

OHJ-1400 Olio-ohjelmoinnin peruskurssi

(Tentaattori Matti Rintala)

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

Muutama sana tenttivastauksen kirjoittamisesta:

1. Muista että vastauksesi tarkoitus on vakuuttaa tarkastaja osaamisestasi.
2. Mieti etukäteen esim. ranskalaisilla viivoilla vastauksesi pääkohdat ja lajittele ne johdonmukaiseen järjestykseen — älä kirjoita yhteen pötköön kaikkea mieleen tulevaa.
3. Muista vastata kaikkiin tehtävän kysymyslauseisiin, sillä täysiä pisteitä ei voi saada jos kaikkiin kysymyksiin asioihin ei ole vastattu.

Palauta kaikki nimetyt vastauspaperit omiin pinoihinsa!

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

1. Alla on selitetty joitain C++:n ja olio-ohjelmoinnin termejä. Mistä termeistä on kysymys? Jos lisäksi selityksessä on jotain vialla, korjaa selitys. Alla on selitetty joitain C++:n ja olio-ohjelmoinnin termejä. Mistä termeistä on kysymys? Jos lisäksi selityksessä on jotain vialla, korjaa selitys.
 - a) Luokan rajapinta, jonka kautta aliluokille voidaan antaa mahdollisuus käyttää jäsenfunktioita ja -muuttujia, joita muut eivät pääse käyttämään.
 - b) C++:n ilmaisu, jonka jälkeen annettuun luokkaan voi tehdä osoittimia, mutta luokan muu käyttäminen ei onnistu.
 - c) Luokka, josta toinen luokka on periyetty joko suoraan tai epäsuorasti usean periytymisen kautta.
 - d) Ne asiat ja toimenpiteet, joita tietyn luokan olioiden on tarkoitus mallintaa ja toteuttaa ohjelmistossa.
 - e) C++:ssä jäsenfunktio, jonka periytetty luokka voi muuttaa mieleisekseen.
 - f) UML:n kaavio, josta käy ilmi ohjelman ajoaikainen käyttäytyminen kokonaisuudessaan.
2. Seuraavassa on muutama essee. Pyri vastaamaan kuhunkin kaikki olennaiset asiat mukaan ottaen.
 - a) C++:n const-mekanismi ja olio-ohjelmointi. Missä kaikkialla const tulee vastaan ja mitä hyötyä siitä saadaan olio-ohjelmoinnissa?
 - b) Keksi ja kuvaa mahdollisimman monta keskenään erilaista tilannetta, jossa oikeanlainen periytymisen käyttö lisää koodin yleiskäyttöisyyttä ja vähentää tarvetta saman koodinpätkän moneen kertaan kirjoittamiseen.
 - c) Olioiden linkaari (luominen ja tuhoutuminen) olio-ohjelmoinnissa ja C++:ssa.

..... **KÄÄNNÄ!**

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

3. Selitä (n. max. 6-7 riviä/kohta) seuraavat olio-ohjelmoinnin käsitteet ja mitä hyötyä niistä saadaan olio-ohjelmoinnissa. *Älä* selitä niistä pelkkää syntaksia tms. vaan kerro etupäässä, mitä ko. käsitteet *tarkoittavat*.

- Abstraktio (*abstraction*)
- Dynaaminen sitominen (*dynamic binding*)
- Abstrakti kantaluokka (*abstract base class*)
- Olion tila (*object's state*)
- Kopiorakentaja (*copy constructor*)
- Koostumissuhde (*aggregation*)

4. Alla on UML-luokkakaavio, joka kuvaa osaa kuvitteellisen ohjelman rakenteesta.

- Mikä merkitys on kaavion kahdessa yhteysviivoissa olevilla "salmiakeilla"?
- Mikä merkitys on merkinnöillä "{abstract}"?
- Mikä merkitys on nuolenkärjellä yhteysviivassa, joka on kiinni luokassa "Teekkarilakki"?
- Kirjoita kaavion perusteella C++:lla luokkien "Henkilö" ja "Opiskelija" esittelyt *esittelyt* Huom! Siis tosiaan vain luokkien esittelyt (class XX { ... });, ei jäsenfunktioiden toteutuksia. Perustele lyhyesti tekemäsi ratkaisut.

