

# Datatähti 2004 -loppukilpailu

## algoritmitehtävä

Tietojenkäsittelytieteen laitos  
HELSINGIN YLIOPISTO

5. helmikuuta 2004

## 1 Koodinkorjaaja

Voisiko tietokoneen ohjelmoida etsimään ja korjaamaan ohjelmakoodissa piileviä virheitä automaattisesti? Ainahan voi yrittää...

## 2 Tehtävä

Syötteenä saadaan ohjelmalistaus, joka on kirjoitettu seuraavalla hyvin suppealla ohjelmointikielillä. Listaus koostuu peräkkäin numeroiduista riveistä  $1, 2, 3, \dots, N$ . Kukin rivi  $1 \leq r \leq N$  koostuu seuraavanlaisesta ehtolauseesta:

Jos syötebitin  $i_r$  arvo on  $b_r$ , niin jatka suoritusta riviltä  $s_r$ .

Jos syötebitin  $i_r$  arvo onkin vastakkainen (eli  $1 - b_r$ ), niin suoritus jatkuukin tätä riviä numero  $r$  seuraavalta riviltä numero  $r + 1$ .

Suoritus alkaa listauksen ensimmäiseltä riviltä 1. Suoritus päättyy, jos sen kuluessa siirrytään sellaiselle riville  $N + 1$ , jota ei olekaan olemassa.

Ohjelmalistaus on virheetön, jos on olemassa sellaiset syötebittien  $i_r$  arvot, joilla suoritus päättyy.

Kirjoita sellainen ohjelma, joka saa syötteenään tällaisen ohjelmalistauksen, ja tulostaa

- sellaiset rivit  $r_j$ , että listauksesta tulee virheetön, kun muutetaan niillä olevat arvot  $b_{r_j}$  vastakkaisiksi
- sellaiset syötebittien  $i_r$  arvot, joilla tämän muutetun ohjelmalistauksen suoritus päättyy.  
(Halutaan siis paitsi muutokset myös todiste sille, että ohjelmalistaus todellakin on virheetön muutosten jälkeen.)

Lisäksi halutaan, että muutettavien rivien  $r_j$  lukumäärä olisi mahdollisimman pieni.

## 3 Syöttö

Ohjelmasi lukee syötteenä saatavan ohjelmalistauksen tekstitiedostosta nimeltään `rectify.in`.

Tiedoston ensimmäinen rivi koostuu ohjelmalistauksen rivien lukumäärästä kokonaislukuina  $1 \leq N \leq 2500$ .

Loppu tiedostosta koostuu itse ohjelmalistauksesta siten, että listauksen rivi  $1 \leq r \leq N$  on tiedoston rivinä  $r + 1$ . Tiedoston rivi  $r + 1$  koostuu kolmesta kokonaisluvusta

$$1 \leq i_r \leq N \quad 0 \leq b_r \leq 1 \quad 1 \leq s_r \leq N$$

jotka on eroteltu toisistaan välilyönnillä. Näiden kolmen kokonaisluvun merkitykset on selostettu kappaleessa 2.

Tässä on esimerkki syötteestä:

```
3
1 1 1
3 0 1
3 1 1
```

## 4 Tulostus

Ohjelmasi kirjoittaa tuloksenaan tekstitiedoston nimeltään `rectify.out`. Sen sisällön pitää olla seuraavanlainen:

1. Ensimmäisenä rivinä on ohjelmalistauksesta muutettavien rivien lukumäärä kokonaislukuna  $0 \leq M \leq N$ .
2. Seuraavaksi ovat muutettavien rivien numerot  $r_j$  (missä siis  $1 \leq j \leq M$  ja  $1 \leq r_j \leq N$ ). Jokainen rivinnumero on kokonaislukuna  $r_j$  omana rivinään.  
Rivinumeroiden keskinäinen järjestys on vapaa. Sama rivinnumero saa esiintyä korkeintaan kerran.
3. Rivinumeroiden jälkeisenä rivinä on kokonaislukuna  $0 \leq L \leq N$  niiden syötebittien  $i_r$  lukumäärä joille haluat antaa arvon 1.
4. Lopuksi ovat niiden syötebittien  $i_k$  numerot joille haluat antaa arvon 1 (missä siis  $1 \leq k \leq L$  ja  $1 \leq i_k \leq N$ ). Jokainen syötebittinnumero  $i_k$  on kokonaislukuna omana rivinään.  
Syötebittinumeroiden keskinäinen järjestys on vapaa. Sama syötebittinnumero saa esiintyä korkeintaan kerran.  
Ne syötebitit  $1 \leq i'_k \leq N$ , jotka eivät esiinny tulostiedostossa, saavat arvon 0.

Tulostiedosto kelpaa vastaukseksi, jos se täyttää seuraavan ehdon: Kun kappaleen 3 syötteenä saatuun ohjelmalistaukseen tehdään kohdan 2 muutokset, niin tuloksena saadun ohjelmalistauksen suoritus pysähtyy kohdan 4 mukaisella syötteellä.

Tässä on esimerkki kappaleen 3 esimerkkisyötetiedostoa vastaavasta vastaukseksi kelpaavasta tulosteesta:

```
1
2
0
```

Se ehdottaa yhtä ( $M = 1$ ) muutosta: Jos ohjelmalistauksen rivi 2 (eli syötteen kolmas rivi) muutetaan muotoon

```
3 1 1
```

niin muutetun listauksen suoritus päättyy sellaisella syötteellä, jossa kaikkien syötebittien 1, 2 ja 3 arvona on 0 (koska  $K = 0$ ).

Se on samalla eräs ratkaisu, jossa  $M$  on mahdollisimman pieni: Alkuperäinen ohjelmalistaus ei pysähtynyt millään syötebittien arvoilla, joten ainakin yksi muutos tarvitaan.

## 5 Pisteytys

Jos ohjelmasi käänнос suorituskelpoiseksi tiedostoksi epäonnistuu, niin saat koko tehtävästä 0 pistettä. Loppukilpailun tiukan aikataulun vuoksi ei ole mahdollista korjailla edes triviaaleja virheitä jälkikäteen.

Muuten ohjelmaasi testataan 20 syötteellä. Ohjelmallesi annetaan kuhunkin testiajoon työmuistia 64 megatavua. Tämä työmuistiraja sisältää kaiken muun ohjelmasi käyttämän muistin paitsi sen suorituskelpoista ohjelmakoodia.

Kukin näistä testiajoista pisteytetään seuraavasti:

- Jos kappaleen 4 tulostiedostoa `rectify.out` ei syntynyt tai se ei kelpaa vastaukseksi, niin ohjelmasi saa tästä testistä 0 pistettä.
- Muuten ohjelmasi peruspistemäärä tästä testistä on 5 pistettä.

Tästä peruspistemäärästä vähennetään pisteitä seuraavasti.

- Jos ohjelmasi käytti yli 2 sekuntia keskusyksikköaikaa, niin siltä vähennetään 1 piste jokaiselta täyttyneeltä sekunnilta.

Siis 3,5 sekunnin käyttämisestä vähennetään 3 pistettä, ja niin edelleen.

Tämä keskusyksikköaika sisältää kaiken sen ajan, jonka tietokoneen keskusyksikkö käyttää suorittaessaan sinun omaa ohjelmakoodiasi. Se ei sisällä sitä aikaa, jonka se käyttää suorittaessaan käyttöjärjestelmäpalveluita. Erityisesti se ei sisällä sitä aikaa, joka kuluu syötteen lukemiseen tiedostosta `rectify.in` työmuistiin tai valmiin vastauksen tulostamiseen tiedostoon `rectify.out`. Toisaalta se sisältää erilaisissa kirjastokutsuissa käytetyn ajan, esimerkiksi luetun syötteen muuntamisen tekstimuodosta vastaavaksi numeroksi.

- Jos ohjelmasi tulostama muutosten lukumäärä  $M$  on suurempi kuin pienin tarvittava muutosten lukumäärä, niin siltä vähennetään 1 piste jokaisesta turhasta muutoksesta.
- Jos näin lasketut pisteet olisivat negatiiviset, niin ohjelmasi saa tästä testistä 0 pistettä.

Kaikkien testien suurin yhteispistemäärä on siten 100 pistettä ja pienin 0 pistettä.

Ohjelmat testataan samanlaisella tietokoneella kuin kilpailukoneesikin on (eli AMD Athlon -mikroprosessori, jonka kellotaajuus on 1333 MHz ja 512 megatavua muistia). Testiympäristönä on samanlainen Linux-käyttöjärjestelmä kuin kilpailukoneessasikin on (eli Red Hat -pohjainen laitoksen oma CS Linux). Ehdotankin ikävien yllätysten välttämiseksi, että testaat ohjelmaasi myös Linux-ympäristössä, vaikka kirjoittaisitkin sen Windows-ympäristössä!

### 5.1 Lähdekooditiedosto

Kirjoita ratkaisusi yhteen ASCII-lähdekooditiedostoon, jonka nimi riippuu käyttämästäsi ohjelmointikielestä seuraavasti:

KIELI	NIMI
C	<code>rectify.c</code>
C++	<code>rectify.cpp</code>
Pascal	<code>rectify.pas</code>

Laita tiedoston alkuun kommentti, jossa on seuraavat asiat:

- Kilpailun nimi `Datatähti 2004 -loppukilpailu`.
- Oma nimesi ja syntymäaikasi.

- Tämän tehtävän nimi "rectify".
- Ilmoita, kääntyykö ohjelmasi kääntäjän oletusvalitsimilla. Jos ei, niin ilmoita lisäksi, mitkä valitsimet on lisättävä kääntäjän komentoriville. Ilmoita vain sellaiset aivan välttämättömät valitsimet, joita ilman ohjelmasi ei edes käänny!

Tällainen välttämätön lisävalitsin voisi olla esimerkiksi matematiikkakirjastoja käyttävissä C-kielisissä ohjelmissa tarvittava "-lm".

Muuten tiedoston kirjoitusasu on vapaa. Älä kuitenkaan käytä yli 80 merkkiä pitkiä rivejä!

Ohjelmointikielillä C ja C++ kirjoitetut lähdekooditiedostot käännetään kääntäjällä GCC <<http://gcc.gnu.org/>>. Ohjelmointikielillä Pascal kirjoitetut tiedostot käännetään puolestaan kääntäjällä Free Pascal <<http://www.freepascal.org/>>, jonka pitäisi olla Turbo Pascal -yhteensopiva. Käytä kuitenkin vain valitsemasi ohjelmointikielen peruspiirteitä yhteensopivuuden takaamiseksi.

Käännökset tehdään kääntäjien oletusarvoisilla valitsimilla, ellei ole ilmoittanut muuta tiedoston alkukommentissa.

Huomioi vielä seuraavat tekniset vaatimukset:

- Ohjelmasi saa lukea syötettä vain tiedostosta `rectify.in` ja tulostaa vain tiedostoon `rectify.out`!

Erityisesti se *ei* saa lukea mitään näppäimistöltä tai tulostaa mitään näytölle.

- Kun ohjelmasi suoritus päättyy, niin sen täytyy palauttaa käyttöjärjestelmälle poistumisarvonaan 0.

Ohjelmointikielissä C ja C++ tämä tarkoittaa, että pääohjelman `main` (ja koko ohjelman) suorituksen pitää päättyä komentoon `return 0;` (eikä vain "valua" viimeiseen sulkevaan kaarisulkuun saakka).

Ohjelmointikielissä Pascal kääntäjä huolehtii tästä automaattisesti.

Näiden vaatimusten rikkomisesta saattaa seurata pistemenetyksiä testiajojen epäonnistuessa!

## 5.2 Palautustapa

Lähetä kilpailuajan loppuessa seuraavanlainen sähköpostiviesti:

- Vastaanottajana on `matti.nykanen@cs.helsinki.fi`.
- Viestin otsikkona on kilpailun nimi `Datatähti 2004 -loppukilpailu`.
- Viestin tekstinä on oma nimesi ja syntymäaikasi.
- Viestin MIME-liitteenä on lähdekooditiedostosi nimenään `rectify.c`, `rectify.cpp` tai `rectify.pas`.

Jos MIME-liitteen käyttö on mahdotonta, niin lähetä kaksi peräkkäistä viestiä:

1. ensin yhtenä viestinä nämä muut tiedot sekä maininta jatkoviesticistä
2. sitten jatkoviesti, jonka otsikkona on lähdekooditiedostosi tiedostonimi ja sisältönä itse lähdekoodi.