

# 81011 Ohjelmointi I – tentti 14.05.2002



## Tehtävä 1

Vastaa *ensimmäisenä* tämän tehtävän kysymyksiin:

- (a) Kirjoita nimesi ja opiskelijanumerosi *selkeästi* jokaisen palauttamasi paperin vasempaan yläkulmaan. [3 p]
- (b) Kopioi seuraava taulukko siististi päälimmäiselle vastauspaperille nimesi ja opiskelijanumerosi alle siten, että jokainen "ruutu" on kooltaan  $2 \times 2$  konseptipaperin ruutua:

1	2	3	4	5	$\Sigma$
---	---	---	---	---	----------

[3 p]

- (c) Mikäli olet *avoimen korkeakoulun opiskelija* ja olet pyrkimässä teekkariksi *hyvän opintomenestyksen avulla* tai olet *valmistumassa* kuuden viikon kuluessa tentistä, niin piirrä päälimmäisen vastauspaperisi oikeaan yläkulmaan saman kokoinen  $\otimes$ -merkki kuin tämän tehtäväpaperin oikeassa yläkulmassa on. Jos et kuulu kumpaankaan em. kategoriaan, älä piirrä merkkiä. [2 p]

## Tehtävä 2

Halutun päivämäärän *pp.kk.vvvv* nk. juliaaninen päivännumero (*JP*) voidaan laskea algoritmilla:

- Jos kuukausi (*kk*) on tammikuu tai helmikuu, niin vähennä vuodesta (*vvvv*) yksi lisää kuukauteen (*kk*) 12
- Suorita sitten seuraavat laskutoimitukset niin, että kaikki laskutulosten desimaalisosat jätetään huomiotta (l. pyöristetään kohti nollaa: tähän tapahtuu  $C++$ :ssa tietyissä tilanteissa automaattisesti):

$$a = vvvv / 100$$

$$c = 2 - a + a / 4$$

$$e = 365.25 * (vvvv + 4716)$$

$$f = 30.6001 * (kk + 1)$$

$$JP = c + pp + e + f - 1524.5$$

- JP* on haluttu juliaaninen päivännumero.

Toteuta algoritmi  $C++$ -funktiona *Juliaaninen*, joka palauttaa parametrina annettua päivämäärää vastaavan juliaanisen päivännumeron. [6 p]

## Tehtävä 3

Essee: tiedon etsintä (eli tiedon haku). [6 p]

## Tehtävä 4

Selitä lyhyesti (max. 3–5 virkettä) seuraavat käsitteet:

- (a) paikallinen muuttuja, [1 p]  
(b) sitomisjärjestys (l. assosioituminen), [1 p]  
(c) looginen virhe, [1 p]  
(d) tietovirta (stream), [1 p]  
(e) muuttuja- eli viilteparametri ja [1 p]  
(f) ohjelman tulkaaminen. [1 p]

Ota vastatessasi huomioon seuraavat:

- Esimerkki ei yksinään riitä vastaukseksi: anna yleinen selitys.
- Älä selitä kysyttyä termiä sen itsensä (tai sen taivutusmuotojen) avulla.
- Selitä yksikäsitteisesti: jos vastaukseksi voi tulkita väärin, se tulkitaan väärin.

## Tehtävä 5

Oletetaan, että sinulla on prosessoitavana levytiedosto, johon on talletettu kokonaislukuja (yksi luku rivillä). Tiedostossa olevien lukujen määrästä ei ole tarkempaa tietoa: siis kaikkien lukujen lukeminen suureen taulukoon ei ole mahdollista. Sama luku voi myös esiintyä tiedostossa useammin kuin kerran. Esitä algoritmi (luentomonisteen algoritmikielellä, ei  $C++$ :lla), joka selvittää tiedoston *toiseksi suurimman* luvun.

Esimerkiksi, jos tiedostossa olisi luvut (jokainen omalla rivillään):

903, 821, 733, 641, 558, 46, -372, 287, 903 ja -19

antaisi algoritmi tulokseksi 821.

Algoritmin ei tarvitse varautua seuraaviin ongelmatilanteisiin:

- Tiedostossa on vähemmän kuin kaksi lukua.
- Kaikki tiedostossa olevat luvut ovat samoja. [6 p]