

## 1. Bensiinihind

1 sekund 20 punkti

Majandusanalüütik Toomas kirjutab iga päev üles bensiiniliitri hinna tema kodu ees olevas tanklas. Nüüd tahab ta nende andmete põhjal leida pikima tõusu- ja langusperioodi. Kirjutada programm, mis aitab tal seda teha.

**Sisend.** Tekstifaili `bh.sis` esimesel real on täisarv  $N$  ( $1 \leq N \leq 10\,000$ ): Toomase hinnajada pikkus. Faili teisel real on  $N$  tühikutega eraldatud täisarvu  $H_i$  ( $0 < H_i \leq 10\,000$ ,  $1 \leq i \leq N$ ): bensiini liitrihinnad päevade kaupa.

**Väljund.** Tekstifaili `bh.val` esimesele reale väljastada kaks tühikuga eraldatud täisarvu  $A$  ja  $L$ , mis näitavad, et pikim periood, mille vältel bensiini liitrihind kordagi ei langenud, algas päeval  $A$  ja lõppes päeval  $L$ . Kui maksimaalse pikkusega perioode on mitu, väljastada neist kõige hilisem. Faili teisele reale väljastada samal kujul pikim periood, mille vältel hind kordagi ei tõusnud.

<b>Näide.</b>	<code>bh.sis</code>	<code>bh.val</code>
	7	1 3
	1 2 3 2 3 2 1	5 7

## 2. Hiirekursor

1 sekund 20 punkti

Programmeerija Priit peab oma arvuti ekraanil oleva akna ümberpaigutamiseks liigutama hiirekursori selle servale. Aken on riskülik, mille servad on paralleelsed ekraani äärtega ja seda võib lohistamiseks haarata ükskõik millisest servast või nurgast. Tüüpilise programmeerijana püüab Priit minimeerida füüsilist pingutust. Kirjutada programm, mis annab Priidule teada, millisest punktist tuleks tal akent haarata.

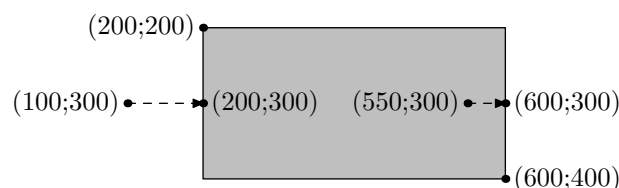
**Sisend.** Tekstifaili `hk.sis` esimesel real on kaks tühikuga eraldatud täisarvu  $X$  ja  $Y$  ( $0 \leq X \leq 10\,000$ ,  $0 \leq Y \leq 10\,000$ ): hiirekursori lähtekoha koordinaadid. Faili teisel ja kolmandal real on samuti kummalgi kaks tühikuga eraldatud täisarvu: vastavalt akna vasaku ülemise nurga koordinaadid  $X_1$  ja  $Y_1$  ning parema alumise nurga koordinaadid  $X_2$  ja  $Y_2$  ( $0 \leq X_1 \leq X_2 \leq 10\,000$ ,  $0 \leq Y_1 \leq Y_2 \leq 10\,000$ ).

**Väljund.** Tekstifaili `hk.val` ainsale reale väljastada kaks tühikuga eraldatud täisarvu: kursori algasukohale lähima servapunkti koordinaadid. Kui minimaalse kaugusega punkte on mitu, väljastada ükskõik milline neist.

<b>Näide.</b>	<code>hk.sis</code>	<code>hk.val</code>
	100 300	200 300
	200 200	
	600 400	

<b>Näide.</b>	<code>hk.sis</code>	<code>hk.val</code>
	550 300	600 300
	200 200	
	600 400	

Näiteid illustreerib järgmine joonis:



### 3. Ajaintervallid

1 sekund 20 punkti

Veebiserver logib kõik päringud kuupäevaliselt ja kellaaajaliselt sekundi täpsusega. Statistika tegemiseks on vaja logikirjeid grupeerida erinevate ajaintervallide kaupa, näiteks kõik ühe päeva või ühe tunni kirjed.

Kõige lihtsam on erinevaid ühepikkusi ajaintervalle esitada vastava intervalli algushetke kaudu, ümardades kuupäeva ja kellaaaja allapoole vastava ühiku kordseks. Näiteks kellaja 18:22:55 ümardamisel allapoole 5 minuti kordseks saame 18:20:00.

Kirjutada programm, mis teisendab antud ajahetke teda sisaldava antud pikkusega intervalli algushetkeks.

**Sisend.** Tekstifaili `ai.sis` esimesel real on antud ajahetk kujul `YYYY-MM-DD hh:mm:ss`, kus `YYYY` tähistab aastat, `MM` kuud, `DD` kuupäeva, `hh` tundi, `mm` minutit ja `ss` sekundit. Aastaarv on alati antud neljakohalisena, kõik teised kahekohalistena. Võib eeldada, et tegu on korrektse kuupäeva ja kellaaajaga. Faili teisel real on intervalli pikkuse tähis: `MIN` (1 minut), `5MIN` (5 minutit), `HOURL` (tund), `DAY` (ööpäev), `MON` (kuu) või `YEAR` (aasta).

**Väljund.** Tekstifaili `ai.val` ainsale reale väljastada otsitava ajaintervalli algushetk sisendiga samal kujul.

<b>Näide.</b>	<code>ai.sis</code> 2008-11-10 09:08:07 MIN	<code>ai.val</code> 2008-11-10 09:08:00
---------------	---	--

<b>Näide.</b>	<code>ai.sis</code> 2008-11-10 09:08:00 5MIN	<code>ai.val</code> 2008-11-10 09:05:00
---------------	--	--

<b>Näide.</b>	<code>ai.sis</code> 2008-11-10 00:00:00 MON	<code>ai.val</code> 2008-11-01 00:00:00
---------------	---	--

## 4. Kaubaveod

1 sekund

40 punkti

Elboonia valitsus otsustas investeerida kaubavedudesse raudteel ja muretseda uued vedurid. Paraku on ka raudteed ise üsna kehvast seisukorras ja igale teelõigule on kehtestatud teljekoormuse ülempiir.

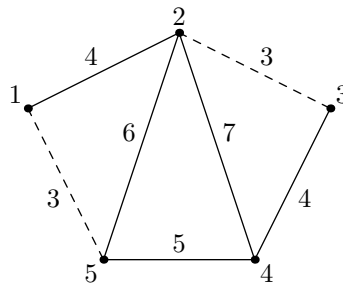
Kirjutada programm, mis leiab maksimaalse teljekoormuse, millega veduril on veel võimalik sõita igast linnast igasse teise (kuigi võib-olla mitte kõige lühemat teed mööda).

**Sisend.** Tekstifaili `kv.sis` esimesel real on kaks tühikuga eraldatud täisarvu: linnade arv  $N$  ( $2 \leq N \leq 100$ ) ja raudteede arv  $M$  ( $1 \leq M \leq 1000$ ). Linnad on nummerdatud  $1 \dots N$ . Järgmisel  $M$  real on igaühel kolm tühikutega eraldatud täisarvu: ühe teelõigu otstes olevate linnade numbrid  $A_i$  ja  $B_i$  ( $1 \leq A_i \leq N$ ,  $1 \leq B_i \leq N$ ,  $A_i \neq B_i$ ) ning sellel teel kehtiv teljekoormuse piirang  $C_i$  ( $0 < C_i \leq 10\,000$ ). Ühe linnapaari vahel võib olla ka mitu teed.

**Väljund.** Tekstifaili `kv.val` ainsale reale väljastada üks täisarv: maksimaalne teljekoormus, millega on veel võimalik vedada kaupa ükskõik kust ükskõik kuhu kogu riigi ulatuses.

Näide.	kv.sis	kv.val
	5 7	4
	1 2 4	
	1 5 3	
	2 3 3	
	2 4 7	
	2 5 6	
	3 4 4	
	4 5 5	

Alloleval joonisel pidevjoonega märgitud teelõigud taluvad kõik teljekoormust vähemalt 4 ühikut ja neid mööda on võimalik pääseda igast linnast igasse teise. Mistahes suurema teljekoormuse korral poleks see enam võimalik.



## 5. Rubiku kuubik

80 punkti

Rubiku väike kuubik koosneb  $2 \times 2 \times 2$  omavahel ühendatud kuubist. Nummerdame nende kuupide väljapoole paistvad tahud, nagu näidatud alloleval kuubiku pinnalaotusel:

	1	2					
	3	4					
5	6	9	10	13	14	17	18
7	8	11	12	15	16	19	20
	21	22					
	23	24					

Algseisus on kuubiku iga tahk ühte, teistest tahkudest erinevat värvi. Kui tähistame neid värve tähtedega A...F, võime kuubiku iga oleku märkida üles 24-tähelise sõnena, milles positsioonil  $i$  olev täht näitab kuubi pinnal kohal  $i$  oleva ruudu värvi. Algseisus on ruudud 1...4 värvi A, ruudud 5...8 värvi B jne. Selle seisuga tähis on AAAABBBBCCCCDDDEEEFFFFF.

Edasi on võimalik igal sammul kuubiku kihte kolmes erinevas tasapinnas üksteise suhtes  $90^\circ$ ,  $180^\circ$  või  $270^\circ$  võrra keerata. Kokku on seega igas seisus üheksa erinevat võimalikku sammu. Keerates ruute 1...4 sisaldavat tahku  $90^\circ$  võrra päripäeva, saame seisuga AAAACCBDDCCEEDDBBEEFFFFF. Keerates seejärel ruutude 9...12 tahku  $90^\circ$  võrra vastupäeva, saame AAEDCABADCDCEFFDBBEECBFFF.

Leida viis antud seisus kuubiku viimiseks tagasi algseisu võimalikult väheste sammudega. Enne antud seisuga lahendamist võib kuubiku tervikuna keerata suvalisse asendisse (st ükskõik millise tahu pealmiseks ja ükskõik millise tahu enda poole). Seda keeramist sammuks ei loeta.

**Sisend.** Tekstifaili `rk.sis` ainsal real on tähtedest A...F koosnev sõne pikkusega 24.

**Väljund.** Kui sisendis antud sõne on korrektne Rubiku kuubiku seis, väljastada tekstifaili `rk.val` esimesele reale kuubiku algseisu viimiseks vajalike sammude arv  $N$ , teisele reale kuubiku seis enne esimest sammu (see võib sisendis antud seisust erineda kuubiku tervikuna keeramise võrra) ja järgmisele  $N$  reale kuubiku seisud pärast iga sammu.

Kui sisendis antud sõne ei ole korrektne Rubiku kuubiku seis, väljastada faili esimesele reale sõna VIGA ja alates teisest reast põhjendus. Põhjendus väljastada vabas vormis tekstina (seda hindab inimene) eesti, vene või inglise keeles.

<b>Näide.</b>	<code>rk.sis</code>	<code>rk.val</code>
	ADAEBBEECABADCDCEFFDFCFB	2 AAEDCABADCDCEFFDBBEECBFFF AAAACCBDDCCEEDDBBEEFFFFF AAAABBBBCCCCDDDEEEFFFFF

**Hindamine.** Selles ülesandes on antud 12 sisendandmete komplekti failides `rktest.01.sis` kuni `rktest.12.sis` ja lahendusena on vaja esitada neile vastavad väljundandmete komplektid failides `rktest.01.val` kuni `rktest.12.val`. Programmi esitamine pole vajalik ja seda ei hinnata.

Ülesande osad annavad punkte järgmiselt:

- Sisendite klassifitseerimine korrektseteks ja ebakorrektseteks: 10% ülesande väärtusest.
- Ebakorrektsete sisendite ebakorrektsuse põhjendused: 30% ülesande väärtusest.
- Korrektsete sisendite lahendamine: 30% ülesande väärtusest.
- Korrektsete sisendite lahendamine minimaalse sammude arvuga: 30% ülesande väärtusest.