

# 81011 Ohjelmointi I – tentti 9.9.2002

## Tehtävä 1

Vastaa *ensimmäisenä* tämän tehtävän kysymyksiin:

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

1	2	3	4	5	$\Sigma$

 [3 p]

## Tehtävä 2

Vuosien 1900 ja 2099 välisenä aikana pääsiäissunnuntain päivämäärä vuonna  $v$  voidaan määrittää seuraavalla algoritmilla:

- $a = v \bmod 19$
- $b = v \bmod 4$
- $c = v \bmod 7$
- $d = (19 * a + 24) \bmod 30$
- $pp = 22 + d + (2 * b + 4 * c + 6 * d + 5) \bmod 7$
- jos  $v$  on 1954, 1981, 2049 tai 2076, niin  
 $pp = pp - 7$
- jos  $pp \leq 31$ , niin  
pääsiäissunnuntai on maaliskuun  $pp$ :s päivä,  
muussa tapauksessa  
pääsiäissunnuntai on huhtikuun  $(pp - 31)$ :s päivä

Tee algoritmin pohjalta c++-funktio *Paasiainen*, joka laskee ja palauttaa parametrina saamansa vuoden pääsiäissunnuntain päivämäärän. Koska päivämäärä koostuu kahdesta arvosta (päivä ja kuukausi), kertaa ensin mielessäsi, kuinka tehdään funktio, joka "palauttaa" useita arvoja. [6 p]

## Tehtävä 3

Essee: tiedostojen käsittely c++-kielellä. [6 p]

## Tehtävä 4

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

- muodollinen parametri, [1 p]
- ylivuoto, [1 p]
- semanttinen virhe, [1 p]
- looginen operaattori, [1 p]
- algoritmi ja [1 p]
- staattinen tietorakenne. [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 vastauksesi 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 eikä sallittua*. 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]