

## 1. Pildiraam

1 sekund 10 punkti

Maril on kaks pilti, mida ta tahab kordamööda oma toa seinale riputada. Selleks on tal vaja tellida pildiraam. Kuna raamiliist on üsna kallid, tahab Mari tellida ühe sellise raami, mille sisse mahuks ükskõik kumb pilt, ja mille nelja serva peale kokku kuluks võimalikult vähe liistu.

Kirjutada programm, mis leiab minimaalse ümbermõõduga raami, mille sisse mahub ükskõik kumb Mari pilt, kui on teada, et raami saab seinale riputada nii püsti (pikem serv vertikaalselt) kui ka rõhtsalt (pikem serv horisontaalselt).

**Sisend.** Tekstifaili `raam.sis` esimesel real on esimese ja teisel real teise pildi küljepikkused. Kõik pikkused on täisarvud  $1 \dots 100$ .

**Väljund.** Tekstifaili `raam.val` ainsale reale väljastada kaks tühikuga eraldatud täisarvu, pildiraami servade pikkused. Kui minimaalse ümbermõõduga raame on mitu, väljastada ükskõik milline neist.

<b>Näide.</b>	<code>raam.sis</code>	<code>raam.val</code>
	30 30	30 40
	20 40	

Mari piltide mõõdud on  $30 \times 30$  ja  $20 \times 40$ . Esimene (ruudukujuline) pilt mahub  $30 \times 40$  raami sisse ükskõik kumba pidi ja raami pikkusest jääb 10 cm üle. Teine (piklik) pilt mahub raami sisse ainult pikuti ja raami laiuusest jääb 10 cm üle. Raami ümbermõõt on kokku 140 cm ja vähemaga läbi ei saa.

## 2. Pikkuse järjekorras

1 sekund 20 punkti

On antud  $N$  õpilase pikkused.

Kirjutada programm, mis leiab pikkuselt kolmanda õpilase pikkuse.

**Sisend.** Tekstifaili `pikk.sis` esimesel real on täisarv  $N$  ( $3 \leq N \leq 10\,000$ ) ja järgmisel real  $N$  täisarvu  $A_1, A_2, \dots, A_N$  ( $0 < A_i \leq 1000$ ), õpilaste pikkused.

**Väljund.** Tekstifaili `pikk.val` ainsale reale väljastada üks arv, pikkuselt kolmanda õpilase pikkus.

<b>Näide.</b>	<code>pikk.sis</code>	<code>pikk.val</code>
	5	173
	175 173 169 173 171	

Pikima õpilase kasv on 175 cm, teist ja kolmandat kohta jagavad 173 cm pikkused õpilased.

### 3. TopSwops

1 sekund 30 punkti

TopSwops on ühe mängija kaardimäng, mida mängitakse  $N$  kaardiga. Kaardid on nummerdatud  $1 \dots N$ . Iga numbriga kaarti on pakis täpselt üks ja number on kaardi mõlemal küljel.

Mängu alguses kaardid segatakse ja pannakse lauale ühte pakki. Edasi võtab mängija igal käigul paki pealt nii palju kaarte, kui on pealmise kaardi väärtus, keerab võetud pakiosa kummuli ja paneb selle paki peale tagasi.

Selge on see, et kui paki pealmiseks saab kaart number 1, siis edasi enam midagi ei muutu, sest kõik järgmised käigud keeraks seda kaarti kohapeal ringi.

Kirjutada programm, mis leiab, mitme käigu järel selline seis tekib.

**Sisend.** Tekstifaili `tops.sis` esimesel real on kaartide arv  $N$  ( $1 \leq N \leq 300$ ). Faili teisel real on  $N$  tühikutega eraldatud täisarvu  $A_i$  ( $1 \leq A_i \leq N$ ;  $A_i \neq A_j$ , kui  $i \neq j$ ), kaartide järjestus paki algseisus ülalt alla lugedes.

**Väljund.** Tekstifaili `tops.val` ainsale reale väljastada üks täisarv, käikude arv sisendis näidatud seisust alates. Võib eeldada, üheski testis ei ületa vastus 10 000.

<b>Näide.</b>	<code>tops.sis</code>	<code>tops.val</code>
	4	2
	3 2 4 1	

Algseis on 3 2 4 1. Esimesel käigul pööratakse ümber kolmest pealmisest kaardist koosnev osa (3 2 4  $\rightarrow$  4 2 3). Tulemus on 4 2 3 1. Teisel käigul pööratakse ümber neljast kaardist koosnev osa (ehk kogu pakk). Tulemus on 1 3 2 4 ja mäng saabki läbi.

## 4. Copy-Paste

1 sekund

40 punkti

Juku tegi tunnis koerust ja õpetaja käskis tal karistuseks kirjutada sada korda lauset “Ma pean korralikult käituma.”

Looduse säästmiseks käskis õpetaja lauset paberile kirjutamise asemel arvutisse sisestada.

Juku kirjutas ühe lause valmis ja avastas siis, et õpetaja arvutis olev programm laseb lisaks teksti sisestamisele teha järgmisi operatsioone: klahvikombinatsioon Ctrl-A märgib kogu sisestatud teksti; kombinatsioon Ctrl-C kopeerib märgitud teksti lõikepuhvrisse ja Ctrl-V lisab lõikepuhvri sisu dokumendile.

Kui Ctrl-V vajutamise ajal on dokumendis mingi tekst märgitud, siis märgitud tekstiosa kustutatakse ja lõikepuhvri sisu pannakse selle asemele; kui midagi märgitud ei ole, siis lihtsalt lisatakse lõikepuhvri sisu dokumendile.

Juku ei taha vahele jääda sellega, et ta on lauset kirjutanud rohkem kui nõutud arvu kordi. Õpetaja ei usuks mingil juhul, et ta kogemata üle pingutas ja hakkaks kohe valskust kahtlustama.

Kirjutada programm, mis aitab Jukul leida lühima klahvivajutuste jada, millega ta saab oma ühte lauset paljundada täpselt vajalikuks arvuks koopiateks.

**Sisend.** Tekstifaili `acvv.sis` ainsal real on vajalike eksemplaride arv  $N$  ( $1 \leq N \leq 1\,000\,000$ ).

**Väljund.** Tekstifaili `acvv.val` ainsale reale väljastada üks täisarv: minimaalne klahvivajutuste arv, millega on võimalik teha ühest lausest täpselt  $N$ .

<b>Näide.</b>	<code>acvv.sis</code>	<code>acvv.val</code>
	3	5

Ctrl-A valib kogu teksti, Ctrl-C kopeerib selle lõikepuhvrisse, Ctrl-V asendab dokumendi sisu lõikepuhvri omaga (dokumendis on nüüd endiselt üks lause) ja veel kaks Ctrl-V vajutust lisavad veel kaks lauset.

## 5. Multi-Ant

1 sekund 50 punkti

Suure, paljudest üksteisest sõltuvatest moodulitest koosneva tarkvaratoote kompileerimine ja pakendamine võib olla päris mahukas töö. Ant on programm selle automatiseerimiseks.

Ant saab ette kõigi sammude omavahelised sõltuvused ja teeb vajalikud tegevused õiges järjekorras, see tähendab nii, et ühtegi sammu ei alustata enne kui kõik need, mille tulemustest ta sõltub, on juba ära tehtud.

Paraku on suuremate projektide puhul ka sellest vähe, sest kõigi sammude ühes arvutis järjest tegemine võtab lihtsalt liiga kaua aega. Multi-Ant on Anti edasiarendus eesmärgiga jagada vajalikke tegevusi mitme arvuti vahel.

Kirjutada Multi-Anti komponent, mis saab ette info vajalike sammude omavaheliste sõltuvuste kohta ja koostab plaani nende jagamiseks mitme arvuti vahel nii, et lõpptulemus valmib minimaalse võimaliku ajaga. Iga sammu tegemiseks kulub üks ajaühik ja kui kaks sammu teineteisest ei sõltu, võib neid teha paralleelselt eraldi arvutitel.

**Sisend.** Tekstifaili `mant.sis` esimesel real on vajalike sammude arv  $N$  ( $1 \leq N \leq 1000$ ) ja järgmisel  $N$  real igaühel info ühe sammu eelduste kohta. See info on antud kujul *samm eeldus1 eeldus2 ...*, kus *samm* on sammu enda nimi ja *eeldus1*, *eeldus2*, ... nende sammude nimed, mis peavad olema tehtud enne kui *samm* alata võib. Iga nimi koosneb 1...20 väikesest ladina tähest, kahe kõrvuti oleva nime vahel on alati täpselt üks tühik ja ühegi rea kogupikkus ei ületa 250 märki.

**Väljund.** Tekstifaili `mant.val` esimesele reale väljastada kõigi sammude tegemiseks vajalik aeg  $T$  ja järgmisele  $T$  reale tegevusplaan ajaühikute kaupa. Faili teisele reale väljastada tühikutega eraldatult esimesel ajaühikul paralleelselt tehtavate sammude nimed, kolmandale reale teisel ajaühikul tehtavad sammud j.n.e. Võib eeldada, et avuteid on piiramatul koguses — vajadusel ostab firma neid juurde. Kui minimaalse ajaga plaane on mitu väljastada ükskõik milline neist.

<b>Näide.</b>	<code>mant.sis</code>	<code>mant.val</code>
	8	6
	checkout clean	init
	clean init	clean
	compile checkout clean	checkout
	coverage compile	compile
	deploy test	test coverage
	init	deploy report
	report coverage	
	test compile	

## 6. Minotauros

50 punkti

Mäletatavasti püüdsid arhitekt Daidalos ja tema poeg Ikaros Kreeta saarelt lennates põgeneda, kuid Ikaros tõusis liiga kõrgele ja päikese kuumus sulatas vaha, mis tema tiibu koos hoidis. Kuigi legend väidab, et see juhtus mere kohal ja Ikaros uppus, näitavad uusimad arheoloogilised avastused, et tegelikult kukkus ta labürinti, mille Daidalos oli ehitanud, et sulgeda sinna inimese keha ja härja peaga koletis Minotauros.

Ülalt labürindi suunas langev Ikaros näeb hästi nii labürindi plaani kui ka Minotaurose asukohta selles. Kuna tema tiivad enam eriti ei kannu, on tal väga vähe aega, et otsustada, kuhu poole ennast suunata.

Aita Ikarosel leida maandumiseks selline koht, et Minotaurose lühim tee oma asukohast Ikarose maandumiskohani oleks võimalikult pikk.

Labürindi plaani võib vaadelda kui  $N \times M$  ruudustikku, mille iga ruut on kas käik või sein ja kus ühelt käiguruudult saab teisele liikuda ainult siis, kui neil on ühine serv.

**Sisend.** Sisendfaili esimesel real on testi järjekorranumber  $T$  ning labürindi pikkus  $N$  ja laius  $M$ . Järgmisel  $N$  real on igaühel täpselt  $M$  märki. Need read kujutavad labürindi plaani, kus '.' tähistab käiku, 'X' seina ja 'M' Minotaurose asukohta. On teada, et labürindil on maksimaalselt üks sisse-väljapääs.

**Väljund.** Väljundfaili esimesele reale väljastada testi järjekorranumber  $T$  ja järgmisele  $N$  reale igaühele täpselt  $M$  märki. Need  $N$  rida peavad kujutama sisendis antud labürindi plaani, kus täpselt üks märk '.' on asendatud märgiga 'I' nii, et kui Ikaros maandub labürindis sellele kohale, siis on Minotaurose lühim tee oma asukohast Ikarose maandumiskohani maksimaalselt pikk. Kui maksimaalse kaugusega vastuseid on mitu, väljastada ükskõik milline neist.

Näide.	Sisendfail	Väljundfail
	0 5 5	0
	XXXXX	XXXXX
	X...X	XI..X
	XXX.X	XXX.X
	XM..X	XM..X
	XXXXX	XXXXX

**Hindamine.** Selles ülesandes on antud 10 sisendandmete komplekti failides `minotest.01.sis` kuni `minotest.10.sis` ja lahendusena on vaja esitada neile vastavad väljundfailid. Programmi esitamine pole vajalik ja seda ei hinnata.

**Märkus.** Lisaks on selles ülesandes antud abiprogramm labürintide vaatamiseks ja kauguste hindamiseks. Programm on failis `MinoTool.jar` ja selle kasutamiseks on vajalik Java Runtime, mille saab tasuta aadressilt <http://www.java.com/getjava/>.