

Tentissä ei saa käyttää ylimääräistä kirjallista materiaalia, laskimia, tietokoneita tai muita lunttausvälineitä.



Palauta kaikki *nimetyt* vastauspaperit *omiin pinoihinsa!*

..... Tehtävät 1. & 2. omalle paperilleen! Nimi paperiin!

1. Selitä (max. 7 riviä/kohta) seuraavat käsitteet ja mitä hyötyä niistä saadaan olio-ohjelmoinnissa. (1 p/kohta)

- | | |
|---|--|
| a) Nimiavaruus (<i>namespace</i>) | d) Virtuaalifunktio (<i>virtual member function</i>) |
| b) Kapselointi (<i>encapsulation</i>) | e) Ennakkoesittely (<i>forward declaration</i>) |
| c) Purkaja (<i>destructor</i>) | f) Käyttötapaus (<i>use case</i>) |

2. Seuraavassa on joukko väittämiä olio-ohjelmoinnista ja C++:sta. Mitkä väittämät ovat oikein, mitkä väärin? Perustele mielestäsi vääristä väittämistä parilla lauseella, *miksi/miten* väittäjä on väärin ja miten asia todellisuudessa on. (6 p)

- ✓ a) Olion tilalla tarkoitetaan olion sisältämää informaatiota.
- ✓ b) Periytymisen "aliluokan olio kuuluu myös kantaluokkaan" tarkoittaa, että aliluokan oliot ovat toiminnallisuudeltaan samanlaisia kuin kantaluokan oliot.
- ✓ c) Vakio-osoittimen (esim. const Kirja*) läpi oliolle saa kutsua vain jäsenfunktioita, jonka parametrit ovat vakio-osoittimia tai -viitteitä.
- ✓ d) Luokan vastuualueella tarkoitetaan sitä osaa ohjelmasta, joka on vastuussa luokan olioiden luomisesta ja tuhoamisesta.
- e) Jos kaksi oliota kuuluu samaan luokkaan, C++:ssa olio pääsee käsiksi toisen olion private-osaan.
- f) UML:n lukumäärämerkinnät assosiaatioissa kertovat, kuinka monta oliota luokasta ohjelman ajon aikana luodaan.

..... Tehtävät 3. & 4. omalle paperilleen! Nimi paperiin!

3. Paperin toisella puolella on kolme luokkaa toteuttava listaus ja pieni pääohjelma.

- a) Ohjelma vuotaa dynaamista muistia. Miten tämä olisi järkevintä korjata? (Jos mieleesi tulee useita järjeviä vaihtoehtoja, luettele ne kaikki) (3 p)
- b) Periytetyyn luokkaan Opiskelija jäsenfunktion annaTiedot uudelleenmäärittely ei toimi. Mikä siinä on vikana ja miten sen voi korjata? (2 p)
- c) Ohjelman kirjoittaja ei ole tainnut kuulla const-sanasta. Korjaa ohjelma tältä osin (tenttipaperiin vain korjatut rivit rivinumeroineen). (2 p)
- d) Onko ohjelmassa vielä C++:n kannalta jotain huomautettavaa (muuta kuin ulkoasullisia tyyli-seikkoja)? (2 p)

4. Paperin kääntöpuolella on osa TTKK:n kursseja ja niiden suorituksia kuvaavaa luokkakaaviota (josta on yksinkertaisuuden vuoksi jätetty itse rajapinnat pois). Vastaa sen perusteella seuraaviin kysymyksiin:

- a) Mitä tarkoittaa Opintojakson ja Kurssin välinen "avoin salmiakki"? (1 p)
- b) Mahdollistaako kaavio sen, että jonkin kurssin opiskelija voisi toimia myös luennoijana? Perustele vastauksesi. (2 p)
- c) Kopioi kaavio vastauspaperille ja lisää luokkien välisiin suhteisiin lukumäärä-merkinnät ja tarvittaessa suunnat. Perustele lyhyesti valintasi. (2 p)
- d) Onko kaaviossa virheitä tai jotain kyseenalaista? (1 p)

8100400 Olio-ohjelmoinnin peruskurssi Tentti 19.1.2004

```
#include <string>
#include <sstream>
#include <iostream>
using namespace std; //HYI!
```

```
5 class Henkilo
{
public:
    Henkilo(string nimi);
10 string annaTiedot();
private:
    string nimi_;
};
15 Henkilo::Henkilo(string nimi) : nimi_(nimi) {}
```

```
string Henkilo::annaTiedot()
{
    return nimi_;
20 }
```

```
class Opiskelija : Henkilo
{
public:
25 Opiskelija (string nimi, string opnro);
    string annaTiedot();
private:
    string opiskelijanumero_;
30 };
```

```
Opiskelija :: Opiskelija (string nimi, int opnro)
: Henkilo(nimi), opiskelijanumero_(opnro) {}
```

```
string Opiskelija :: annaTiedot()
35 {
    return nimi_ + "_" + "(" + opnro_ + ")";
}
```

```
class Hlohallinta
40 {
public:
    Henkilo* tuotaHlo(string nimi, string opnro);
    // Muuta ohjelmointia tarvittavaa tavaraa
};
```

```
45 Henkilo* Hlohallinta::tuotaHlo(string nimi, string opnro)
{
    if (opnro == "") { return new Henkilo(nimi); }
    else { return new Opiskelija(nimi, opnro); }
50 }
```

```
int main()
{
    Hlohallinta hhal;
55 Henkilo* hlo1 = hhal.tuotaHlo("F._Uksi", "182736");
    Henkilo* hlo2 = hhal.tuotaHlo("P._Eelo", "");
    cout << hlo1->annaTiedot() << "\n"
         << hlo2->annaTiedot() << endl;
}
```

