähim.

Näiteks järjendi  $\{-10, 17, 13, 17, -10, 21\}$  korral tagastatakse  $-10$ .

- b) *Järjendite „paarsusühildamine“*. Antud: kaks täisarvujärjendit. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud nii, et esimesest etteantud järjendist on võetud paaritu indeksiga elemendid ja teisest paarisindeksiga

elemendid. Elementide järjestus uues järjendis on eksaminandi valida, aga seda tuleb kommentaaris kirjeldada.

Näiteks lähtejärjendite  $\{0, 1, 2, 3, -1\}$  ja  $\{2, 4, -3\}$  puhul sobib tulemuseks nii järjend  $\{1, 3, 2, -3\}$  kui ka  $\{1, 2, 3, -3\}$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Algandmed ning kõik vahetulemused väljastada ekraanile koos selgitava tekstiga.

### Ülesanne 3                      Signum-funktsioon                      (2003/2004)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Järjendite aritmeetiline keskmine.* Antud: kolm täisarvujärjendit. Tulemus: tagastatakse uus täisarvujärjend, mille iga element on etteantud järjendite vastavate elementide aritmeetiline keskmine, ümardatult täisarvuks. Kui järjendid pole ühepikkused, siis võetakse tulemusjärjendi pikkuseks pikima järjendi pikkus ja viimasteks elementideks on vastavalt kahe või ühe elemendi täisarvuks ümardatud aritmeetiline keskmine.

Näiteks järjendite  $\{1, 0, -4\}$ ,  $\{2, 3, 7, -5\}$ ,  $\{2, -2, 5, 1, 5, 4\}$  puhul arvutatakse tulemusjärjendi elemendid:  $(1 + 2 + 2) / 3 \approx 2$ ;  $(0 + 3 + (-2)) / 3 \approx 0$ ;  $(-4) + 7 + 5) / 3 \approx 3$ ;  $((-5) + 1) / 2 \approx -2$ ;  $5 / 1 \approx 5$ ;  $4 / 1 \approx 4$ .

- b) *Järjendile signum-funktsiooni rakendamine.* Antud: täisarvujärjend. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud nii, et etteantud järjendi kõigile positiivsetele elementidele vastab 1, negatiivsetele elementidele  $-1$  ja nullile 0.

Näiteks järjendi  $\{2, -2, 5, 1, 0, -4\}$  puhul saame  $\{1, -1, 1, 1, 0, -1\}$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Järjendite aritmeetilist keskmist testida erineva pikkusega järjenditega. Algandmed ning kõik vahetulemused väljastada ekraanile koos selgitava tekstiga.

### Ülesanne 4                      Järjendite kokkupanek                      (2003/2004)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Järjendite kokkupanemine.* Antud: kolm täisarvujärjendit. Tulemus: tagastatakse uus täisarvujärjend, milles on kõigepealt lühima etteantud järjendi elemendid, seejärel keskmise ja lõpuks pikima. Kui on ühepikkuseid järjendeid, siis nende järjekord pole oluline.

Näiteks järjendite  $\{2, -2, 5, 1, 5, 4\}$ ,  $\{1, 0, -4\}$ ,  $\{2, 3, 7, -5\}$  puhul on tulemusjärjend  $\{1, 0, -4, 2, 3, 7, -5, 2, -2, 5, 1, 5, 4\}$ .

- b) *Järjendi elementide paarsuse kontroll.* Antud: täisarvujärjend. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud nii, et etteantud järjendi kõigile paarisarvulise väärtusega elementidele vastab 2 ja paaritarvulise väärtusega elementidele vastab 1.

Näiteks järjendi  $\{2, -4, 5, 1, 0, -4\}$  puhul saame  $\{2, 2, 1, 1, 2, 2\}$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Järjendite kokkupanekut testida erineva pikkusega järjenditega. Algandmed ning kõik vahetulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 5 Järjendi ümberpööramine (2003/2004)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Järjendite ühildamine*. Antud: kaks täisarvujärjendit, milles elemendid paiknevad mittekahanevas järjekorras. Tulemus: tagastatakse uus täisarvujärjend, mis koosneb parajasti kõigist etteantud järjendite elementidest. Seejuures peab säilima elementide mittekahanev järjestus, st tulemusjärjendi elemendid peavad samuti olema mittekahanevas järjestuses.

Näiteks lähtejärjendite  $\{-10, 17, 21\}$  ja  $\{-2, 5, 17, 28\}$  ühildamisel saame tulemuseks järjendi  $\{-10, -2, 5, 17, 17, 21, 28\}$ .

- b) *Järjendi „ümberpööramine“*. Antud: täisarvujärjend. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud etteantud järjendi ümberpööramise (viimane element esimeseks, eelviimane teiseks jne) ja elementide märkide muutmise teel.

Näiteks lähtejärjendi  $\{0, 1, 2, 3, -1\}$  puhul on tulemuseks järjend  $\{1, -3, -2, -1, 0\}$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Järjendite ühildamist testida erineva pikkusega järjenditega, kusjuures algjärjendid tuleb koostada nii, et tulemus sisaldaks ka võrdseid elemente. Algandmed ning kõik vahetulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 6 Järjendite liitmine ja korrutamine (2003/2004)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Järjendite liitmine*. Antud: kaks täisarvujärjendit. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud etteantud järjendite elementhaaval liitmise tulemusena. Kui järjendid pole ühepikkused, siis võetakse tulemusjärjendi pikkuseks pikema järjendi pikkus ja tulemusjärjendi viimasteks elementideks pikema järjendi elemendid.

- b) *Järjendi korrutamine täisarvuga*. Antud: täisarvujärjend ja täisarv. Tulemus: tagastatakse uus täisarvujärjend, mis on saadud etteantud järjendi kõigi elementide korrutamisel etteantud täisarvuga.

Koostada peameetod ülaltoodud meetodite testimiseks. Järjendite liitmist testida erineva pikkusega järjenditega. Algandmed ning kõik vahetulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 7                      Korduvad elemendid                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Täisarvu esinemiste arvu leidmine.* Antud: täisarvujärjend ja täisarv. Tulemus: tagastatakse täisarv, mis näitab, mitu korda antud arv antud järjendis sisaldub.
- b) *Järjendi korduvate elementide summa arvutamise.* Antud: täisarvujärjend. Tulemus: tagastatakse selliste arvude summa, mis esinevad antud järjendis rohkem kui üks kord, kusjuures korduvat elementi tuleb arvesse võtta nii mitmes eksemplaris, nagu teda järjendis esineb.

Näiteks järjendis  $\{1, 0, -4, 2, 3, 5, -5, 2, -2, 5, 1, 5, 4\}$  esinevad rohkem kui üks kord parajasti arvud 1, 2, 5. Antud juhul on summa  $1+1+2+2+5+5+5 = 21$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Algandmed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 8                      Täisarvu jaguvus                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Arvu jaguvuse kindlakstegemine.* Antud: positiivne täisarv. Tulemus: *true*, kui arv jagub kolmega, neljaga või viiega; *false* vastasel juhul.
- Näiteks arvu 12 puhul on tulemus *true*, sest 12 jagub kolmega ja neljaga. Arvu 18 puhul on tulemus *true*, sest 18 jagub kolmega. Arvu 14 puhul on tulemus *false*, sest 14 ei jagu ei kolme, nelja ega viiega.
- b) *Järjendi selliste arvude korrutise leidmine, mis jaguvad kolmega, neljaga või viiega.* Antud: positiivsete täisarvude järjend. Tulemus: tagastatakse selliste arvude korrutis, mis jaguvad kolmega, neljaga või viiega.

Näiteks järjendi  $\{12, 3, 14, 10, 19\}$  puhul on korrutiseks on  $12 \cdot 3 \cdot 10 = 360$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Algandmed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 9                      Täisarvu üheliste number                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Arvu üheliste numbri neljaga jaguvuse kindlakstegemine.* Antud: positiivne täisarv. Tulemus: *true*, kui arvu üheliste number jagub neljaga; *false*, kui arvu üheliste number ei jagu neljaga.

Arvu 12 puhul on tulemus *false*, sest 2 ei jagu neljaga. Arvu 14 puhul on tulemus *true*, sest 4 jagub neljaga.



- b) *Järjendi selliste arvude summa leidmine, mille üheliste number jagub neljaga.* Antud: positiivsete täisarvude järjend. Tulemus: tagastatakse selliste arvude summa, mille üheliste number jagub neljaga.

Näiteks järjendi  $\{12, 35, 118, 19, 4\}$  puhul on summaks on  $118 + 4 = 122$ .

Koostada peameetod ülaltoodud meetodite testimiseks. Algandmed ning tulemusd väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 10                      Aritmeetiline keskmine                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Järjendi elementide aritmeetilise keskmise leidmine.* Antud: täisarvujärjend. Tulemus: tagastatakse järjendi elementide aritmeetiline keskmine.

Näiteks järjendi  $\{2, 3, 8, 12, 15, 5\}$  elementide aritmeetiline keskmine on  $(2 + 3 + 8 + 12 + 15 + 5) / 6 = 7,5$ .

- b) *Järjendi põhjal uue järjendi moodustamine.* Antud: täisarvujärjend. Tulemus: täisarvujärjend, mille elemendid omandavad väärtusi hulgast  $\{-1, 0, 1\}$ . Uues järjendis on sama palju elemente kui esialgses. Elemendi väärtuseks on:

- $-1$ , kui esialgse järjendi vastav element on väiksem esialgse järjendi elementide aritmeetilisest keskmisest;
- $0$ , kui esialgse järjendi vastav element on võrdne esialgse järjendi elementide aritmeetilise keskmisega;
- $1$ , kui esialgse järjendi vastav element on suurem esialgse järjendi elementide aritmeetilisest keskmisest.

Koostada peameetod ülaltoodud meetodite testimiseks. Algandmed ja tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 11                      Täisarvu numbrite arv                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Täisarvu numbrite arvu leidmine.* Antud: kümnest tuhandest väiksem positiivne täisarv. Tulemus: tagastatakse täisarvu numbrite arv.

Näiteks arvu 24 numbrite arv on 2; arvu 3434 numbrite arv on 4.

- b) *Järjendi põhjal uue järjendi moodustamine.* Antud: täisarvujärjend, kus esinevad ainult kümnest tuhandest väiksemad positiivsed arvud. Tulemus: täisarvujärjend, milles on sama palju elemente kui esialgses. Iga elemendi väärtuseks on esialgse järjendi vastava elemendi numbrite arv.

Näiteks järjendi  $\{12, 356, 3455, 8\}$  puhul on uus järjend  $\{2, 3, 4, 1\}$ .

Peameetodis testida ülaltoodud meetodeid. Algandmed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 12 Vastandarvulisus (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Kahe arvu vastandarvulisuse kindlakstegemine.* Antud: kaks täisarvu. Tulemus: *true*, kui antud täisarvud on teineteise vastandarvud; *false*, kui pole.

Näiteks arvude  $-2$  ja  $2$  puhul on tulemus *true*. Kui antud on  $0$  ja  $0$ , siis on tulemus samuti *true*.

- b) *Kahe järjendi elementide vastandarvulist kirjeldava järjendi moodustamine.* Antud: kaks ühepikkust täisarvujärjendit. Tulemus: täisarvujärjend, mille elementide väärtused kuuluvad hulka  $\{1, 0\}$ . Uues järjendis on elemente sama palju kui kummaski antud järjendis. Elemendi väärtuseks on  $1$ , kui antud järjendite vastavad elemendid on vastandarvud, ja  $0$ , kui pole.

Näiteks järjendite  $\{-2, 1, 0, 9\}$  ja  $\{2, 7, 0, 4\}$  puhul on tulemuseks järjend  $\{1, 0, 1, 0\}$ .

Peameetodis testida ülaltoodud meetodeid. Alganamed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 13 Harmooniline keskmine (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Kahe täisarvu harmoonilise keskmise leidmine.* Antud: kaks positiivset täisarvu. Tulemus: antud arvude harmooniline keskmine, mis on ümardatud lähima täisarvuni. Arvude harmooniline keskmine on nende arvude pöördväärtuste aritmeetilise keskmise pöördväärtus. Näiteks arvude  $4$  ja  $10$  harmooniline keskmine on

$$\frac{2}{\frac{1}{4} + \frac{1}{10}} = 5,714\dots \approx 6.$$

- b) *Kahe järjendi elementide harmooniliste keskmiste järjendi moodustamine.* Antud: kaks ühepikkust täisarvujärjendit, milles on vaid positiivsed arvud. Tulemus: tagastatakse selline järjend, mis moodustub vastavate elementide täisarvudeni ümardatud harmoonilistest keskmistest.

Näiteks järjendite  $\{4, 1, 4, 9\}$  ja  $\{10, 7, 400, 4\}$  puhul on tulemuseks järjend  $\{6, 2, 8, 6\}$ .

Peameetodis testida ülaltoodud meetodeid. Alganamed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 14 Täisarvu esimene number (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Arvu esimese numbri kolmega jagumise kindlakstegemine.* Antud: kahekohaline positiivne täisarv. Tulemus: tagastatakse *true*, kui antud täisarvu esimene number jagub kolmega; *false*, kui ei jagu.

Arvu 65 puhul on tulemus *true*, sest 6 jagub kolmega. Arvu 12 puhul on tulemus *false*, sest 1 ei jagu kolmega.

- b) *Järjendi selliste arvude korrutise leidmine, mille esimene number jagub kolmega.* Antud: täisarvujärjend, kus esinevad ainult kahekohalised positiivsed arvud. Tulemus: tagastatakse selliste arvude korrutis, mille esimene number jagub kolmega.

Näiteks järjendi {12, 31, 54, 61} puhul on korrutiseks  $31 \cdot 61 = 1891$ .

Ülaltoodud meetodite testimine viia läbi peameetodis. Algandmed ning tulemused tuleb väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 15                      Täisarvu ristsumma                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Arvu ristsumma seitsmega jagumise kindlakstegemine.* Antud: kahekohaline positiivne täisarv. Tulemus: tagastatakse *true*, kui antud täisarvu ristsumma jagub seitsmega, ja *false*, kui ei jagu.

Arvu 86 puhul on tulemus *true*, sest  $8 + 6 = 14$  jagub seitsmega. Arvu 17 puhul on tulemus *false*, sest  $1 + 7 = 8$  ei jagu seitsmega.

- b) *Järjendi selliste arvude summa leidmine, mille ristsumma jagub seitsmega.* Antud: täisarvujärjend, kus esinevad ainult kahekohalised positiivsed arvud. Tulemus: tagastatakse selliste arvude summa, mille ristsumma jagub seitsmega.

Näiteks järjendi {52, 31, 10, 86} puhul on summaks  $52 + 86 = 138$ .

Peameetodis testida ülaltoodud meetodeid. Algandmed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 16                      Paaritud arvud                      (2004/2005)

Koostada programm, mis sisaldab meetodeid järgmiste ülesannete lahendamiseks.

- a) *Arvu numbrite paarituse kindlakstegemine.* Antud: kahekohaline positiivne täisarv. Tulemus: tagastatakse *true*, kui antud täisarvu mõlemad numbrid on paaritud, vastasel juhul tagastatakse *false*.

Arvu 75 puhul on tulemus *true*, sest nii 7 kui 5 on paaritud. Arvu 61 puhul on tulemus *false*, sest 6 ei ole paaritu.

- b) *Järjendi selliste arvude summa leidmine, mille mõlemad numbrid on paaritud.* Antud: täisarvujärjend, kus esinevad ainult kahekohalised positiivsed arvud. Tulemus: tagastatakse selliste arvude summa, mille mõlemad numbrid on paaritud.

Näiteks järjendi  $\{25, 31, 40, 75\}$  puhul on summaks  $31 + 75 = 106$ .

Peameetodis testida ülaltoodud meetodeid. Algandmed ning tulemused väljastada ekraanile koos selgitava tekstiga.

## Ülesanne 17 Võidusõit (2005/2006)

Ühte järjendisse on salvestatud võidusõidul osalevate autode hetkekaugused (km) finišist, teise järjendisse aga nende autode kiirused (km/h). Kirjutada programm, mis leiab, mitu autot on  $t$  tunni pärast finišisse jõudnud. (Teepikkus  $s$ , kiirus  $v$  ja aeg  $t$  on teatavasti seotud valemiga  $s = vt$ .)

Programmis peavad olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- a) Antud on autode kauguste järjend, kiiruste järjend ning aeg. Muuta kauguste järjendi elementide väärtusi järgmisel viisil: iga auto kaugust vähendada antud aja jooksul läbitud teepikkuse võrra; negatiivne uus kaugus asendada nulliga.
- b) Antud on järjend autode kaugustega finišist. Leida nende autode arv, mille kaugus finišist on null.
- c) Antud täisarvude  $n$ ,  $a$  ja  $b$  ( $n > 0$ ,  $a < b$ ) korral luua juhuarvude järjend, mille pikkus on  $n$  ja mille elementide väärtused on juhuarvud lõigust  $[a, b]$ .

### Peameetodis

- (1) kirjeldatakse kaks näitejärjendit (nt kaugused 100, 115, 150, 135 ja kiirused 61, 65, 80, 60);
- (2) kui programm on käivitatud ilma käsureaargumentideta, siis arvutatakse tulemus nende näitejärjendite korral;
- (3) vastasel korral loetakse esimene käsureaargument autode arvuks ( $n$ ) ja näitejärjenditeks võetakse kaks  $n$ -elemendilist juhujärjendit: kaugused lõigust  $[200, 300]$  ja kiirused lõigust  $[70, 170]$ ;
- (4) mõlemal juhul on sõiduaajaks 2 tundi.

### Väljundi vorm

Kaugused:

... ..

Kiirused:

... ..

Kaugused ... tunni pärast:

... ..

Saabunuid: ...

## Ülesanne 18 Raamatukoi (2005/2006)

Riiulil paikneb järjestikku  $n$  entsüklopeediaköidet, iga köite esimene kaas vahetult eelmise köite tagumise kaane vastas. Köidete lehekülgede arvud on kirja pandud järjendisse  $a$ . Raamatukoi alustas närimist  $k$ -nda köite  $x$ -ndalt leheküljelt ja lõpetas  $l$ -nda köite  $y$ -ndal leheküljel. Ühe lehekülje parandamine maksab 20 senti ja ühe kaane parandamine 2 krooni. Kirjutada programm, mis arvutab välja, millise kahju koi tekitas.

Programmis peab olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- a) Leida juhuslik täisarv lõigult  $[p, q]$ .
- b) Antud täisarvude  $p, q$  ja  $n$  ( $p < q, n > 0$ ) korral luua järjend, mille pikkus on  $n$  ja mille iga element on juhuslik täisarv lõigult  $[p, q]$ .
- c) Antud järjendi  $a$  (köidete lehekülgede arvud) ja täisarvude  $k, x, l, y$  (närimise alustamise ja lõpetamise kohad) korral leida, mitu lehekülge koi läbi näris.

*Näide.* Kui lehekülgede arvud on  $a = \{24, 48, 66, 42, 36\}$  ning koi alustab 3. köite 55. leheküljelt ( $k = 3, x = 55$ ) ja lõpetab 5. köite 20. leheküljel ( $l = 5, y = 20$ ), siis ta närib läbi  $(66 - 54) + 42 + 20 = 74$  lehekülge.

- d) Antud järjendi  $a$  (köidete lehekülgede arvud) ja täisarvude  $k, l$  (närimise alustamise ja lõpetamise köited) korral leida, mitu kaant koi läbi näris.

*Näide.* Kui andmed on samad nagu eelmises näites, siis närib koi läbi 3. köites ühe kaane, 4. köites kaks kaant ja 5. köites ühe kaane, kokku 4 kaant.

- e) Antud järjendi  $a$  (köidete lehekülgede arvud) ja täisarvude  $k, x, l, y$  (närimise alustamise ja lõpetamise kohad) korral arvutada (kahte eelmist meetodit kasutades) koi tekitatud kahju.

*Näide.* Koi näris läbi 74 lehekülge, mille parandamine maksab  $74 \cdot 0,2 = 14,8$  krooni, ning 4 kaant, mille parandamine maksab  $4 \cdot 2 = 8$  krooni. Kokku seega 22,8 krooni.

- f) Väljastada antud järjend.

### Peameetodis

- (1) luuakse  $n$ -elemendiline järjend, mille elemendid on juhuslikud täisarvud 500 ja 1000 vahel; arv  $n$  saadakse käsurealt; kui aga käsureargumente pole, siis  $n = 10$ ;
- (2) väljastatakse see järjend ekraanile;
- (3) valitakse juhuslikult  $k, x, l, y$  väärtused:  $k$  väärtus lõigult  $[1, 5]$ ,  $x$  väärtus lõigult  $[200, 400]$ ,  $l$  väärtus lõigult  $[6, 10]$ ,  $y$  väärtus lõigult  $[200, 400]$ , ja väljastatakse kõik need väärtused ekraanile;
- (4) arvutatakse kokku koi tekitatud kahju ja väljastatakse see ekraanile.

## Väljundi näidis

Köidete lehekülgede arvud:

538 990 787 737 943 959 932 826 696 766 622 778

Koi alustas 2. köite 395. leheküljelt  
ja lõpetas 8. köite 200. leheküljel.

Koi tekitas kahju 1060.8 krooni.

## Ülesanne 19 Suusasprint (2005/2006)

Suusasprindivõistlusel startisid suusatajad vastavalt stardinumbrile iga 15 sekundi järel. Stopper, mis mõõdab aega sekundites, käivitati esimese startija stardihetkel. Esimesse järjendisse salvestati suusatajate stardinumbrid lõpetamise järjekorras (täisarvudena) ning teise järjendisse salvestati stopperinäidud lõpetamise hetkedel samas järjekorras (reaalarvudena). Katkestajate näitudeks märgiti null sekundit ja nende andmed asetati järjendite lõppu.

Koostada programm, mis väljastab võitja stardinumbrile ja aja ning katkestajate arvu.

Programmis peab olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- Antud on stardinumbrite järjend. Moodustada stardiaegade järjend. Näiteks kui stardinumbrile on 6, siis vastav stardiaeg on 75.
- Antud on stardiaegade järjend ja stopperinäitude järjend. Moodustada distantsi läbimisaegade järjend (läbimisaeg = stopperinäit – stardiaeg, katkestajate puhul jääb distantsi läbimisaegaks null).
- Antud on distantsi läbimisaegade järjend. Leida võitja tulemuse indeks järjendis.
- Antud on stopperinäitude järjend. Leida katkestajate arv.

### Peameetodis

- kirjeldatakse kaks sama pikkusega näitejärjendit (üks stardinumbrite ja teine stopperinäitude jaoks);
- leitakse võitja stardinumbrile ja distantsi läbimisaeg ning väljastatakse see ekraanile;
- leitakse katkestajate arv ning väljastatakse see ekraanile.

## Väljundi näidis

Stardinumbrite {2, 1, 4, 5, 8, 7, 3, 6} ning stopperinäitude {148,2; 150,4; 165,7; 181,2; 226,0; 231,9; 0,0; 0,0} puhul on väljund

Võitja stardinumbrile oli 4 ja tema aeg oli 120.7.

Katkestajaid oli 2.

## Ülesanne 20 Toimetulek (2005/2006)

Osakonna töötajate kuupalgad (täisarvud) on salvestatud ühte järjendisse. Teise järjendisse on samas järjekorras salvestatud töötajate igakuised kulutused (samuti täisarvud). Nendel, kelle palk on osakonna keskmisest palgast väiksem, tõstetakse palka 15%. Kirjutada programm, mis leiab, mitmel töötajal ei ole isegi pärast palgatõusu võimalik oma palgast kõiki kulutusi ära katta.

Programmis peab olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- a) Antud on palgade järjend. Leida keskmine palk ja ümardada see sajandikeni.
- b) Antud on palgade järjend ja keskmine palk. Moodustada uus järjend järgmisel viisil: kui töötaja palk on väiksem kui keskmine palk, siis uus palk on senisest 15% võrra suurem (ümardada täisarvuni), vastasel juhul on uus palk senisega võrdne.
- c) Antud on palgade järjend ja kulutuste järjend. Leida, mitmel töötajal ületavad kulutused palka.
- d) Antud täisarvude  $n$ ,  $a$  ja  $b$  ( $n > 0$ ,  $a < b$ ) korral luua juhuslike täisarvude järjend, mille pikkus on  $n$  ja mille elementide väärtused on lõigust  $[a, b]$ .

### Peameetodis

- (1) väljastatakse ekraanile autori nimi ja matriklinumber;
- (2) kirjeldatakse kaks sama pikkusega näitejärjendit (nt palgad 4503, 2003, 4001, 3002, 8014, 3456 ja kulutused 4500, 3000, 2000, 5000, 2000, 3000);
- (3) kui programm on käivitatud ilma käsureargumentideta, siis arvutatakse tulemus nende näitejärjendite korral;
- (4) vastasel korral loetakse käsurea esimene argument töötajate arvuks ( $n$ ) ja näitejärjenditeks võetakse kaks  $n$ -elementilist juhujärjendit: palgad lõigust  $[2000, 10000]$  ja kulutused lõigust  $[1000, 9000]$ ;
- (5) leitakse keskmine palk, palgad pärast palgatõusu ning nende töötajate arv, kelle kulutused pärast palgatõusu ületavad palka;
- (6) väljastatakse andmed vastavalt järgnevale väljundi näidisele.

### Väljundi näidis

Osakonna töötajate palgad on  
4503; 2003; 4001; 3002; 8014; 3456.

Igakuised kulutused on  
4500; 3000; 2000; 5000; 2000; 3000.

Keskmine palk on 4163.17. Uued palgad on  
4503; 2303; 4601; 3452; 8014; 3974.

Kulutused ületavad palka 2 töötajal.

## Ülesanne 21 Maleklubi (2005/2006)

Klubide Ratsu ja Oda vaheline malematš toimub  $n$  laual. Ühe klubi mängijate reitingud (täisarvud lõigust  $[1000, 3000]$ ) on laudade järjekorras (mis ei pruugi olla sama mis reitingute suuruste järjekord) salvestatud ühte järjendisse. Teise järjendisse on samalaadselt salvestatud teise klubi mängijate reitingud. Igal laual toimub üks mäng. Kui kahe vastasmängija reitingute erinevus on suurem kui 400, siis võidab parema reitinguga mängija tõenäosusega 50%, halvema reitinguga mängija tõenäosusega 25% ja mäng jääb viiki tõenäosusega 25%. Kui reitingute erinevus ei ole suurem kui 400, siis on kummagi mängija võiduvõimalused 25% ning mäng jääb viiki tõenäosusega 50%. Kirjutada programm, mis väljastab mängijate reitingud ja mängude tõenäolised tulemused.

Programmis peab olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- a) Antud on kahe mängija reitingud. Määrata ülaltoodud reeglite abil juhuslikult mängu tulemus ja tagastada see täisarvuna (1, kui võitis esimene mängija; 2, kui võitis teine mängija; 0, kui mäng jäi viiki).
- b) Antud on kaks reitingute järjendit. Tagastada tulemuste järjend, mis koosneb vastavalt eri laudade mängude tulemustele arvudest 1, 2 ja 0.
- c) Antud on tulemuste järjend ja täisarv. Tagastada järjend, mis koosneb selle täisarvuga võrduvate elementide indeksitest antud järjendis.

### Peameetodis

- (1) kirjeldatakse kaks sama pikkusega näitejärjendit (nt klubi Ratsu mängijate reitingud 2200, 2220, 2300, 1500, 1800, 1400 ja klubi Oda mängijate reitingud 2700, 2900, 1800, 1500, 2000, 1700);
- (2) leitakse mängude tulemused ning väljastatakse mängijate reitingud ja mängude tulemused vastavalt järgnevale väljundi näidisele.

### Väljundi näidis

Ratsu mängijate reitingud on  
2200, 2220, 2300, 1500, 1800, 1400.

Oda mängijate reitingud on  
2700, 2900, 1800, 1500, 2000, 1700.

Ratsu võidab mängu(d) 1, 4.

Oda võidab mängu(d) 2, 3, 5.

Viigistatakse mäng(ud) 6.

## Ülesanne 22 Eksam (2005/2006)

Programmeerimise eksamiks on reserveeritud  $n$  arvutiklassi numbritega 0, 1, 2, ...,  $n - 1$ . Arvutiklasside arvutite arvud (täisarvud lõigust  $[10, 40]$ ) on salvestatud ühte järjendisse. Teise järjendisse on arvutiklasside järjekorras salvestatud



nende üliõpilaste arvud (täisarvud lõigust  $[0, 200]$ ), kes soovivad vastavates klassides eksamit sooritada. Kirjutada programm, mis leiab iga klassi puhul, mitmes voorus tuleb selles klassis eksamit korraldada, mitmes klassis tegelikult eksam toimub ja millistes klassides toimub rohkem kui 3 vooru.

Programmis peab olema testimisotstarbeline peameetod ja meetodid järgmiste alamülesannete lahendamiseks.

- a) Antud on ühe klassi arvutite arv ja üliõpilaste arv, kes soovivad selles klassis eksamit sooritada. Leida mitmes voorus tuleb selles klassis eksamit korraldada.
- b) Antud on arvutite arvude ja üliõpilaste arvude järjendid. Tagastada järjend, mis näitab iga klassi puhul, mitmes voorus tuleb selles klassis eksamit korraldada.
- c) Antud on klasside eksamivoorude arvude järjend. Leida, mitmes klassis tegelikult eksam toimub, st mitme klassi puhul on voorude arv positiivne.
- d) Antud on täisarvujärjend ja täisarv. Tagastada järjend, mis koosneb sellest täisarvust suuremate elementide indeksitest antud järjendis.

### Peameetodis

- (1) kirjeldatakse kaks sama pikkusega näitejärjendit (nt arvutite arvud 10, 15, 30, 20, 18 ja eksamit sooritada soovivate üliõpilaste arvud 21, 0, 148, 74, 36);
- (2) leitakse, mitmes voorus igas klassis tuleb eksamit korraldada, mitmes klassis tegelikult eksam toimub ja millistes klassides toimub eksam rohkem kui 3 voorus;
- (3) väljastatakse andmed ja tulemused vastavalt järgnevale väljundi näidisele.

### Väljundi näidis

Arvutite arvud on

10, 15, 30, 20, 18.

Eksamit sooritada soovivate üliõpilaste arvud on

21, 0, 148, 74, 36.

Voorude arv klasside kaupa on

3, 0, 5, 4, 2.

Eksam toimub 4 klassis.

Voorude arv on üle 3 klassi(de)s

2, 3.

## Ülesanne 23                      Miljonäri spordiklubi                      (2006/2007)

Miljonär asutas korvpalliklubi ja palkas mängijad. Algul ei osanud ta palku targemini määrata kui korrutas iga mehe pikkuse (cm) 100-ga. Varsti aga hakkas talle paistma, et mitte kõik mehed ei õigusta oma palka. Miljonär palkas analüütiku ja

käskis sellel asjal uurida ning teha ettepanek uute palkade määramiseks. Analüütik osutus Java-programmeerijaks ning koostas järgmise programmi, mille eest sai ka heldelt tasutud. Koostage nüüd Teie ka selline programm.

Peameetodi alguses on kirjeldatud algandmed kolme täisarvujärgendina: meeste pikkused, mängus oldud ajad, visatud punktid. Programmis on meetodid järgmiste ülesannete lahendamiseks.

- Arvutada meeste kasutegurid (punktid/aeg).
- Leida kasutegurite järjendi suurim väärtus.
- Arvutada uus palk valemiga

$$(\text{mehe vana palk}) \cdot (\text{mehe kasutegur}) / (\text{suurim kasutegur}).$$

Kui aga selle kohaselt tuleks uus palk alla 10000, siis võtta uueks palgaks 10000.

Peameetodis kirjeldatakse kolm sama pikkusega täisarvujärjendit ja väljastatakse andmed vastavalt järgnevale väljundi näidisele.

### Väljundi näidis

Pikkused:

178      189      210      196      210      196

Ajad:

200      150      210      180      110      100

Punktid:

145      160      145      150      145      130

Kasutegurid:

0.73      1.07      0.69      0.83      1.32      1.3

Uued palgad:

10000    15293    11000    12390    21000    19329

## Ülesanne 24

## Tantsutund

(2006/2007)

Tantsutunni alguseks on kohal  $x$  neidu ja  $y$  noormeest, kusjuures alati  $x > y$ . Tunni alguses moodustatakse tantsupaarid loosi teel: neidudest valitakse juhuslikult välja  $y$  neidu, kes pannakse noormeestega pikkuste järgi selleks tunniks paari. Koostada programm paaride moodustamiseks.

Peameetodi alguses olgu kirjeldatud kaks täisarvujärjendit — neidude pikkuste (cm) järjend ja noormeeste pikkuste (cm) järjend.

Programmis peab kindlasti olema meetod *valim* neidude valimiseks. Meetodi parameetriteks on neidude pikkuste järjend ja noormeeste arv ( $y$ ). Tulemuseks on järjend pikkusega  $y$ , mille elementideks on väljavalitud neidude indeksid, st mittekorduvad juhuslikud täisarvud poollõigul  $[0, x)$ . Leitud järjend sorteeritakse enne tagastamist kasvavalt.

Meetodi *valim* tööks on otstarbekohane kirjeldada veel muid meetodeid.

**Peameetodis**

- (1) väljastatakse algandmed (kaks täisarvujärjendit), näiteks

Neiud: 176 165 148 161 159 155 154 150 149 157

Noormehed: 180 190 183 176 165 169

- (2) rakendatakse meetodit *valim* ning selle tulemuse põhjal väljastatakse vastus järgmise näidise kohaselt:

noormees nr 0 (190) & neiu nr 1 (165)

noormees nr 1 (183) & neiu nr 2 (161)

noormees nr 2 (180) & neiu nr 4 (157)

noormees nr 3 (176) & neiu nr 5 (155)

noormees nr 4 (169) & neiu nr 6 (154)

noormees nr 5 (165) & neiu nr 8 (149)

## Ülesanne 25                      Raudteeülesõidukoht                      (2006/2007)

Linnapea otsustas raudteeülesõidukoha liiklusprobleemile leevenduse otsimiseks algatada läbirääkimised raudteefirmaga. Jõuti kokkuleppele, et raudteefirma seab rongide sõiduplaani nii, et igas ööpäevas on ülesõidukoht nn tipptunnil autodele lahti 75 minutit järjest. Samas võib raudteefirma ööpäevas ülesõidukoha autodele suletuna hoida 60 minutit järjest, nn rahutunnil. Tipptunni ja rahutunni ajad määrab linn.

Linn tellis liiklusfirmalt uuringu, mille eesmärgiks on koguda andmeid autode liiklussageduse kohta raudteeülesõidukohas. Uuringu tulemuseks on 96-elementiline järjend, mis on saadud järgmiselt. Ööpäeva 24 tundi on jaotatud järjestikusteks 15-minutilisteks intervallideks. Tulemusjärjendi iga elemendi väärtuseks on nende autode keskmine arv, mis ületavad raudteeülesõidukohta vastava intervalli jooksul.

Koostada programm, mis leiab tipptunni kellaaaja ja rahutunni kellaaaja. Tipptunni moodustavad sellised 5 järjestikust intervalli, milles autode koguarv on suurim. On teada, et kesköö ei kuulu tipptunni sisse. Rahutunni moodustavad sellised 4 järjestikust intervalli, milles autode koguarv on vähim. Kesköö võib kuuluda rahutunni sisse. Kui tipptunni või rahutunni jaoks on võrdseid kandidaate, siis valitakse neist ükskõik milline. Tipptunni ja rahutunni lõikumist programselt vältima ei pea.

Programmis peavad olema vähemalt järgmised meetodid.

- Liiklussageduse modelleerimine: moodustada 96-elementiline järjend, milles 16 esimest ja 8 viimast elementi on juhuslikud täisarvud lõigul  $[0, 25]$ , vahepealsed aga juhuslikud täisarvud lõigul  $[20, 99]$ .
- Antud järjendi korral leida elemendi indeks, millest alates viie järjestikuse elemendi summa on suurim.
- Antud järjendi korral leida elemendi indeks, millest alates nelja järjestikuse elemendi summa on vähim. Arvesse tulevad ka järjendi viimased elemendid (vastavalt jätkatakse summat liidetavatega järjendi algusest).

- d) Antud intervalli-indeksi korral leida intervalli alguse kellaeg sõnena, nt kui indeks = 5, siis kellaeg = "1.15".

### Vastuse näidis

Andmed: ... ..

Tiip tund: 7.15 - 8.30

Rahutund: 2.45 - 3.45

## Ülesanne 26 Tenniseturniir (2006/2007)

Tenniseturniirist võtab osa  $m$  meest ja  $n$  naist ( $m > n$ ). Iga osavõtja kohta on teada tema reiting — täisarv vahemikus 1...100. Osavõtjatest moodustatakse segapaarid järgmise skeemi põhjal.

- A. Kõige kõrgema reitinguga mees ja kõige madalama reitinguga naine.
- B. Ülejäänud naistest kõige kõrgema reitinguga naine ja ülejäänud meestest algele meeste keskmisele reitingule kõige lähemal oleva reitinguga mees.

Edasi korraldatakse ülejäänud mängijate jaoks samme A ja B, kuni paarid on moodustunud. Kui mõnele kohale on reitingute järgi võrdseid kandidaate, siis võetakse järjekorras eespool seisev. Koostada programm segapaaride moodustamiseks.

Peameetodi alguses olgu kirjeldatud kaks täisarvujärjendit — meeste ja naiste reitingute järjendid. Peameetodis peavad olema meetodid järgmiste ülesannete lahendamiseks.

- a) Kõrgeima reitinguga (veel valimata) mängija järjekorranumbri leidmine.
- b) Madalaima reitinguga (veel valimata) mängija järjekorranumbri leidmine.
- c) Keskmise reitingu leidmine.
- d) Algele keskmisele reitingule kõige lähemal oleva reitinguga isiku järjekorranumbri leidmine.

### Peameetodis

- (1) väljastatakse algandmed (kaks täisarvujärjendit), näiteks

Mehed: 39 23 42 73 2 31

Naised: 80 31 44 16

- (2) koostatakse paarid, kasutades eelkirjeldatud meetodeid.

*Näpunäide.* Kirjutada järjendisse juba segapaari koosseisu valitud mängija reitinguks null.

### Väljundi näidis

mees nr. 3 (73) & naine nr. 3 (16)

mees nr. 0 (39) & naine nr. 0 (80)

mees nr. 2 (42) & naine nr. 1 (31)

mees nr. 5 (31) & naine nr. 2 (44)

## Ülesanne 27                      Kartulite sorteeriija                      (2006/2007)

Kartulite sorteerimismasina põhiosaks on renn, milles on viis erineva suurusega ringikujulist auku. Augud paiknevad renni põhjas läbimõõdu suurenemise järjekorras. Sorteeritavad kartulid lastakse mööda renni veerema. Renni iga augu all on salv, kuhu kukuvad vastavast august läbi mahtunud kartulid. Renni lõpus on veel üks salv nende kartulite jaoks, mis ühestki august läbi ei mahu.

Kirjutada programm, milles on kirjeldatud renn viieliikmelise aukude diameetrite järjendina ja kartulite diameetrite järjend. Diameetrid on antud sentimeetrites, nad ei tarvitse olla tingimata täisarvud. Programm leiab kuueliikmelise järjendi — renni all ja lõpus olevatesse salvedesse antud kartulijärjendi sorteerimisel sattunud kartulite arvud. Võib eeldada, et kartulid on kerakujulised ja auku kinni ei jää.

Programmis peab olema lisaks peameetodile vähemalt meetod, mis saab parameetritena ette aukude ja kartulite diameetrite järjendid ning moodustab 6-liikmelise tulemusjärjendi.

### Peameetodis

- (1) väljastatakse algandmed, näiteks

Kartulid: 3.1 5.05 7.0 3.9 8.5 12.0 3.8 4.3 4.4 8.1 9.0 1.1 8.8  
 Augud: 2.1 3.9 4.5 7.0 8.11

- (2) väljastatakse tulemus, näiteks

Salvedes: 1    2    3    1    2    4

# Sõned ja failid

Käesolevasse peatükki on koondatud sõnetöötlemise ülesanded. Need ülesanded (peale paari erandi) nõuavad ühtlasi ka failitöötlemise vahendite kaasamist.

Enamasti saadakse töödeldavad sõned tekstifaili ridadest. Mõnel juhul peab lahendusprogrammi väljundiks olema samuti tekstifail. Lähtefaili ridade arvu pii-rang ülesande tekstis vihjab võimalusele töödelda andmed alles pärast kõigi faili-ridade sisestamist.

## Ülesanne 28 Nihkešiffer (2003/2004)

Üks vanemaid šifreid teksti kodeerimiseks on nn nihkešiffer, mis töötab järgmi-selt. Olgu antud tähestik *0123456789abcdefghijklmnopqrstuvwxyöääü*. Nihkešifri puhul määrab kodeerimisvõti nihke tähestikus. Näiteks kodeerides selles tähes-tikus võtmega 3 sõna *tere*, saame tulemuseks *whuh*, st iga sümbol asendatakse temast tähestikus kolme sümboli võrra paremal paikneva sümboliga:  $t \rightarrow w$ ,  $e \rightarrow h$  ja  $r \rightarrow u$ . Tähestiku lõpuosa sümbolite puhul jätkatakse vajadusel asendaja otsimist tähestiku algusest, näiteks  $\ddot{o} \rightarrow 1$ .

Dekodeerimine toimub analoogiliselt, kuid nihet arvestatakse vasakule, st ko-deeritud teksti sümbolid asendatakse võtme väärtuse võrra tähestikus vasakul paiknevate sümbolitega. Tähestiku algusosa sümbolite puhul jätkatakse vajadu-sel asendaja otsimist tähestiku lõpust, näiteks  $0 \rightarrow \ddot{a}$ .

Kirjutada programm sõne kodeerimiseks ja dekodeerimiseks. Kodeeritav sõne ja võti antakse ette käsurealt, tähestik on programmi sisse kirjutatud, näiteks

```
String sümbrid = "0123456789abcdefghijklmnopqrstuvwxyöääü";
```

Programmis peavad olema järgmised meetodid.

- Meetod *kodeeri*, mis kodeerib etteantud sõne etteantud võtmega ja tagastab kodeeritud sõne. Sümbolid, mis tähestikus puuduvad, tuleb jätta muutmata.
- Meetod *dekodeeri*, mis dekodeerib etteantud sõne etteantud võtmega ja ta-gastab dekodeeritud sõne. Sümbolid, mis tähestikus puuduvad, tuleb jätta muutmata.
- Peameetodis analüüsida esmalt käsuresisendit. Ebasobivate argumentide korral väljastada kasutusjuhend. Argumendiks olevale sõnele rakendada es-malt meetodit *kodeeri* ja tulemusele seejärel meetodit *dekodeeri*. Väljastada tuleb kõik vahetulemused: esialgne sõne, kodeeritud sõne ja dekodeeritud sõne.

## Ülesanne 29 Alamsõne asendamine (2003/2004)

Koostada programm sõnede asendamiseks tekstifailis. Antud: failinimi, asendatav sõne, asendav sõne. Tulemus: asendustega tekst kirjutatakse faili *asendatud.txt*; ekraanile väljastatakse tehtud asenduste arv.

Algandmed saadakse käsureaargumentidena. Ebasobiva argumentide arvu korral väljastatakse kasutusjuhend.

### Näide

Olgu failis `loomad.txt` tekst

*Ega hunt hunti murra. Hunt kodu lähedalt ei murra. Paberihunt on kantseleis.*

Pärast käsku

```
java Asenda loomad.txt hunt kriimsilm
```

ilmub ekraanile

Failis `loomad.txt` asendati sõne "hunt" 3 korda sõnega "kriimsilm".

ja faili `asendatud.txt` kirjutatakse tekst

*Ega kriimsilm kriimsilmi murra. Hunt kodu lähedalt ei murra.  
Paberikriimsilm on kantseleis.*

## Ülesanne 30 Sõne otsimine (2003/2004)

Koostada programm tekstifailist sõne otsimiseks. Antud: faili nimi, ridade arv leheküljel ja sõne. Faili käsitletakse tekstifailina, milles iga *ridu\_leheküljel* ( $>0$ ) rea järel algab uus lehekülg. Tulemus: ekraanile väljastatakse faili kõigi nende lehekülgede numbrid koos reanumbritega ( $1, \dots, \text{ridu\_leheküljel}$ ), millel etteantud sõne esineb.

Algandmed saadakse käsureaargumentidena. Ebasobiva argumentide arvu või tüübi (arvuks mitteteisendatava sõne) korral väljastatakse kasutusjuhend.

### Näide

Pärast käsku

```
java Otsija SV.java 30 static
```

ilmub ekraanile

Sõna 'static' esineb failis 'SV.java' järgmistel lehekülgedel

ja ridades:

```
lk 1    7
        10
lk 2    3
        7
        11
```

Käsu

```
java Otsija
```

puhul aga saame

Programm teeb etteantud tekstifaili puhul kindlaks, millistel lehekülgedel ja millistes lehekülje ridades esineb etteantud sõna.

Programmi kasutamine:

```
java Otsija failinimi ridu_leheküljel sõna
    failinimi: selle faili nimi, millest sõna otsitakse
    ridu_leheküljel: ridade arv leheküljel (positiivne täisarv)
    sõna: otsitav sõna
```

### Ülesanne 31 Sulghäälikud (2003/2004)

Kirjutada programm, mis etteantud sõnade puhul moodustab sõne, mis koosneb kõigi sõnade esimestest sulghäälikutest (k, p, t, g, b, d). Kui sõnas sulghäälikuid pole, siis sellest sõnast ei võeta midagi. Näiteks sõnade *arbuus*, *kalad*, *armastus*, *vesi* ja *probleem* puhul on tulemuseks *bktp*.

Programmi sisendi etteandmiseks olgu kaks võimalust: käsurealt või tekstifailist. Kui sõnad on antud käsurealt, siis kasutatakse neid. Vastasel korral loetakse sõnad failist *sõnad.txt*.

Programmis peavad olema järgmised meetodid.

- a) Meetod *sulg*, mis etteantud sõnejärjendi puhul leiab ja tagastab sõne, mis koosneb kõigi sõnade esimestest sulghäälikutest.
- b) Meetod *failist*, mis tagastab failis olevatest sõnedest koosneva sõnejärjendi. Failis on iga sõne eraldi real.
- c) Meetod *väljasta*, mis väljastab programmi töö tulemused ekraanile kujul
 

Sõnade arbuus, kala, armastus, vesi ja probleem puhul  
on tulemuseks bktp.
- d) Peameetodis väljastada lühiselgitus programmi töö kohta, analüüsida käsurealisendit, kui see on olemas, vajadusel rakendada meetodit *failist* ning lõpuks rakendada meetodit *väljasta* programmi töö tulemuse näitamiseks.

### Ülesanne 32 Sõnade kombineerimine (2003/2004)

Koostada programm, mis moodustab käsurealt ette antud sõnadest kõikvõimalikud kahe- ja kolmeosalised liitsõnad (arvestamata sõnade tähenduste kokkusobivust või seda, kas sõnadel on üldse tähendus mingis keeles). Etteantud sõnu võib olla kaks või rohkem. Iga sõna võib ühes liitsõnas esineda ülimalt üks kord. Programm peab väljastama tulemuse nii ekraanile kui ka faili *liitsõnad.txt*.

Kui käsurealt ette antud sõnu on vähem kui kaks, siis väljastatakse kasutusjuhendi ja lõpetatakse programmi töö.

Näiteks käsu

```
java Liitsõnad karu kuld kell
```

korral on faili *liitsõnad.txt* sisu pärast programmi töö lõppu järgmine:



*kuldkell*  
*kellkuld*  
*karukell*  
*kellkaru*  
*karukuld*  
*kuldkaru*  
*karukuldkell*  
*karukellkuld*  
*kuldkarukell*  
*kuldkellkaru*  
*kellkarukuld*  
*kellkuldkaru*

### Ülesanne 33 Sõne pistmine teise sõne keskele (2003/2004)

Kirjutada programm, mis kahe erineva pikkusega sõne puhul moodustab sõne, milles pikema sõne võimalikult keskmiste sümbolite vahele on paigutatud lühem sõne. Näiteks sõnede *kala* ja *probleem* puhul on tulemuseks *probkalaleem*.

Programmi sisendi etteandmiseks olgu kolm võimalust: käsurealt, tekstifailist või klaviatuurilt. Kui sõned on antud ette käsurealt, siis kasutatakse neid. Vastasel korral laseb programm kasutajal valida, kas lugeda sõned failist *sõnad.txt* või klaviatuurilt.

Programmis peavad olema järgmised meetodid.

- Meetod *keskele*, mis leiab ja tagastab kahe etteantud sõne puhul sõne, kus pikema sõne võimalikult keskmiste sümbolite vahele on paigutatud lühem sõne.
- Meetodid *failist* ja *klaviatuurilt*, mis töötlevad sisendit. Mõlemad peavad tagastama kahest sõnest koosneva sõnejärjendi. Meetodis *failist* loetakse sõned faili *sõnad.txt* kahelt esimeselt realt (failis on iga sõne eraldi real). Meetodis *klaviatuurilt* loetakse sõned klaviatuurilt, eraldajaks on tühik.
- Meetod *väljasta*, mis väljastab programmi töö tulemused ekraanile, näiteks  
**Sõnede kala ja probleem puhul saame sõne probkalaleem.**
- Peameetodis väljastada lühiselgitus programmi töö kohta, analüüsida käsureasisendit, kui see on olemas; vajadusel rakendada meetodeid *klaviatuurilt* ja *failist* ning lõpuks rakendada meetodit *väljasta* programmi töö tulemuse kuvamiseks.

### Ülesanne 34 Kaas- ja täishäälikute loendamine (2004/2005)

Kirjutada programm, mis loeb käsurealt sõnejärjendi, väljastab selle ekraanile ja leiab sõnede hulgast need, milles on kaashäälikuid rohkem kui täishäälikuid. Eeldatakse, et sõned koosnevad ainult täis- ja kaashäälikutest, kuid võivad sisaldada

nii suur- kui ka väiketähti. Kriteeriumile vastavad sõned väljastada ekraanile koos sõnes sisalduvate täishäälikute ja kaashäälikute arvuga.

Kindlasti peavad programmis olema järgmised meetodid.

- a) Meetod, mis saab ette sõnejärjendi ning väljastab selle ekraanile.
- b) Meetod, mis etteantud sõne korral leiab selles sisalduvate täishäälikute arvu.
- c) Meetod, mis saab ette sõnejärjendi ja analüüsib tema elemente täishäälikute ja kaashäälikute arvu seisukohalt. Kui sõnes on kaashäälikuid rohkem kui täishäälikuid, siis väljastab meetod sõne koos täis- ja kaashäälikute arvuga ekraanile,

Eesti keele täishäälikud on a, e, i, o, u, õ, ä, ö, ü.

### Ülesanne 35                      Arvude faili summa                      (2004/2005)

Faili arvud.txt igal real asub parajasti üks arv. Arve on vähem kui 100. Täis- ja murdosa eraldajana on vahel kasutatud koma ja vahel punkti.

Kirjutada programm, mis loeb failist arvud ja leiab nende summa, teisendades arvud vajadusel sobivale kujule. Arvud ja nende summa väljastada ka ekraanile.

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis teisendab sõnena antud reaalarvu *double*-tüüpi arvuks, asendades selles vajadusel eelnevalt koma punktiga.
- b) Meetod, mis loeb failist andmed reaalarvujärjendisse.
- c) Meetod, mis arvutab reaalarvujärjendi summa.
- d) Meetod, mis väljastab reaalarvujärjendi ja selle summa ekraanile.
- e) Peameetod, milles rakendatakse eespool loetletud meetodeid.

### Ülesanne 36                      Kaugushüppe tulemused                      (2004/2005)

Failis kaugus.txt on tabel kaugushüppevõistluse tulemustega kujul

*Jaan 4.50*

*Karl 4.80*

*Madis 4.10*

Nime ja tulemuse eraldajaks on tabulaator (Javas väljendatav kui \t). Eeldame, et kõik tulemused on kujul *x.xx* (ebaõnnestunud katse 0.00). Tabelis võib tulemusi olla 1 kuni 20 ja iga inimese kohta on üks tulemus.

Kirjutada programm, mis väljastab ekraanile kõigi osalejate nimed ühes reas (ilma tulemusteta), leiab võistluse võitja ning väljastab tema nime ja tulemuse järgmisel real.

Osalejad: Jaan, Karl, Madis.

Võitis Karl tulemusega 4.80.

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis loeb failist `kaugus.txt` andmed sõnejärjendisse.
- b) Meetod võitja väljaselgitamiseks.
- c) Peameetod, milles rakendatakse kahte esimest meetodit ja väljastatakse ekraanile osalejate nimed ning võitja koos tema tulemusega.

### Ülesanne 37 Leheküljenumbrite intervallid (2004/2005)

Kirjutada programm, mis loeb käsurealt printitavate lehekülgede numbrid sellisel kujul, nagu tekstiredaktorist printimisel tuleb anda, näiteks „3, 5-7, 10, 14 -18“, ja väljastab kõikide printitavate lehekülgede numbrid üksteise alla. Võib eeldada, et numbrid on kasvavas järjekorras ja intervallid omavahel ei lõiku. Eraldajate ümber võib mõlemal pool olla suvaline arv tühikuid.

Ülaltoodud näite puhul on väljundiks

```
3
5
6
7
10
14
15
16
17
18
```

Programm peab sisaldama meetodit, mis saab ette täisarvude lõigu või ühe täisarvu (sõnena, koos võimalike tühikutega), ja väljastama selles lõigus sisalduvad arvud või vastavalt sellele arvu.

NB! Kui käsureasisend sisaldab tühikuid, siis tuleb talle jutumärgid ümber panna. Näide programmi käivitamisest:

```
java Leheküljed "3, 5-7, 10, 14 -18"
```

### Ülesanne 38 Kuupäeva lahtikirjutus (2004/2005)

Kirjutada programm, mis saab käsurealt ühe või enam kuupäeva kujul `pp.kk.aaaa` või `pp.kk.aa` ja väljastab need ekraanile kujul

```
kuupäev: pp, kuu: kuunimi, aasta: aaaa
```

Võib eeldada, et kuupäevad on antud korrektselt ja lühikujul kuupäeva aastaarvu eest ärajäetud numbrid on 2 ja 0. Väljundis mitte kirjutada ühekohalise päeva-  
numbri ette nulli.

Näiteks sisendile

```
java Kuupäev 28.12.04 01.02.1996 15.09.1765
```

```
vastab väljund
```

päevanumber: 28, kuu: detsember, aasta: 2004

päevanumber: 1, kuu: veebruar, aasta: 1996

päevanumber: 15, kuu: september, aasta: 1765

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis saab ette kuupäeva kujul *pp.kk.aaaa* või *pp.kk.aa* ja tagastab kolmeelemendilise täisarvujärjendi, mille esimene element märgib päeva, teine kuud ja kolmas aastat.
- b) Meetod, mis saab ette eelmises punktis kirjeldatud kolmeelemendilise täisarvujärjendi ja väljastab selle ekraanile kujul  
kuupäev: pp, kuu: kuunimi, aasta: aaaa
- c) Peameetod, milles rakendatakse eespool loetletud meetodeid.

### Ülesanne 39 Tsensuur (2004/2005)

Failis *tsenseeritav.txt* on sõnad, mis on eraldatud tühikute ja reavahetustega. Sõnad koosnevad ainult täppideta väiketähtedest. Kirjutada programm, mis loeb käsurealt „keelatud“ sõnad ja eemaldab faili sõnade hulgast kõik „keelatud“ sõnade esinemised. Loodav tulemusfail peab olema nimega *tsenseeritud.txt*.

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis moodustab failis sisalduvatest sõnadest sõnejärjendi.
- b) Meetod, mis kontrollib, kas etteantud sõna esineb järjendis.
- c) Meetod, mis kirjutab sõna faili.
- d) Peameetod, milles rakendatakse eespool loetletud meetodeid.

### Ülesanne 40 Jooksuvõistluse tulemused (2004/2005)

Failis *jooks.txt* on tabel 10000 meetri jooksu aegadega kujul

*Jaak 31.10,78*

*Peeter 29.30,15*

*Mati 26.14,97*

*Kalev 27.40,61*

Eeldame, et kõik tulemused on kujul *aa.bb,cc*, kus *aa* näitab minuteid, *bb* sekundeid ja *cc* sekundisajandikke. Tabelis on kuni 25 tulemust ja iga inimese kohta on üks tulemus.

Kirjutada programm, mis väljastab ekraanile kõigi osalejate nimed eri ridadel ja nende tulemuste võrdlused Eesti rekordiga 27.40,61.

Tulemuse vormi näide:

Jaak: Eesti rekordist aeglasem

Peeter: Eesti rekordist aeglasem

Mati: Eesti rekordist kiirem

Kalev: Eesti rekordi kordamine

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis moodustab failis sisalduvatest andmetest sõnejärjendi.
- b) Meetod, mis teisendab eelpool kirjeldatud kujul antud aja sajandikeks. (Sajandikena esitatud aegu on hõlpsam võrrelda.)
- c) Meetod, mis võrdleb tulemust Eesti rekordiga.
- d) Peameetod, milles rakendatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemused nõutud vormis.

## Ülesanne 41 Ühisprefiksiga telefoninumbrid (2004/2005)

Failis `telefon.txt` on telefoninumbrid kujul

*Jaak: 723 456*

*Peeter: 555 0123*

*Mati: 5556 1234*

*Juhan: 5550 4321*

Eeldame, et kõik telefoninumbrid on kujul `xxx xxxx` või `xxxx xxxx`. Failis võivad olla 1 kuni 30 inimese telefoninumbrid, iga inimese kohta on üks telefoninumber.

Kirjutada programm, mis loeb käsurealt telefoninumbri alguse (ilma tühikuteta, 1—5 numbrit) ja väljastab ekraanile kõigi nende inimeste telefoninumbrid, kelle telefoninumber vastavalt algab.

Näiteks ülaltoodud faili ja käsurea

```
java Telefon 5550
```

puhul ilmub ekraanile

```
Numbritega 5550 algavad järgmised telefoninumbrid:
```

```
Peeter: 555 0123
```

```
Juhan: 5550 4321
```

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis eemaldab telefoninumbri tühiku.
- b) Meetod, mis kontrollib, kas tühikuta telefoninumber algab antud numbritega.
- c) Peameetod, milles rakendatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemus nõutud kujul.

## Ülesanne 42 Premeerimine (2004/2005)

Failis `palk.txt` on kirjas kuupalgad kujul

*Peeter: 6590*

*Jaak: 5000*

*Juhan: 10000*

*Mati: 12900*

Eeldame, et kõik palgad on kujul *xxxx* või *xxxxx*. Failis võivad olla 1 kuni 20 inimese palgad, iga inimene esineb tabelis üks kord.

Kirjutada programm, mis loeb käsurealt preemiaprotsendi (mis on eeldatavalt kujul  $x\%$  või  $xx\%$ ) ja väljastab ekraanile kõigi nende inimeste kuusissetulekud ümardatuna täisarvuks. Kuusissetulek koosneb palgast ja preemiast, mis võrdub etteantud protsendiga palgast. Näiteks  $25\%$  arvust 5000 on 1250, järelikult on ülaltoodud andmete põhjal Jaagu kuusissetulek  $5000 + 1250 = 6250$ .

Ülaltoodud näitefaili ja käsurea

```
java Preemia 25%
```

puhul ilmub ekraanile

Kuusissetulekud on järgmised:

Peeter: 8238

Jaak: 6250

Juhan: 12500

Mati: 16125

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis eemaldab preemiaprotsendilt protsendimärgi ja tagastab järelejäänud täisarvu.
- b) Meetod, mille parameetriteks on palk ja preemiaprotsent (täisarvudena) ning mis tagastab sissetuleku ümardatuna täisarvuks.
- c) Peameetod, milles rakendatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemus nõutud kujul.

## Ülesanne 43

## Allahindlus

(2004/2005)

Failis `teler.txt` on kirjas telerite alghinnad kujul

*Telcomp: 6590*

*Teleur: 5000*

*Eurtel: 10000*

*Telek: 12900*

Eeldame, et kõik hinnad on kujul *xxxx* või *xxxxx*. Failis võivad olla 1 kuni 15 teleri hinnad, iga teler esineb tabelis üks kord.

Kirjutada programm, mis loeb käsurealt allahindlusprotsendi (mis on eeldatavalt kujul  $x\%$  või  $xx\%$ ) ja väljastab ekraanile kõigi telerite lõpphinnad ümardatuna täisarvuks. Lõpphind kujuneb alghinnast vastava protsendi lahutamisel. Näiteks  $25\%$  arvust 5000 on 1250, järelikult on ülaltoodud andmete põhjal Teleuri hind  $5000 - 1250 = 3750$ .

Ülaltoodud näitefaili ja käsurea

```
java Allahindlus 25%
```

puhul ilmub ekraanile

Telerite lõpphinnad on järgmised:

Telcomp: 4943

Teleur: 3750

Eurtel: 7500

Telek: 9675

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis eemaldab allahindlusprotsendilt protsendimärgi ja tagastab järelejäänud täisarvu.
- b) Meetod, mille parameetriteks on alghind ja allahindlusprotsent (täisarvudena) ning mis tagastab lõpphinna ümardatuna täisarvuks.
- c) Peameetod, milles rakendatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemus nõutud kujul.

## Ülesanne 44 Konverentsinimede lühendid (2005/2006)

Failis `konverentsid.txt` on teaduskonverentside nimed, näiteks

*International Conference on Computer Games 1997*

*Dark Side of the Moon 2004*

*Juulikuise lume uurimise konverents 1998*

Eeldame, et iga nimi koosneb sõnadest, mille järel on neljakohaline aastaarv. Failis võib olla 1 kuni 30 konverentsinime, iga nimi asub eraldi real.

Kirjutada programm, mis väljastab ekraanile kõigi konverentsinimede lühendid, võttes igast sõnast esitähe (suurtähena) ja aastaarvu kaks viimast numbrit ning asetades nende vahele ülakoma.

Näiteks ülaltoodud faili puhul ilmub ekraanile

ICCG'97

DSOTM'04

JLUK'98

Programm peab sisaldama vähemalt järgmisi meetodeid.

- a) Meetod, mis tagastab etteantud sõna esimese tähe suurtähena.
- b) Meetod, mis tagastab etteantud sõne kaks viimast sümbolit sõnena.
- c) Peameetod, mille täitmisel kasutatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemus nõutud kujul.

## Ülesanne 45 Veebiaadresside genereerimine (2005/2006)

Luaa tekstiredaktoriga fail `asutused.txt`, milles on asutuste ja organisatsioonide nimed, kusjuures iga nimi asub eraldi real, näiteks

*Universitas Tartuensis*

*Tartu Roheline Brigaad*

*Kultuuriklubi Tartu Vaim*

Kirjutada programm, mis väljastab ekraanile vastavad veebiaadressid, mis moodustatakse järgmiselt.

- Veebiaadress algab osaga **www**.
- Kui nimes ei sisaldu eraldi sõnana *Tartu*, siis aadressi teine komponent on lühend, mis sisaldab asutuse nime kõigi sõnade esitähti väiketähtedena.
- Kui nimes sisaldub eraldi sõnana *Tartu*, siis aadressi teine komponent on nimes sõnale *Tartu* vahetult järgnev sõna ja kolmas komponent **tartu**.
- Veebiaadress lõpeb osaga **ee**.
- Kõikide komponentide vahel on punktid.

Näiteks ülaltoodud faili puhul ilmub ekraanile

`www.ut.ee`

`www.roheline.tartu.ee`

`www.vaim.tartu.ee`

Programm peab sisaldama vähemalt järgmisi meetodeid.

- Meetod, mis kontrollib, kas nimes sisaldub eraldi sõnana *Tartu*.
- Meetod, mis moodustab nime lühendi, võttes iga sõna esitähe (väiketähena).
- Meetod, mis lisab aadressi algusesse **www**. ja lõppu **.ee**.
- Peameetod, mille täitmisel kasutatakse eespool loetletud meetodeid ja väljastatakse ekraanile tulemus nõutud kujul.

## Ülesanne 46 Sünonüümid (2005/2006)

Faili `sünonüümid.txt` igas reas on komplekt omavahel sünonüümseid (samatähenduslikke) sõnu, näiteks

*karu, mesikäpp, mõmmik, karumõmm, ott, karuott  
koer, kutsu, kutsa, kuts, koeranäss, peni, krants*

Ridades on naabersõnad eraldatud koma ja tühiku(te)ga.

Kirjutada programm, mis saab käsurealt ette sõna ja väljastab selle sünonüümid, mis sisalduvad failis `sünonüümid.txt`. Otsitavat sõna väljundis ei korrata. Otsing ei arvesta suurte ja väikeste tähtede erinevust.

### Näide

Kui käsurealt antakse ette sõna *Peni*, siis ekraanile väljastatakse

Sõna 'peni' sünonüümid on:

koer

kutsu

kutsa

kuts

koeranäss

krants



Kui antud sõna (nt *Java*) ei esine selles failis, siis väljastatakse vastav teade, näiteks

Sõnal 'java' ei ole sünonüüme.

## Ülesanne 47                      Sõnaline avaldis                      (2005/2006)

Faili `avaldised.txt` igas reas on üks eesti keeles kirjutatud aritmeetiline avaldis, näiteks

*kaks pluss seitse miinus kolm*  
*neli miinus üks pluss kuus pluss üheksa*

Avaldistes on ainult ühekohalised positiivsed arvud (1–9) ja tehtemärkidest ainult pluss ja miinus. Kirjutada programm, mis võtab failist `avaldised.txt` avaldised, arvutab nende väärtused ning kirjutab vastused arvudena faili `vastused.txt`.

Näiteks ülaltoodud avaldiste puhul kirjutatakse faili `vastused.txt` read

*6*  
*18*

## Ülesanne 48      Tähtede asendamine sõnadega      (2005/2006)

Faili `sõnad.txt` igas reas on üks sõna, näiteks

*berta*  
*anna*  
*dora*  
*umberto*  
*tere*

Failis ei esine kahte sama algustähega sõna.

Kirjutada programm, mis loeb käsurealt sõna ja kirjutab selle ekraanile kahel viisil. Kõigepealt nii nagu sõna käsureale kirjutati ning seejärel nii, et selle sõna kõik tähed asendatakse vastavalt esitähle failist `sõnad.txt` võetud sõnadega, ühendades need sidekriipsudega. Kui vastavat sõna failis pole, siis kirjutatakse selle asemel kolm küsimärki. Suuri ja väikesi tähti ei eristata.

Kui käsurealt antakse ette näiteks sõna *Tartu*, siis väljastatakse ekraanile

Sõnast *Tartu* saadakse  
`tere-anna-???-tere-umberto`

## Ülesanne 49                      Telefoninumbri otsing                      (2005/2006)

Faili `telefonid.txt` igal real on erinev eesnimi ja telefoninumber, näiteks

*Arnold 6316202*  
*Ene 3726316301*  
*Andrus 6935555*

Nimi ja telefoninumber on teineteisest eraldatud ühe või enama tühikuga. Telefoninumbriid võivad olla erineva pikkusega.

Kirjutada programm, mis vastavalt käsureaargumendina etteantud nimele otsib failist `telefonid.txt` telefoninumbri ja väljastab selle. Otsingul suuri ja väikesi tähti ei eristata. Nimi väljastatakse läbivalt suurtähelisena.

Näiteks kui käsurealt antakse ette nimi *andrus*, siis ekraanile väljastatakse  
Isiku ANDRUS telefoninumber on 6935555.

Kui isikut ei leita, siis väljastatakse vastav teade, näiteks

Isiku ALLAR telefoninumbrit ei leitud.

## Ülesanne 50                      Bussiinfo                      (2006/2007)

Faili `linnad.txt` igas reas on ühe Eesti linna nimi ja arv, mitu korda päevas buss Tartust sinna sõidab, näiteks

*Elva 10*  
*Tallinn 20*  
*Narva 2*

Linna ja arvu vahel on täpselt üks tühik. Kirjutada programm, mis küsib kasutajalt (konsooldialoogi või dialoogiakna kaudu), millise linna kohta infot soovitakse ja seejärel väljastab ekraanile, mitu korda päevas sellesse linna Tartust buss sõidab. Kui sellist linna failis pole, siis väljastatakse vastav teade.

### Näide

Ülaltoodud faili ja sisestuse *Narva* puhul ilmub ekraanile teade

Linna Narva suunduvate busside arv on 2.

Sisestuse *Kärdla* puhul aga saame teate

Sihtlinna Kärdla failis pole.

## Ülesanne 51                      Telefoninumbrid klahvidelt                      (2006/2007)

Faili `klahvid.txt` igas reas on number ja tähed, mis on ühel telefoniklahvil:

2: *abc*  
3: *def*  
4: *ghi*  
5: *jkl*  
6: *mno*  
7: *pqrs*  
8: *tuv*  
9: *wxyz*

Numbri ja arvu vahel on koolon ja tühik(ud). Kirjutada programm, mis küsib kasutajalt (konsooldialoogi või dialoogiakna kaudu) ühekaupa kolm „telefoninumbrit“,

milles võib olla nii tähti kui ka numbreid. Programm kirjutab faili `numbrid.txt` eraldi ridadele vastavad telefoninumbriid numbrilisel kujul.

Näiteks sisestuste *me1ma0b*, *7esko01* ja *kum1* puhul kirjutatakse faili

*6316202*

*7375601*

*5861*

## Ülesanne 52                      Töötundide arvestus                      (2006/2007)

Faili `nimed.txt` igas reas on üks eesnimi ja täisarv, mitu tundi on inimene töötanud (nimed võivad korduda, siis vastavad tunnid summeeritakse), näiteks

*Birute 4*

*Ene 10*

*Volli 8*

*Birute 5*

Nime ja arvu vahel on vähemalt üks tühik. Kirjutada programm, mis küsib kasutajalt (konsooldialoogi või dialoogiakna kaudu), millise nime kohta infot soovitakse, ja seejärel väljastab ekraanile, mitu tundi kokku see inimene on töötanud. Kui sellist nime failis pole, siis väljastatakse vastav teade.

### Näide

Ülaltoodud faili ja sisestuse *Birute* puhul ilmub ekraanile teade

Birute: 9 tundi.

Ülaltoodud faili ja sisestuse *Jaak* puhul ilmub ekraanile teade

Nime Jaak failis pole.

## Ülesanne 53                      Hinnakirja sisestamine                      (2006/2007)

Faili `hinnakiri.txt` igas reas peab olema info ühe kauba kohta: kauba nimetus (üks või kaks sõna), kauba kood (viiekohaline positiivne täisarv) ja hind, näiteks

*kamm 23416 6.40*

*sinine mängukaru 58692 76.99*

*sädelev pliiats 63928 30*

Koostada programm, mille abil saab sellise faili luua ja sisuga täita (konsooldialoogi või dialoogiakna kaudu). Sisestajalt küsitakse nimetust, koodi ja hinda eraldi. Kui sisestatakse kauba koodi või hinda, siis peab küsimuses sisalduma kauba nimetus. Võib eeldada, et sisestaja sisestamisel vigu ei tee.

Sisestusseansi lõpetab tühja rea sisestamine. Kui kauba andmed on seejuures poolikud, siis neid andmeid faili ei lisata. Töö lõpus kuvatakse ekraanile, mitme kauba andmed sisestati.

**Ülesanne 54                      Numbrite liitmise test                      (2006/2007)**

Koostada programm, mis kuvab liitmistehteid ja küsib (konsooldialoogi või dialoogiakna kaudu) kasutajalt summasid. Korruga küsitakse ühte kahe ühekohalise positiivse arvu summat (nt  $3 + 5 =$ ). Liidetavad valib programm juhuslikult. Võib eeldada, et vastuseks sisestatakse ainult arve. Tehete küsimise lõpetab tühja rea sisestamine ja seda tehet arvesse ei võeta.

Küsitud tehted koos kasutaja antud vastustega lisatakse failidesse nii, et õigesti vastatud tehted lisanduvad faili `õiged.txt` ja valesti vastatud faili `valed.txt`.

Nii võivad näiteks failis `õiged.txt` olla read

$$3 + 4 = 7$$

$$5 + 5 = 10$$

$$2 + 4 = 6$$

failis `valed.txt` aga read

$$3 + 5 = 2$$

$$8 + 7 = 9$$

# Graafika

Käesoleva peatüki ülesannetes tuleb kirjutada programm, mis joonistab nõutud pildi. Iga ülesande puhul on toodud väljundi näidis. On ülesandeid, kus algandmed sobib programmis esitada nt järjendina, on ülesandeid, kus olulisel kohal on joonistusakna mõõtmete arvestamine. Paljude ülesannete puhul on ette nähtud mingi parameetri väärtuse juhuslik genereerimine.

Võrreldes tekstide originaalkujudega, on ära jäetud lõigud, mis mainisid, et programmi kirjutamisel võib aluseks võtta J. Kiho *Java programmeerimise aabitsa* (Tartu 2001, 1. tr; 2002, 2. tr) programmiklassid *Jooniseraam*, *Tahvel* ja *Joonis*, millest muuta tulnuks peamiselt klassi *Joonis* meetodit *joonistada*. Samuti on ära jäetud mõne ülesande juures esinenud täpsed juhised, kuidas mõnda värvi või fonti vms saab kasutada.

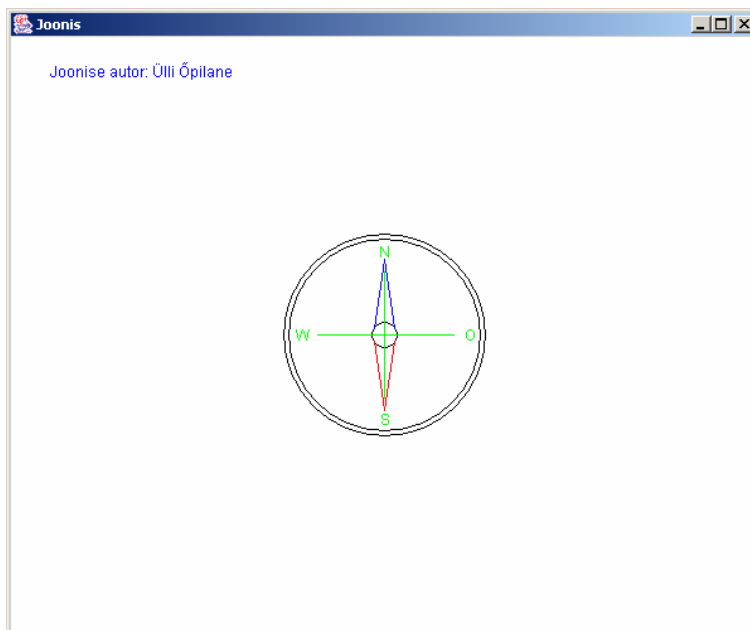
## Ülesanne 55

## Kompass

(2003/2004)

Kirjutada programm, mis kuvab antud joonistusakna keskele kompassi kujutise. Kompassi nõela põhjasuunaline ots joonistada sinist värvi lõikude abil, lõunasuunaline ots aga punast värvi lõikude abil. Joonise ülejäänud elementide värvi võib valida meelevaldselt. Joonisele kirjutada ka autori nimi.

### Väljundi näidis

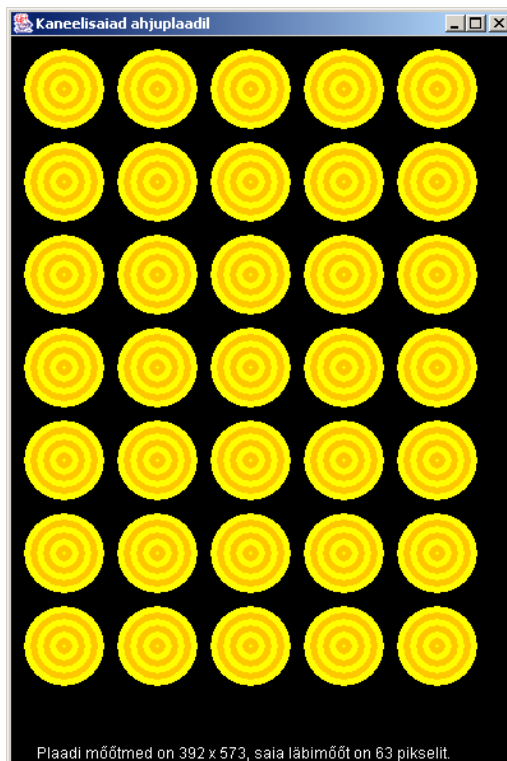


**Ülesanne 56****Kaneelisaiad****(2003/2004)**

Kirjutada programm, mis joonistab musta ahjuplaadi ja sellele nii palju ühesuguseid kaneelisaiu, kui mahub, arvestades järgmisi tingimusi.

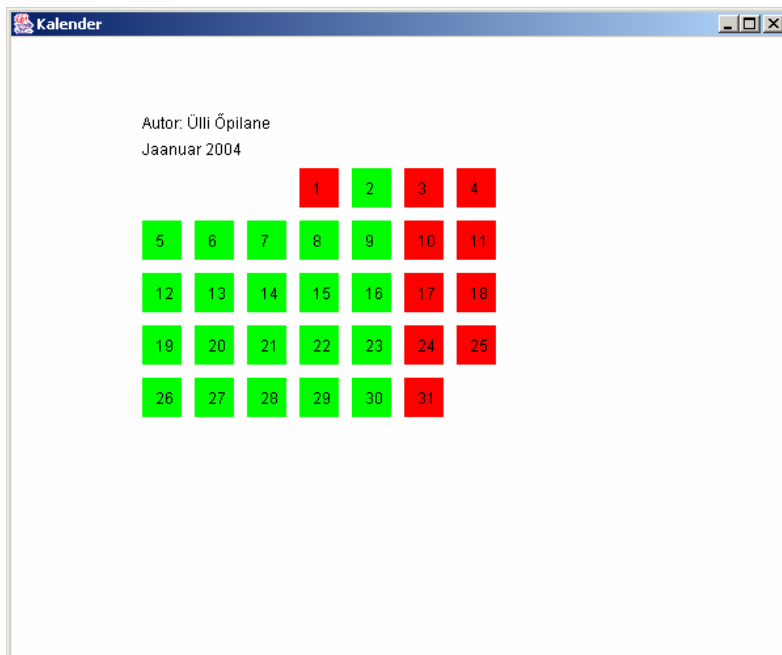
- Kõik kaneelisaiad on ühesuurused ning nendes on vaheldumisi sai ja kaneel ehk kollane ja oranž kiht, kusjuures kõige välimine kiht on sai.
- Plaadi mõõtmed on programmis ette antud (programmeerija võib need vabalt valida)
- Saia läbimõõt genereeritakse juhuslikult, nii et see oleks vahemikus 40—120 pikslit.
- Saiade vahele ning plaadi äärtesse peab jääma „kerkimisruumi“ vähemalt 10 pikslit.
- Plaadi alumises servas näidatakse plaadi ja saia mõõtmed.

Väljundi näidis

**Ülesanne 57****Kalender****(2003/2004)**

Kirjutada programm, mis joonistab 2004. a. jaanuarikuu kalendri, milles puhkepäevad on teist värvi. Joonisele kirjutada ka autori nimi.

## Väljundi näidis



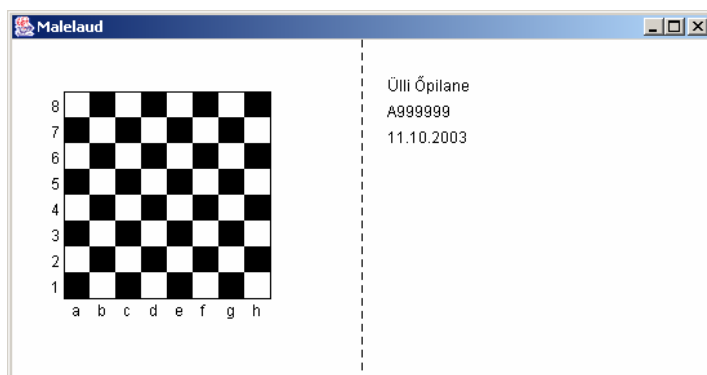
## Ülesanne 58

## Malelaud

(2003/2004)

Kirjutada programm, mis joonistab järgmise pildi. Joonistuspiirkond on vertikaalis jaotatud katkendjoonega kaheks võrdseks osaks, millest ühele poole on joonistatud malelaud pealtvaates koos laua välju tähistavate tähtede a–h ja numbritega 1–8 ning teisele poole on kirjutatud programmi autori nimi, matriklinumber ja kuupäev. Malelaua väljad peavad olema heledad ja tumedad (värvid valida ise).

## Väljundi näidis

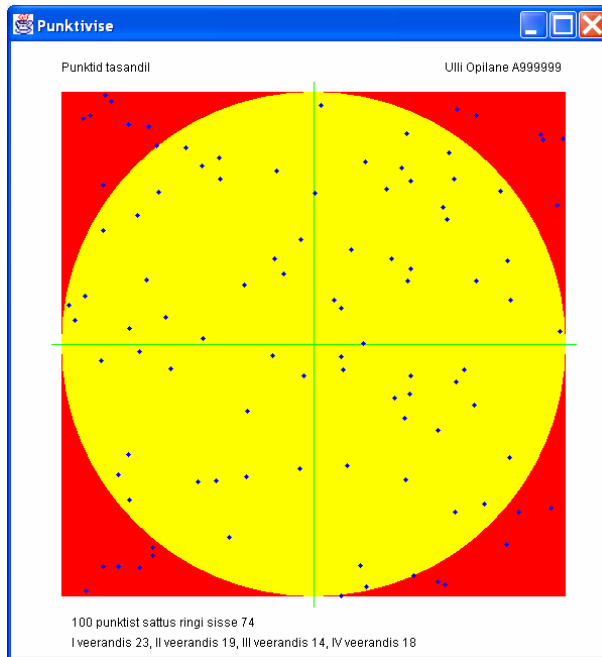


## Ülesanne 59 Punktivise (2003/2004)

Kirjutada programm, mis joonistab tasandile ruudu, selle sisse ringi, mis puutub ruudu külgi, ning ruudu külgede keskristsirged. Seejärel viskab programm ruudu sisse juhuslikult 100 punkti ning loeb kokku, mitu neist sattus ringi sisse, tuues seejuures eraldi välja, mitu punkti sattus I veerandisse ruudu keskpunkti suhtes, mitu punkti II veerandisse jne.

Joonise ülemisse vasakusse nurka väljastada joonist kirjeldav tekst (pealkiri) ja ülemisse vasakusse nurka autori nimi koos matriklinumbriga. Joonise allserva väljastada nõutav informatsioon: mitu punkti sattus ringi sisse ja nende punktide arvu jaotus veerandite lõikes. (Tegemist on Monte Carlo meetodiga, mille abil saab ligikaudselt leida arvu  $\pi$ .)

### Väljundi näidis



## Ülesanne 60 Eksamihinnete jaotus (2003/2004)

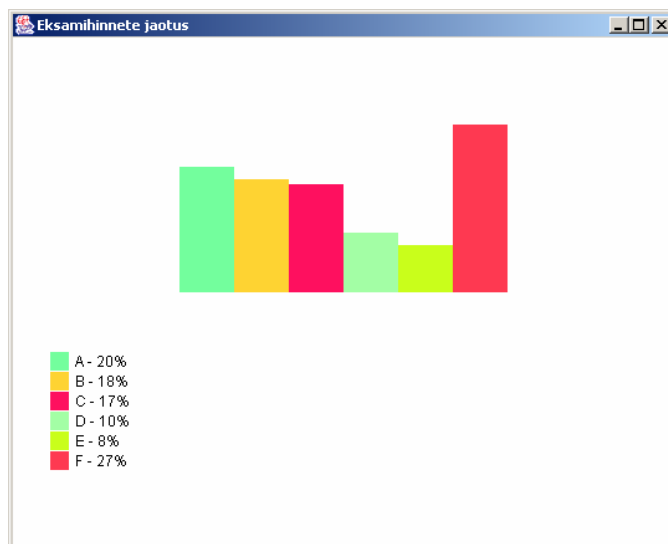
On antud kuueelemendiline täisarvujärjend eksamitulemustega (mitu üliõpilast sai hindeks A, mitu B jne). Näiteks järjendist  $\{21, 19, 18, 10, 8, 28\}$  selgub, et 21 üliõpilast said eksamihindeks A, 19 üliõpilast B, 18 üliõpilast C, 10 üliõpilast D, 8 üliõpilast E ja 28 üliõpilast F.

Antud järjendi põhjal joonistada tulpdiagramm eksamihinnete jaotuse kohta, kus iga tulp on erinevat värvi. Diagrammi alla paigutada ka legend, mis selgitab, milline värv diagrammil millisele hindele vastab ning mitu protsenti kõikidest eksami sooritanud üliõpilastest selle hinde said.



Tulpdiagrammi laius olgu pool joonistustahvli laiusest ja see olgu paigutatud horisontaalsuunas tahvli keskele. Vertikaalsuunas valida asukoht oma maitse järgi.

### Väljundi näidis



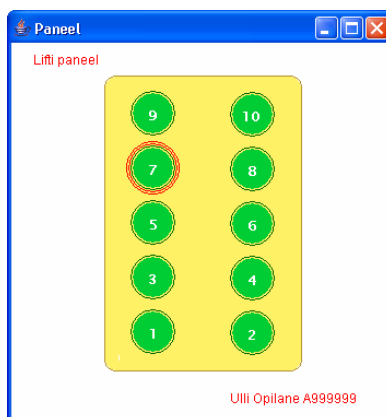
### Ülesanne 61

### Paneel

(2004/2005)

Kirjutada programm, mis joonistab ekraanile kümnekorruselise maja lifti juhtpaneeli. Programm peab paigutama nupud paneelil kahte tulpa ja tõstma esile juhulikult valitud korruse nupu. Joonise ülemisse vasakusse nurka väljastada joonist kirjeldav tekst ning alumisse paremasse nurka autori nimi koos matriklinumbriaga.

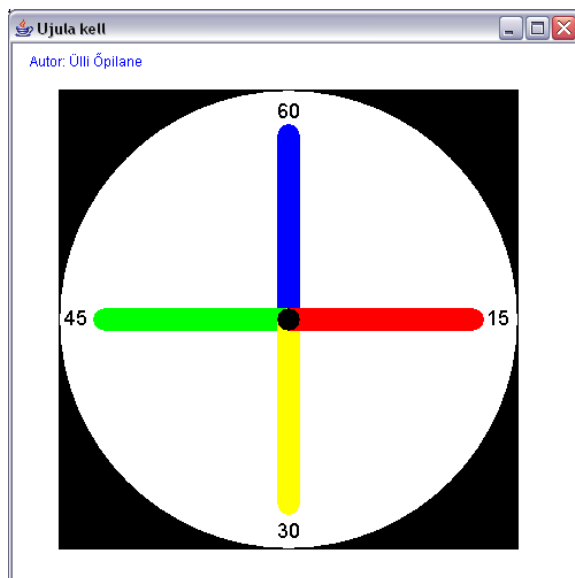
### Väljundi näidis



**Ülesanne 62** **Ujula kell** (2004/2005)

Kirjutada programm, mis joonistab ujula kella (mittetöötava). Kell koosneb ruudukujulisest alusest, mille peal asub nelja sekundiosutiga ring. Sekundiosutid paiknevad ristikujuliselt ja nende värvid on parempoolsest alates päripäeva lugedes punane, kollane, roheline ja sinine. Sekundiosutite juures peab olema kirjas ka sekundite arv: vastavalt 15, 30, 45 ja 60. Kella keskpunktis on must ring.

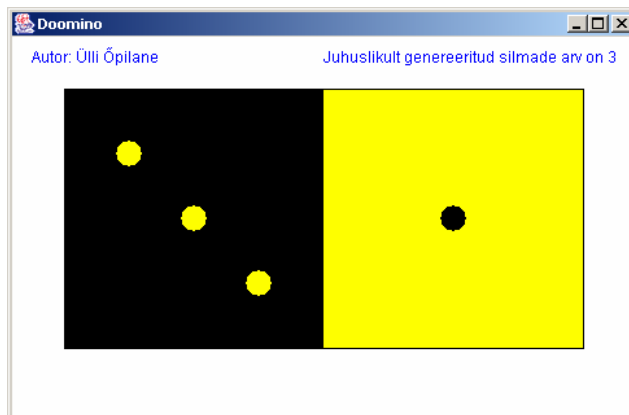
Joonise ülemisse vasakusse nurka kirjutada ka autori nimi.

**Väljundi näidis****Ülesanne 63** **Doomino** (2004/2005)

Kirjutada programm, mis joonistab doominonupu. Doominonupp koosneb heledast ja tumedast poolest (ruudust), millel on vastavalt teist värvi silmad (ringid). Ühel pool on üks silm, teise poole silmade arv, mis genereeritakse juhuslikult, on üks, kolm või viis.

Joonise ülemisse paremasse nurka kirjutada juhuslikult genereeritud silmade arv. Joonise ülemisse vasakusse nurka kirjutada ka autori nimi.

## Väljundi näidis



## Ülesanne 64

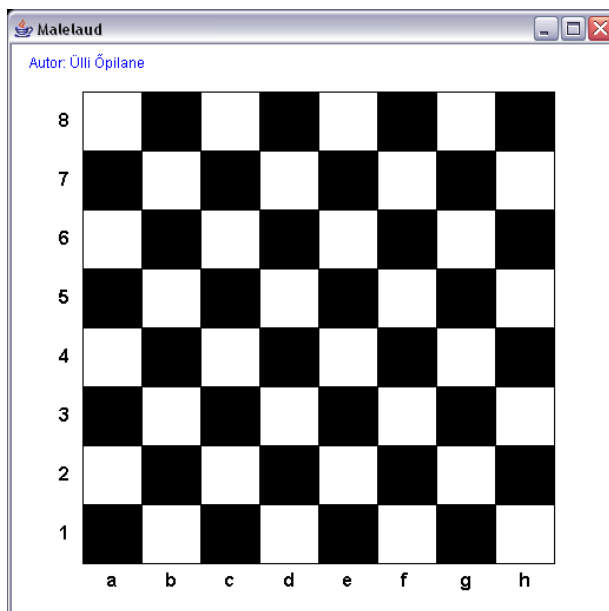
## Malelaud

(2004/2005)

Kirjutada programm, mis joonistab malelauda koos arvude ja tähtedega. Arvud peavad asuma vasakul ja suurenema alt üles, tähed aga peavad asuma all ja olema järjestatud tähestiku järjekorras vasakult paremale.

Joonise ülemisse vasakusse nurka kirjutada ka autori nimi.

## Väljundi näidis

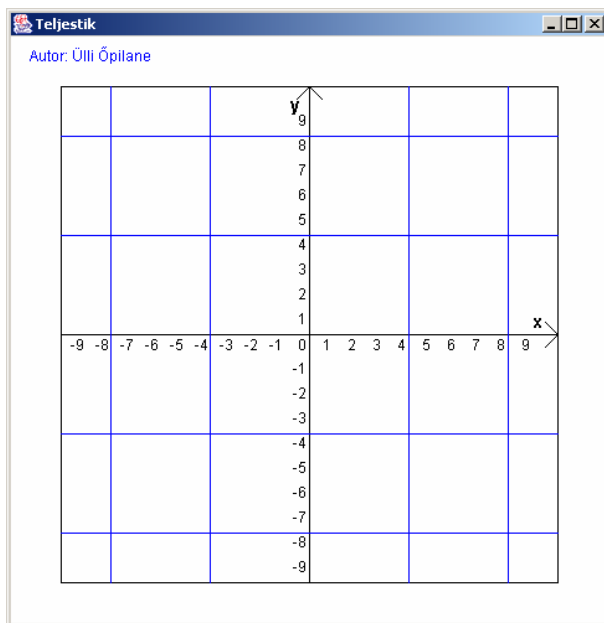


## Ülesanne 65 Teljestik (2004/2005)

Kirjutada programm, mis joonistab ruudu ja ruudu sisse teljestiku. Teljestiku alguspunkt (koordinaatidega  $(0,0)$ ) peab asuma ruudu keskel. Kumbki telg peab olema musta värvi ja lõppema noolega, mille juures on paksu kirjaga näidatud telje nimi ( $x$  või  $y$ ). Telgedel peavad olema märgitud täisarvuliste koordinaatidega punktid. Null kirjutada ainult üks kord. Neljaga jaguvate arvude juures peavad telgi lõikama sinist värvi abijooned.

Joonise ülemisse vasakusse nurka kirjutada ka autori nimi.

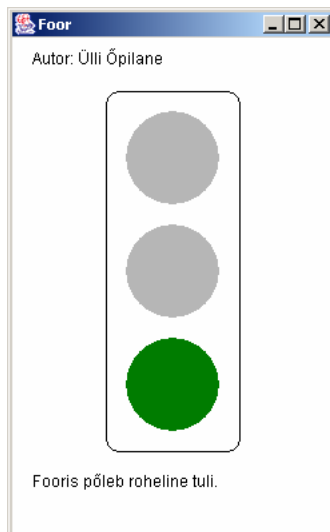
### Väljundi näidis



## Ülesanne 66 Foor (2004/2005)

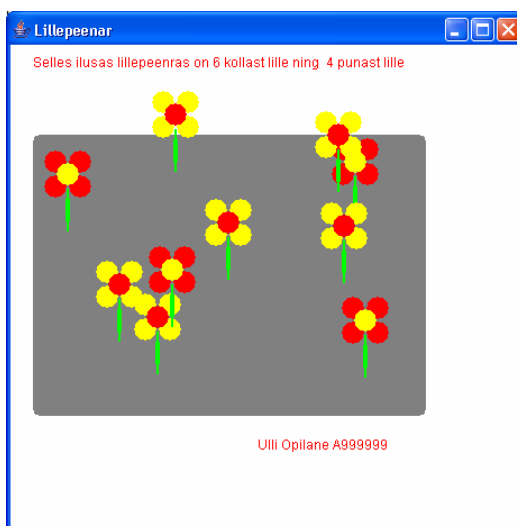
Kirjutada programm, mis joonistab valgusfoori. Valgusfoor on ümarate nurkadega ristkülik, milles on kolm tuld (ringi). Programm peab juhuslikult valima, milline tuli põleb (st on vastavalt punane, kollane või roheline), mittepõlevad tuled on hallid.

Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise alumisse serva kirjutada, milline tuli põleb.

**Väljundi näidis****Ülesanne 67****Lillepeenar****(2004/2005)**

Kirjutada programm, mis joonistab lillepeenra (ümarate nurkadega risküliku), milles kasvab 6 kollast lille ja 4 punast lille. Lilled paiknevad nii, et varre alumised otsad on peenras.

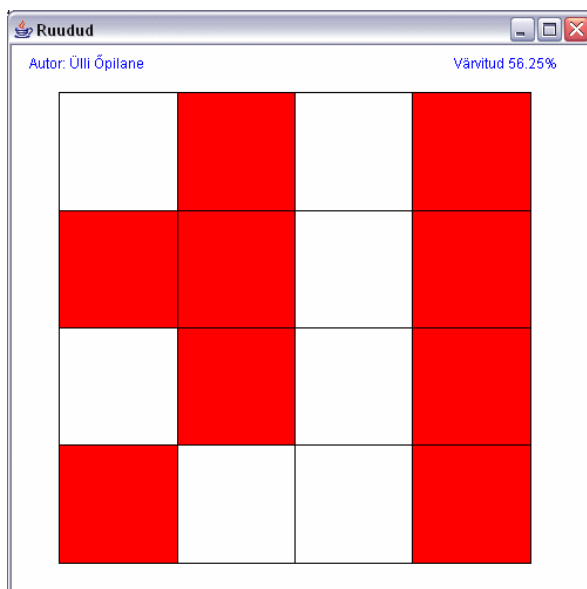
Joonise ülaserva kirjutada, mitu punast ja mitu kollast lille peenras on. Joonise alaserva kirjutada autori nimi ja matriklinumber.

**Väljundi näidis**

**Ülesanne 68** **Ruudud** (2004/2005)

Kirjutada programm, mis joonistab ruudu ning jagab selle 16 võrdseks ruuduks. Iga väikese ruudu jaoks genereeritakse juhuslik arv nullist üheni. Kui genereeritud arv on suurem kui 0,5, siis värvitakse vastav ruut punaseks.

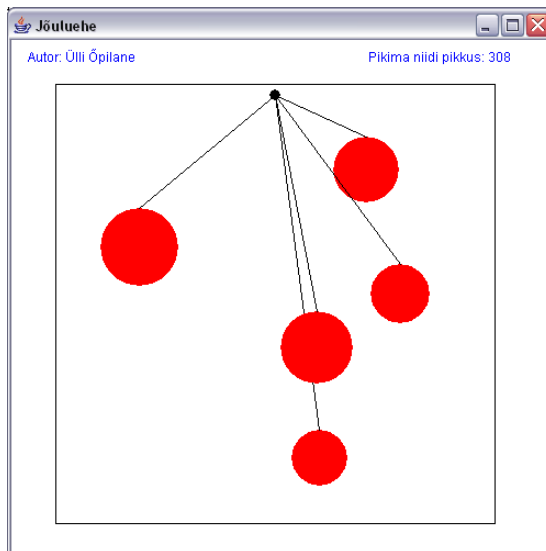
Joonise ülemisse paremasse nurka kirjutada, mitu protsenti suurest ruudust on värvitud. Joonise ülemisse vasakusse nurka kirjutada autori nimi.

**Väljundi näidis****Ülesanne 69** **Jõuluehe** (2004/2005)

Kirjutada programm, mis joonistab ruudu ja ruudu sisse jõuluehte. Jõuluehe koosneb viiest erineva diameetriga punasest kuulikesest (ringist), mis on niitidega (sirlõikudega) ühendatud sõlmkohaga (väiksem must ring). Kuulikesed peavad olema programmi poolt juhuslikult genereeritud arv vahemikus 50–75. Kuulikesed peavad olema samuti juhuslikult genereeritud, kusjuures kuulikesed peavad täielikult mahtuma ruudu sisse.

Joonise ülemisse paremasse nurka kirjutada pikima niidi pikkus. Joonise ülemisse vasakusse nurka kirjutada ka autori nimi.

## Väljundi näidis



## Ülesanne 70

## Kalender

(2005/2006)

Kirjutada programm, mis joonistab 30-päevase kuu kalendri. Kuu algab võrdse tõenäosusega kas laupäevaga (nagu alltoodud joonisel) või pühapäevaga. Üldjuhul värvida päevade numbrid mustaks, välja arvatud pühapäevade ja pühade numbrid, mis värvida roheliseks. Iga päev võib olla püha tõenäosusega 0,05 (genereerida juhuslik reaalarv hulgast  $[0, 1)$  ja võrrelda seda arvuga 0,05).

Joonise ülemisse paremasse nurka kirjutada, mitu päeva selles kuus on puhkepäevad (laupäevad, pühapäevad või pühad). Joonise ülemisse vasakusse nurka kirjutada autori nimi.

## Väljundi näidis

The screenshot shows a window titled "Kalender" with a blue header bar. Below the header, it says "Autor: Ülli Õpilane" on the left and "Kuu on 11 puhkepäeva" on the right. The main content area displays a calendar grid. The days of the week are listed on the left: E, T, K, N, R, L, P. The dates are listed in columns: 3, 4, 5, 6, 7, 1, 2; 10, 11, 12, 13, 14, 8, 9; 17, 18, 19, 20, 21, 15, 16; 24, 25, 26, 27, 28, 22, 23; 29, 30. The numbers 28, 29, 30, 23, and 30 are highlighted in green, indicating holidays.

E		3	10	17	24
T		4	11	18	25
K		5	12	19	26
N		6	13	20	27
R		7	14	21	28
L	1	8	15	22	29
P	2	9	16	23	30

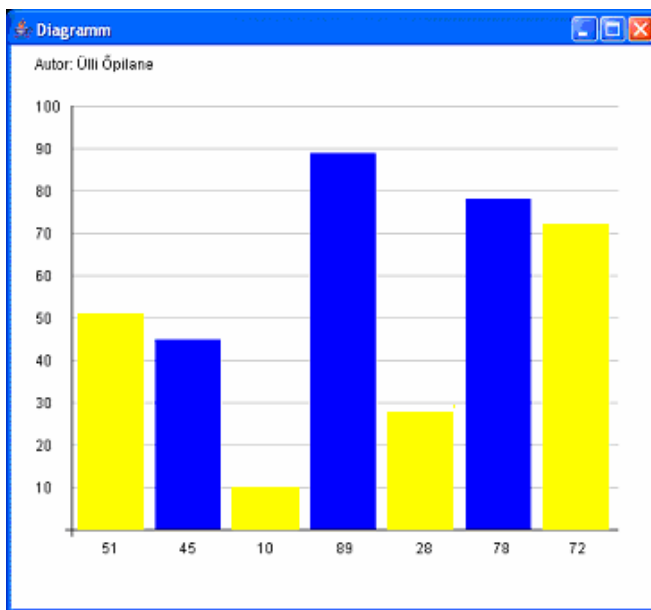
## Ülesanne 71 Diagramm (2005/2006)

Kirjutada programm, mis joonistab tulpdiagrammi. Tulpade arv, mille otsustab programm juhuslikult, on 5, 6 või 7 (joonisel). Kujutatavad täisarvulised väärtused lõigust  $[0, 100]$  valib programm samuti juhuslikult. Tulbad on vaheldumisi punased ja sinised. Iga tulba alla kirjutada sellele vastav väärtus. Vertikaalteljele märkida väärtused 10, 20, ..., 100 ning märgitud punktidest tõmmata horisontaalsed abijooned.

Joonise ülemisse vasakusse nurka kirjutada programmi autori nimi.

NB! Kui juhuslikkusega opereerimine tundub liiga keeruline, siis kirjutada konkreetsete joonisel olevate väärtustega arvestav programm. Sellisel juhul on aga maksimaalne punktide arv 70% võimalikust.

### Väljundi näidis



## Ülesanne 72 Raamaturiil (2005/2006)

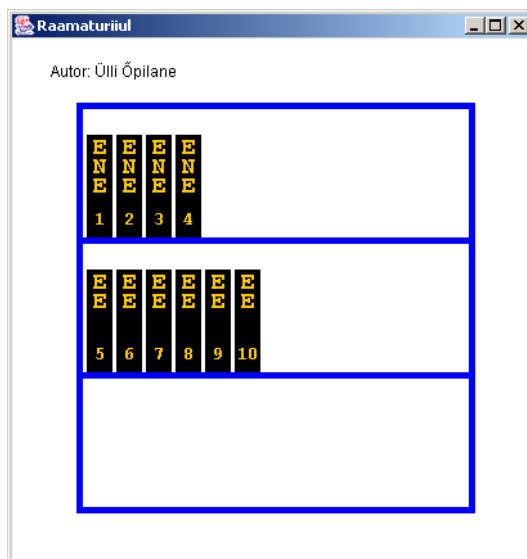
Kirjutada programm, mis joonistab sinise raamaturiili, kus asuvad entsüklopeediakõited. Ülemisele riulile paigutatakse kõited, mille tagaseljal on kirjas ENE (1–4) ja keskmisele riulile kõited, mille tagaseljal on kirjas EE. Keskmisel riulil on kõited alates 5-ndast kuni 9-ndani või 10-ndani (joonisel). Selle, kas viimane kõide on 9. või 10., otsustab programm juhuslikult. Alumine riul jääb tühjaks.

Kõited on musta värvi ning kiri nende seljal on oranž.

Joonise ülemisse vasakusse nurka kirjutada programmi autori nimi.



## Väljundi näidis



## Ülesanne 73

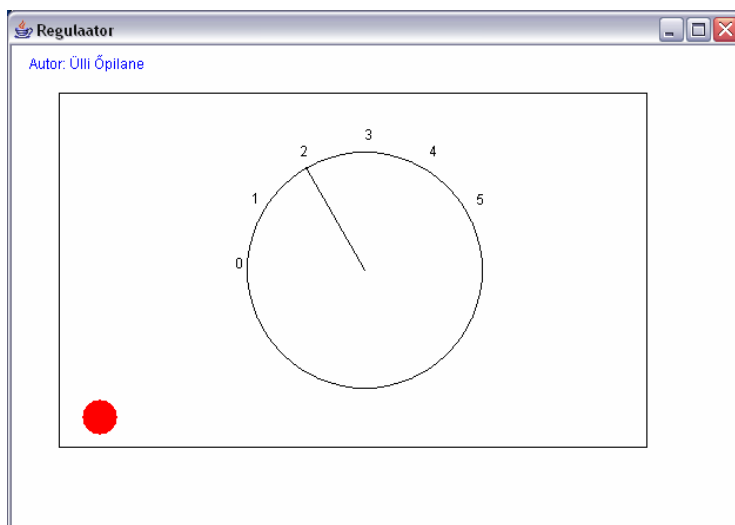
## Regulaator

(2005/2006)

Kirjutada programm, mis joonistab põrandakütteregulaatori esipaneeli. Regulaatori paneel on ristkülikukujuline, selle keskel asub ringikujuline nupp. Skaalal on märgitud numbrid nullist viieni, kusjuures 0 asub „läänes“.

Regulaatori näidiku asend (0, 1, 2, 3, 4 või 5) valitakse juhuslikult. Alumises vasakus nurgas asub ringikujuline indikaator, mis on roheline, kui ei köeta (0), ja punane, kui köetakse (1–5). Joonise vasakusse ülanurka kirjutada autori nimi.

## Väljundi näidis



## Ülesanne 74

## Noolemäng

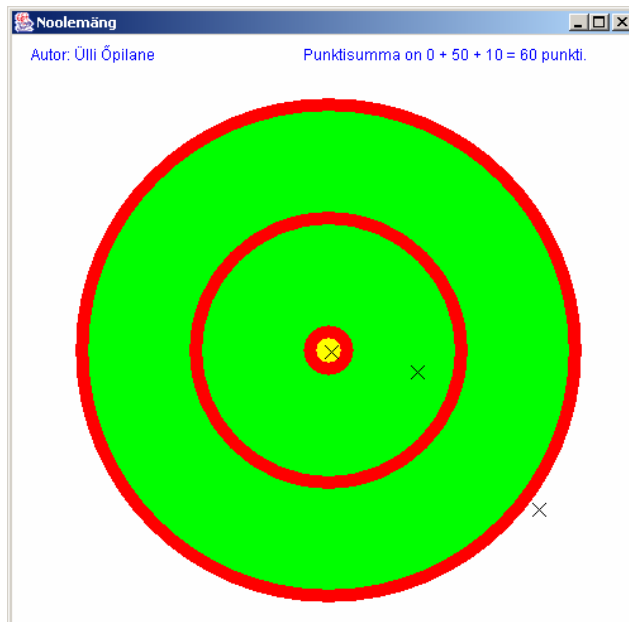
(2005/2006)

Kirjutada programm, mis joonistab noolemängu ringikujulise märklauda. Sooritatakse kolm viset, mida joonisel märgitakse ristikestega. Iga viske jaoks genereeritakse juhuslikult koordinaatide paar. Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise ülemisse paremasse nurka kirjutada visete eest saadud punktid ja punktisumma.

Programm loeb punkte järgmise tabeli järgi.

Piirkond, mida vise tabab	Punktide arv
Südamikring (kollane)	50
Südamikringi ümber olev kitsas rõngas (punane)	25
Sisemine lai rõngas (roheline)	10
Keskmine kitsas rõngas (punane)	30
Välimine lai rõngas (roheline)	10
Välimine kitsas rõngas (punane)	20
Väljaspool märklauda (valge)	0

## Väljundi näidis



## Ülesanne 75

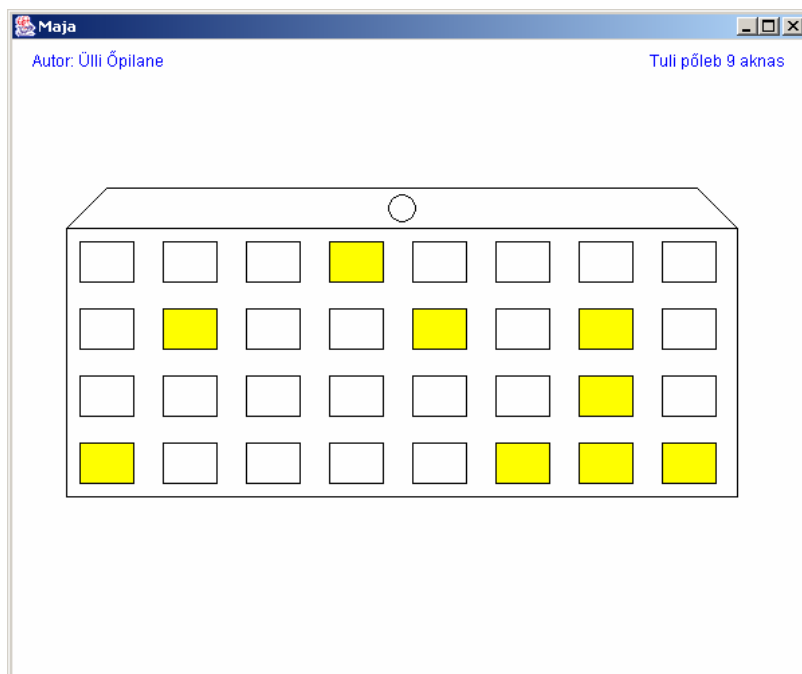
## Maja

(2005/2006)

Kirjutada programm, mis joonistab neljakorruselise maja. Igal korrusel on 8 akent. Iga akna jaoks genereeritakse juhuslik arv nullist üheni. Kui genereeritud arv on suurem kui 0,75, siis selles aknas põleb tuli. Majal on katus ja selles on üks ringkujuline aken, kus kunagi tuli ei põle.

Joonise ülemisse paremasse nurka kirjutada, mitmes aknas põleb tuli. Joonise ülemisse vasakusse nurka kirjutada autori nimi.

### Väljundi näidis



### Ülesanne 76

### Jõusaal

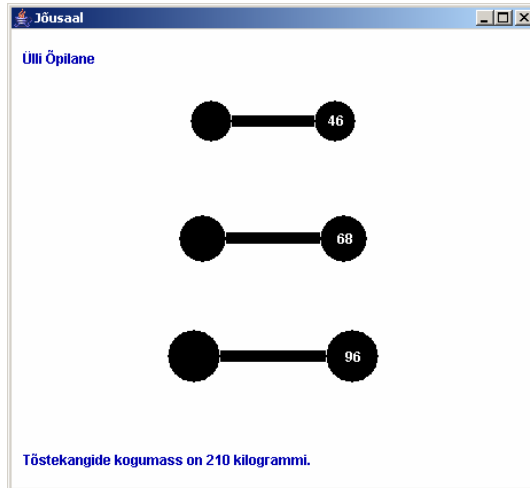
(2006/2007)

Raskejõustiklane teeb treeningut jõusaalis, milles on kolm arhailist tüüpi tõstekangi. Tõstekang koosneb kahest pommiosast ja neid ühendavast latist. Saalis on juhuslikud tõstekangid: kerge massiga vahemikus 40–59 kilogrammi, keskmine massiga vahemikus 60–79 kilogrammi ja raske massiga vahemikus 80–99 kilogrammi. Kangide massid on täisarvud.

Kirjutada programm, mis joonistab kolm tõstekangi vastavalt eelkirjeldatud tingimustele. Pommiosad kujutatakse ringidena, latid ristkülikutena. Tõstekangid on mustad ja ühele pommiosale on kirjutatud tõstekangi mass. Lati pikkus on kaks korda suurem pommiosa diameetrist. Pommiosa diameetriks võtta ligikaudu kuupjuur tõstekangi massist grammides.

Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise allserva kirjutada kangide kogumass.

## Väljundi näidis



## Ülesanne 77

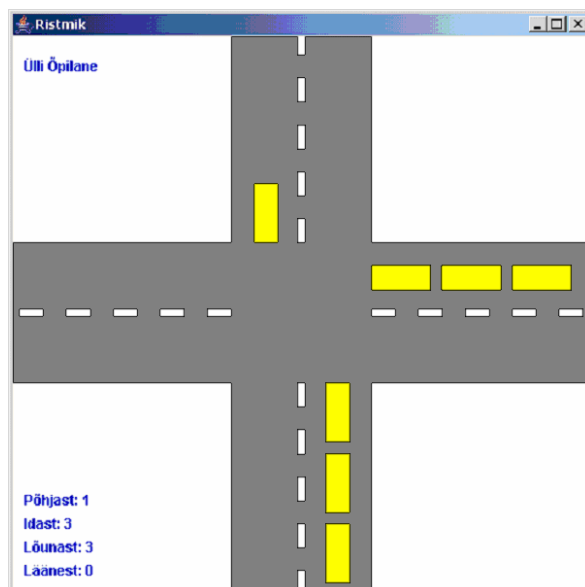
## Ristmik

(2006/2007)

Kirjutada programm, mis joonistab ruudukujulisse aknasse ristmiku. Teed on musta värvi ja neil on valged katkendlikud telgjooned. Igast suunast (põhjast, idast, lõunast ja läänest) on ristmikule jõudnud 0 kuni 3 autot. Autode arv määratakse juhuslikult. Autod on kujutatud heledate (nt kollaste) ristkülikutena.

Joonise ülemisse vasakusse nurka kirjutada autori nimi, alla vasakule kirjutada, kui palju igast suunast autosid ristmikuni on jõudnud.

## Väljundi näidis



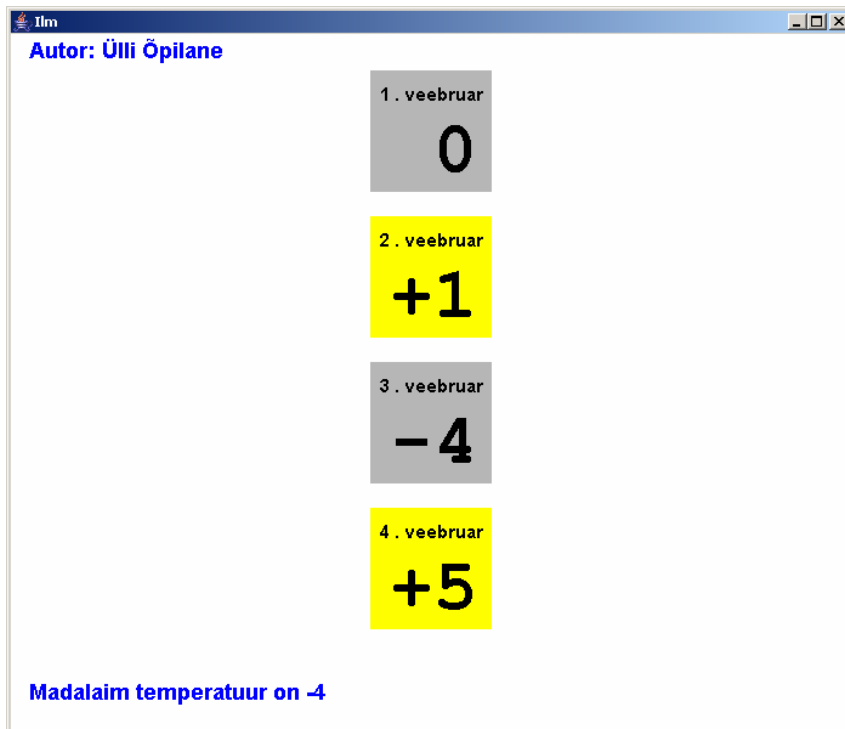
## Ülesanne 78 Ilmaennustus (2006/2007)

Kirjutada programm, mis loob raami ja joonistab sinna 4 ruutu, mis näitavad ennustatavaid temperatuure. Ruudud peavad asuma raami keskel üksteise all.

Ruutudes on kuupäevad 1.—4. veebruarini ning ennustatav temperatuur — juhuslik täisarv vahemikus  $-5 \dots +5$ . Positiivsed temperatuurid tuleb esitada plussmärgiga. Positiivsete temperatuuridega ruudud peavad olema punased, teised sinised.

Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise allserva kirjutada kõige madalam ennustatav temperatuur. Positiivne temperatuur tuleb siingi esitada plussmärgiga.

### Väljundi näidis

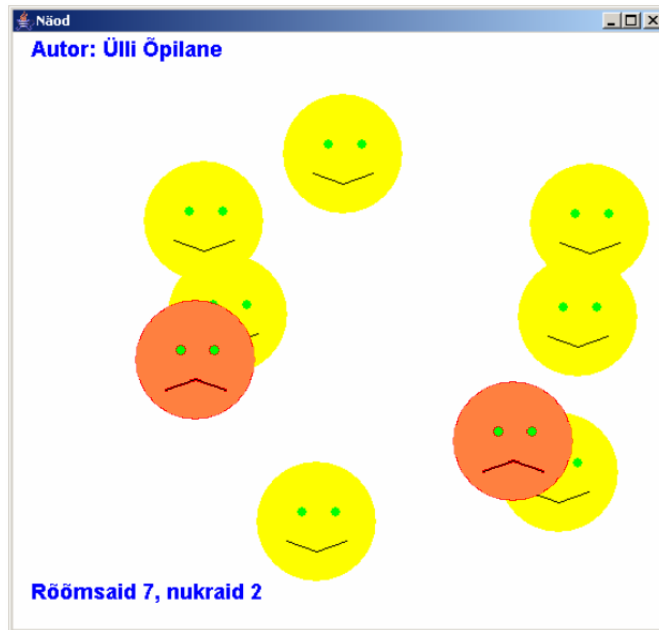


## Ülesanne 79 Näod (2006/2007)

Kirjutada programm, mis joonistab 9 nägu. Nägude asukohad peavad olema juhuslikud. Näod võivad üksteist katta, aga nad peavad täielikult asuma jooniseakna sees. Nägu on rõõmus tõenäosusega 0,8 ja nukker tõenäosusega 0,2. Rõõmus nägu on kollane ja suunurgad üles, nukker nägu on punane ja suunurgad alla.

Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise allserva kirjutada rõõmsate ja nukrate nägude arv.

## Väljundi näidis



## Ülesanne 80

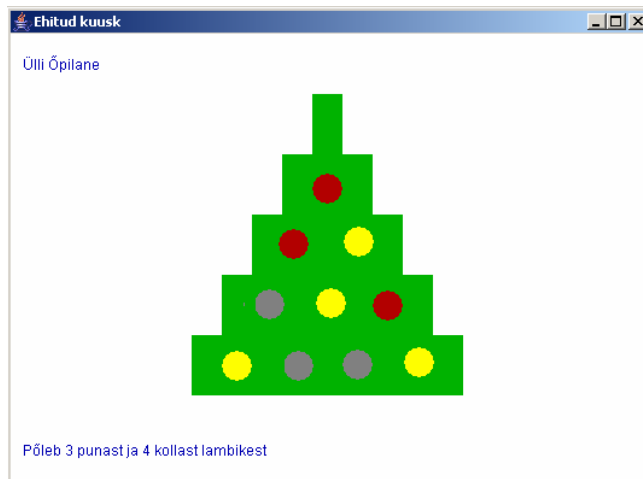
## Ehitud kuusk

(2006/2007)

Kirjutada programm, mis joonistab stiliseeritud roheline kuusepuu, millel paikneb kümme ringikujulist värvilist lambikest. Viis lambikest on põlevana punased, viis kollased, kustunult on lambikesed hallid. Iga lambike põleb tõenäosusega 0,7.

Joonise ülemisse vasakusse nurka kirjutada autori nimi. Joonise allserva kirjutada põlevate punaste lambikeste arv ja põlevate kollaste lambikeste arv.

## Väljundi näidis



# Kirjalikud ülesanded

Käesoleva peatüki ülesannetele nõuti vastuseid kirjalikult paberil, programmi koostada polnud vaja.

## Ülesanne 81                      Puude väetamine                      (2006/2007)

Aias on  $n$  puud, mida aednik peab väetama. Väetis asub kuuris, mille juures ka töö algab ja lõpeb. Kärule mahub väetist kolme puu tarvis. Väetamist alustab aednik kuurile lähimast puust. Pärast puu väetamist siirdub ta lähima veel väetamata puu juurde või (kui kärü sai tühjaks) kuuri juurde tagasi uuesti väetist kärule laadima (ja sealt kuurile lähima väetamata puu juurde). Esimesel tunnil on aedniku kiirus 1,5 m/s, teisel tunnil 1 m/s ja hiljem 0,5 m/s. Ühe puu väetamiseks kulub 3 minutit. Väetise kärule laadimiseks kulub 5 minutit.

Programmeerimishuviline aednik soovib koostada programmi, mis arvutab kõigi puude väetamisel läbitava tee pikkuse ja selleks kulunud aja.

### Küsimused ja ülesanded

- Milliseid sisendandmeid oleks vaja ja milline oleks kõige parem viis neid programmile ette anda? Põhjendada.
- Millised alamtegevused oleks selles programmis mõistlik välja eraldada iseisvateks meetoditeks? Kirjutada välja ka nende meetodite päised (koos formaalsete parameetritega).
- Kas (ja kui, siis kus) selle programmi koostamisel tuleb kasutada kahekordset tsükliit? Selgitada vastust.
- Kas (ja kui, siis kus) oleks selles programmis mõeldav kasutada reaalarvude ümardamist? Selgitada vastust.

## Ülesanne 82                      Vormel                      (2006/2007)

Vormelite võistluse ühel etapil osaleb maksimaalselt 24 võistlejat ja läbitavate ringide arv ei ole suurem kui 90. Aegasid fikseeritakse raja kolmes punktis:  $P_0$  (start/finiš),  $P_1$  ja  $P_2$ . Ajad esitatakse sekundites täpsusega kolm kohta pärast koma. Lisaks ajale fikseeritakse võistleja number. Pärast võitja finišit ei tohi ükski võistleja alustada uut ringi.

Programmeerijalt telliti programm, millega saab kogutud andmete põhjal väljastada paremusjärjestuse, kus iga võistleja kohta on toodud võistleja number, kulunud aeg, läbitud ringide arv. Kui ringide arv on väiksem kui 93% ringide ettenähtud arvust, siis lisada märkus „Ei kvalifitseerunud“.

Samuti peab see programm väljastama, milline võistleja saavutas parima ringiaja, milline parima esimese sektori ( $P_0$  kuni  $P_1$ ), milline parima teise sektori ( $P_1$  kuni  $P_2$ ) ja milline parima kolmanda sektori ( $P_2$  kuni  $P_0$ ) aja.

**Küsimused ja ülesanded**

- a) Millisel kujul oleks kõige parem viis programmile sisendandmeid ette anda? Põhjendada.
- b) Millised alamtegevused oleks selles programmis mõistlik välja eraldada iseisvateks meetoditeks? Kirjutada välja ka nende meetodite päised (koos formaalsete parameetritega).
- c) Milliseid meetodeid saab kasutada korduvalt? Selgitada vastust.
- d) Milliseid programmi osi saaks kasutada ka võistluse ajal ja millistel tingimustel?

**Ülesanne 83                      Tekstide sõnaloend                      (2006/2007)**

Kaasaegse kirjakeele uurimiseks on vaja koostada programm, mis vaatab läbi palju tekstifaile ning koostab kõigi nendes esinevate sõnade loendi (tähestiku järjekorras). Seejuures pole kõik tekstifailid korruga kättesaadavad, vaid neid lisandub järk-järgult, samas peab keeleteadlastele olema igal hetkel kättesaadav seni analüüsitud tekste haarav sõnade loend.

**Küsimused ja ülesanded**

- a) Kuidas peaks programm töötama (millist infot kasutaja programmile ette annab; mida programm teeb/arvutab/salvestab), et nimetatud sõnaloendi järjekordse versiooni loomine kulgeks võimalikult kiiresti?
- b) Eelmise punkti põhimõtteid järgides kirjutada välja sellise meetodi päis, mis koostab sõnaloendi uusimat versiooni. Lisada ka, mis on selle meetodi puhul antud ja mis on tulemus ning kirjeldada meetodi tööd.
- c) Kirjeldada, kuidas saab tekstifaili, milles on üldiselt palju ridu ning igas reas mitu sõna, üksikuteks sõnadeks jagada.
- d) Milliseid faile (kui üldse) peab see programm valitud lahenduse juures avama korduvalt? Selgitada.
- e) Mida oleks sellisel programmil mõistlik kirjutada ekraanile?

**Ülesanne 84                      Rektori valimine                      (2006/2007)**

Rektori valimiste reglement sätestab, et valimistel osutub valituks kandidaat, kelle poolt on hääletanud üle poole valimiskogu liikmetest. Kui enam kui kahe kandidaadi korral ükski kandidaat ei saa vajalikku arvu poolthääli, korraldatakse teine hääletusvoor, milles osalevad kaks enim poolthääli saanud kandidaati. Kui kahe kandidaadi korral ei saa kumbki kandidaat vajalikku arvu poolthääli, korraldatakse uus hääletusvoor, milles osaleb rohkem poolthääli saanud kandidaat. Kui mitu kandidaati (esimeses või teises voorus) saab võrdse arvu hääli ja edasimineja(d) nii ei selgu, siis määratakse edasiminejad loosiga. Kui ka viimases hääletusvoorus ei saa kandidaat vajalikku arvu poolthääli, jääb rektor valimata.



Selleks, et tööd kiirendada, ostetakse spetsiaalne skänner, mis töötleb valimisedelid, leiab kandidaatidele antud häälte arvud ja kirjutab need faili. Häälte arvude töötlemiseks on vaja koostada programm, mis leiab kandidaatide paremusjärjestuse ja valimiste võitja või edasipääsejad.

### **Küsimused ja ülesanded**

- a) Millisel kujul peaksid andmed olema skänneri kirjutatud failis?
- b) Milline peaks olema programmi struktuur, et võimalikult palju meetodeid saaks kasutada kõigi kolme hääletusvooru korral?
- c) Kas kõik vajalikud andmed on igal juhul võimalik saada skännerilt või tuleb midagi ka programmi töö käigus sisestada? Kui tuleb, siis mida?
- d) Mida oleks sellisel programmil mõistlik kirjutada ekraanile?
- e) Milliseid muudatusi tuleks teha või kuidas programm üles ehitada, et teda saaks kasutada mitme skänneri (mitme faili) korral?

## **Ülesanne 85                      Eksamitulemused                      (2006/2007)**

Eksamil on 4 ülesannet, millest kaks esimest annavad kumbki 20 punkti ning kaks viimast kumbki 30 punkti. Vaja on koostada programm, mis paljude eksamitegijate tulemuste põhjal määrab igaihe hinde vastavalt Tartu ülikoolis kehtivale hindamisskaalale ning ühtlasi koostab tulemuste statistilise kokkuvõtte.

### **Küsimused ja ülesanded**

- a) Mis on selle programmi algandmeteks? Kirjeldada muutujad (koos tüüpidega), mis sobivad algandmete hoidmiseks.
- b) Kuidas eristada nendes andmestruktuurides üliõpilast, kes eksamile ei tulnud (hinne „mitteilmunud“), üliõpilasest, kes tuli eksamile, aga sai iga ülesande eest 0 punkti (hinne F)?
- c) Kirjutada programmilõik, mis arvutab punktis 1 kirjeldatud andmestiku abil esimese eksamiülesande tulemuste aritmeetilise keskmise.
- d) Kuidas saab programm teha kindlaks, millises ülesannetest 1—4 olid tulemused kõige paremad?
- e) Mitmekordset tsüklit vähemalt on vaja, et leida iga hinde saajate koguarv ja väljastada see ekraanile? Selgitada.