

Olio-ohjelmoinnin perusteet C++ -kielellä



Kurssin ohjelmointikielenä on C++.

Kurssin alussa kerrataan muutamia C++ -kielen asioita ennen siirtymistä olio-ohjelmoinnin käsitteisiin. Kurssilla ei käsitellä ohjelmoinnin perusteita, joten ennen tämän kurssin käymistä kannattaa käydä esim. kurssi Ohjelmoinnin perusteet C++ -kielellä. Kurssi keskittyy olio-ohjelmoinnin perusteisiin raapaisten myös UML:n perusteita.

Kurssi sisältää 23 ohjelmoinnin harjoitustehtävää sekä 49 monivalintatehtävää. Kurssin suorittaminen kestää arviolta 80-140 tuntia.

Kurssin sisällysluettelo:

1. Johdanto 1.1. Kurssin oppimistavoitteet 1.2. Kurssin sisältö 1.3. Kurssin esitietovaatimukset 1.4. Kirjoittajan alkusanat 1.5. Olio-ohjelmoinnin idea	2. Luokat ja oliot 2.1. Luokka, olio ja niiden suhde toisiinsa 2.2. Luokan kirjoittaminen C++:lla	3. Luokan muodostaminen 3.1. Luokan määrittely 3.2. Näkyvyyttä: public ja private 3.3. Oliotaulukoiden hyödyntäminen 3.4. Kohti harjoituksia
4. Osoittimet, viittausmuuttujat ja dynaaminen muistinhallinta 4.1. Osoitin 4.2. Viittausmuuttuja 4.3. Dynaaminen muistinhallinta 4.4. Kohti harjoituksia	5. Olion elinkaari 5.1. Automaattinen, dynaaminen vai staattinen olio? 5.2. Olion muodostaminen ja muodostimet 5.3. Olion tuhoutuminen ja hävittimet	6. Oliosunnittelu ja UML 6.1. Ohjelmistotuottajat pähkinänkuoressa 6.2. Ohjelmistotuotanto pähkinänkuoressa 6.3. Projektiajattelu ohjelmistotuotannossa 6.4. UML ja olioiden suunnittelu 6.5. CRC-korttien hyödyntäminen luokkien hahmottamisessa 6.6. Suunnittelutyön merkitys ohjelmistotuotannossa 6.7. Rajapintojen suunnittelusta
7. Luokkien väliset suhteet 7.1. Kokoonpanollisen koosteen toteuttaminen 7.2. Jaetun koosteen ja assosiaation toteuttaminen 7.3. Kooste-esimerkki 7.4. Lukumääräsuhteiden toteuttamisesta 7.5. Ennakkoesittely: ratkaisu "muna vai kana" ongelmaan 7.6. this-osoitin luokkien välisten suhteiden toteutuksessa	8. C++:n vakiot rajapintojen määrittelyssä 8.1. Perustietotyypiset vakiot 8.2. Vakio-olioihin 8.3. Vakioviittausmuuttujat ja -osoittimet 8.4. Kun osoittimen arvon pitää olla vakio	9. Periytymisen perusteet 9.1. Periytymisen käsite 9.2. Periytety luokan, eli aliluokan määrittely 9.3. Monimuotoisuus eli polymorfismi 9.4. Abstraktit ylliluokat ja puhtaasti virtuaaliset jäsenfunktiot
10. Kopiointi ja sijoitus 10.1. Olion kopioiminen 10.2. Olion sijoittaminen 10.3. Kopioinnin ja sijoittamisen estäminen	11. Päätösluku 11.1. Mitä opittiin? 11.2. Opiskelisinko lisää C++:aa ja olio-ohjelmointia? 11.3. Muita läheisiä aiheita 11.4. Ohjelmointityylistä 11.5. Java olio-ohjelmointikielenä	