

Pääteohjaustehtäviä 1 / 3.-4.9

1. Pääteohjaukset ja demot

1.1 Pääteohjausten paikka ja aika

Pääteohjaukset pidetään mikroluokassa AgB212.2 ja/tai AgB213.1 (tarkista [Korpista](#)). Ohjaukset alkavat xx.15.

1.2 Pääteohjausten pakollisuus

Pääteohjaukset ovat siinä mielessä pakollisia, että niistä voi olla pois korkeintaan 3 kertaa. Vuonna -06 ohjauksia oli 13 kertaa. 3:en poissaolon jälkeen vähennetään tentissä 1p/poissaolokerta. Eli jos ei tosita yhtään ohjausta, lähtee kokeeseen 10 pistettä muita huonommasta asemasta (tätä voi tietysti kompensoida vähän demoilla)!

1.3 Pääteohjausten palauttaminen sähköisesti

DemoWWW:llä. Katso kurssin WWW-sivut:

<http://www.mit.jyu.fi/vesal/kurssit/winohj07/>.

<http://www.ad.jyu.fi/vesal/kurssit/winohj07/> (varasivut jos pääsivut ei toimi)

<http://users.jyu.fi/~vesal/kurssit/winohj07/> (varavarasivut jos varasivut ei toimi)

Jos on ohjauksessa paikalla, sähköistä palautusta ei tarvita. Jos ei ole paikalla tai tehtävät jäävät pahasti kesken, silloin tarvitaan palautus *DemoWWW*:llä. Huom! *DemoWWW* on konsepti, ei mikään yksittäinen ohjelma, vaikka onkin olemassa *DemoWWW.exe*.

1.4 Demojen tai pääteohjausten tekeminen ryhmässä

Jos demot tai pääteohjaukset tehdään ryhmässä, laittaa jokainen ryhmäläinen oman *DemoWWW*-hakemistonsa kuntoon. *files.txt*-tiedostoon tulee sitten kaikkien tekijöiden nimi:

```
=====
= Demo 1
= Vesa Lappalainen & Markku Vire
=====
```

...

Samoin tietysti ohjelmien kommentteihin tulee kaikkien tekijöiden nimet. Toisten vastausten **kopiointi on kiellettyä** (vaikka *DemoWWW* voikin tehdä kopioinnin "houkuttelevaksi").

2. Työskentely mikrolla ja harjoitustyön näyttäminen

2.1 Työskentele aina kovalevyllä tai verkon levyllä

Käytä työhakemistona **AINA** omaa hakemistoasi KOVALEVYLLÄ, jopa mieluummin kuin VERKON levyllä. Tämä nopeuttaa työskentelyä! Agoran mikroluokissa suositeltu hakemisto on C:\mytemp\tunnus.

ÄLÄ KOSKAAN TYÖSKENTELE SUORAAN KORPULLA TAI USB-tilkulla.

Töiden talletus oman korppusi/verkkolevyn ja kovalevyn välillä on helpointa hoitaa esim. KOPIOI-ohjelmalla tai siihen liitetyillä ajojonoilla (voi yrittää myös Windowsin matkalaukkaa):

```
kopioi *.c *.h a:: /s      - kopioi korpulle vastinpaikkaan
kopioi *.c *.h u:: /s      - kopioi U:-levylle vastinpaikkaan
kopioi *.c *.h a:: /q /s   - "tasaa" korpun ja kovalevyn vastaaviksi
```

2.2 HT:n säilytys omassa WWW-hakemistossa

Kurssin sivuilla on ohjeet miten HT siirretään ja säilytetään WWW-muodossa.

3. C++

```
// \kurssit\winohj\oop\cpp\luku.cpp
#include <iostream.h>

class cLuku {
    int a;
public:
    void set(int n);
    int get(void) const;
};

void cLuku::set(int n)
{
    a = n;
}

int cLuku::get(void) const
{
    return a;
}

int main(void)
{
    cLuku luku;
    luku.set(3);
    cout << "Luku = " << luku.get() << endl;
    return 0;
}
```

Seuraavat voi tehdä esim. *BC 3.1:n* Dos-versiolla (komentoriviltä **bc**), *gcc*:llä tai vaikka *Delphi 2006:lla* (komentoriviltä **d06**) tai *Eclipsellä* (ks:

<http://www.mit.jyu.fi/vesal/kurssit/ohjelmointi2007/materiaali/cpp.html>)

1. Käännä ja aja.
2. Lisää edelliseen ohjelmaan `set` -metodiin tarkistus siitä, ettei luku voi olla alueen `[0,10]` ulkopuolella (esim. lukua ei aseteta lainkaan tai luvuksi asetetaan liian pienille 0 ja liian suurille 10, tai jopa `set` -funktio laitetaan palauttamaan sijoituksen onnistumista kuvaava arvo).
3. Kokeile mitä tapahtuu jos luku on alustamaton (eli ei ole kutsua `a.set`). Kokeile myös dynaamisella (`cLuku *luku = new cLuku();`) muuttujalla.
4. Lisää metodi `lisaa(int i=1);`, joka lisää lukua. Muista lukualue!

5. Lisää luokkaan rakentajafunktio (constructor), joka alustaa luvun 0:ksi tai rakentajan parametrina annetuksi. Muista edelleen lukualue!
6. Kokeile voiko luokan muuttujaan (attribuuttiin) a viitata suoraan! Jollei voi, muuta public-sanan paikkaa ja kokeile uudelleen. Palauta lopulta public paikalleen.
7. Kokeile kirjoittaa get ja set inline-funktioiksi kahdella eri tavalla.
8. Kirjoita pinon käsittelyohjelma, jossa pinoa varten on seuraava luokka:

```
const SIZE=10;
class cPino {
    char alkiot[SIZE];    // Pinoon talletetut alkiot
    int top;              // Pinon pinnan "osoitin"
public:
    cPino();              // Pinon rakentaja (alustus)
    void Push(char ch);  // Laittaa kirjaimen pinon päällimmäiseksi
    char Pop(void);      // Palauttaa pinon päällimmäisen kirjaimen ja
poistaa sen
    int IsEmpty() const; // Palauttaa tiedon siitä, onko pino tyhjä
};
```

9. Kirjoita pääohjelma jossa on kaksi pinoa ja laita pinoihin jotakin ja ota ne sitten pois tulostaen samalla.
10. Mitä täytyisi muuttaa jos pino täytyisi tehdä toiselle tyypille (esim. double)?
11. Entä jos samassa ohjelmassa tarvitaan molempia pinoja yhtä aikaa?