

OHJ-1400 Olio-ohjelmoinnin peruskurssi

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

Muutama sana tenttivastauksen kirjoittamisesta:

1. Vastauksessa olet vastaavasi sellaisen ihmisen esittämään kysymykseen, joka tuntee kohtalaisen hyvin ohjelmistotekniikan aihealuetta muutoin paitsi tämän kysymyksen osalta. Muista että vastauksesi tarkoitus on vakuuttaa tarkastaja siitä, että osaat.
2. Mieti etukäteen esim. ranskalaisilla viivoilla vastauksesi pääkohdat ja lajittele ne johdonmukaiseen järjestykseen — älä kirjoita yhteen pötköön kaikkea mieleen tulevaa.
3. Jos vastaus vaatii ohjelmakoodin kirjoittamista, sen **ei** tarvitse olla pilkulleen syntaksiltaan oikein.

..... Tehtävä 1. **omalle paperilleen!** Nimi paperiin!

1. Seuraavassa on joukko väittämiä olio-ohjelmoinnista ja C++:sta. Mitkä väittämät ovat oikein, mitkä väärin? Perustele mielestäsi vääristä väittämistä parilla lauseella, *miksi/miten* väittäminen on väärin ja miten asia todellisuudessa on.
 - a) Luokan rajapinta on sitä parempi, mitä laajempi se on.
 - b) Luokan vastuualueella tarkoitetaan sitä osaa ohjelmasta, joka käyttää luokan olioita ja on näin niistä vastuussa.
 - c) Jos luokassa jäsenmuuttuja laitettaisiin public-puolelle, voisi sitä muuttaa olion ulkopuolelta ilman, että olio itse sitä huomaa.
 - d) Periytyksen "aliluokan olio kuuluu myös kantaluokkaan" tarkoittaa, että aliluokan olioita ei niiden toiminnallisuuden perusteella voi erottaa kantaluokan olioista.
 - e) Luokkahierarkioiden tarkoituksena on esittää, mitkä luokat käyttävät minkäkin toisen luokan palveluita.
 - f) Määreellä `protected` merkitään luokassa ne jäsenfunktiot ja -muuttujat, joiden halutaan näkyvän aliluokille muttei muualle.
 - g) Kaikki olion tarvitsema tieto talletetaan sen jäsenmuuttujiin.
 - h) UML:n lukumäärämerkinnät assosiaatioissa kertovat, kuinka monta oliota luokasta ohjelman ajon aikana luodaan.
 - i) Jos dynaamisesti `new`lla luodun olion jättää tuhoamatta *deletellä*, se ei haittaa koska ohjelman lopussa käyttöjärjestelmä vapauttaa muistin kuitenkin.

..... Tehtävä 2. **omalle paperilleen!** Nimi paperiin!

2. Selitä lyhyesti (alle 10 riviä) seuraavat olio-ohjelmoinnin käsitteet ja mitä hyötyä niistä saadaan olio-ohjelmoinnissa. *Älä* selitä niistä pelkkää syntaksia tms. vaan kerro etupäässä, mitä ko. käsitteet *tarkoittavat*.
 - a) `const`-sanana merkitys ja käyttötavat C++:lla *olio-ohjelmointia* tehtäessä (tässä vastaus max 15 riviä).
 - b) CRC-kortti (*CRC card*)
 - c) Virtuaalifunktio (*virtual function*)
 - d) Koostumissuhde (*aggregation*)
 - e) Rajapintaluokka (*interface class*)
 - f) Dynaaminen sitominen (*dynamic binding*)
 - g) Purkaja (*destructor*)

KÄÄNNÄ!