

58160 Ohjelmoinnin harjoitustyö (4 op, 2ov)

Kurssin kotisivu

- <http://www.cs.helsinki.fi/kurssit/perus/58160-8/>
- Oppimistavoitteet, arvosteluperusteet, lista työaiheista ym.

Opetus

- **Yhteisluento I** (2 h) - ennen pienryhmien alkua
 - Kurssin hallinnolliset asiat ja ryhmien muodostaminen läsnäolijoista
 - Preppausta harjoitustyön tekemiseen: mitä ja miten tehdään?
 - Sormiharjoituksia ja työkaluihin tutustumista mikroluokassa
 - Luentokalvot, aiheenvalinta ja muuta materiaalia:
<http://www.cs.helsinki.fi/u/nikander/oht>
- **Yhteisluento II** (2h) - harjoitustyön puolivälissä (työviikon 4 alussa)
 - Ohjelmistojen testauksen alkeet: miten testaat harjoitustyösi?
 - Harjoitus: testiajuriin kirjoittaminen Eclipseä + Junitilla mikroluokassa
- **Harjoitustyöryhmät** (6 työviikkoa)
 - Itsenäistä opiskelua n. 0-8h / päivä oman aikataulun mukaan
 - Koko ryhmän tapaamisia yleensä 2 x 4 h (työviikoilla 3 ja 6)
 - Muina viikkoina henkilökohtainen ohjauspalaveri n. 15-20 min/vko
 - Opetusohjelmassa mainittu kokoontumisaika on suuntaa-antava:
<http://www.cs.helsinki.fi/opinnot/opetusohjelma.html>
– henk. koht. ohjauspalaverin aika sovitaan työn alussa, mieluiten ko. aikavälille
- **Neuvontapäivystys**
 - 3 viimeisen työviikon aikana, 2. krs B- ja C-siipien välissä
 - Neuvontapäivystyksen ajat ilmestyvät opintoneuvonnan sivulle:
<http://www.cs.helsinki.fi/opintoneuvonta/>

Esitietovaatimukset

Seuraavien kurssien suoritusmerkintä edellytetään:

- **Ohjelmoinnin perusteet + Java-ohjelmointi/Ohjelmoinnin jatkokurssi**
- **Ohjelmistotekniikan menetelmät / Ohjelmistojen mallintaminen**

Suoritukset tarkastetaan opintorekisteristä kurssin alussa. Poikkeuksia ei tehdä puuttuvien tenttien tms. takia. Harjoitustyön voi suorittaa 5 kertaa vuodessa, joten tilaisuuksia riittää. Erikoistapauksissa ota yhteyttä Ohj. harjoitustyön esimieheen (Tomi Pasanen).

Suoritustapa

Kurssi suoritetaan toteuttamalla pieni (n. 1000 riviä) Java-kielinen ohjelmistoprojekti sovitusta aiheesta. Tarkemmat vaatimukset kurssin kotisivulla, ks. myös luentokalvot.

- Työhön kuuluu ohjelman määrittely, suunnittelu, toteutus ja testaus sekä näiden dokumentaatio. Ajettava ohjelma dokumentteineen palautetaan ohjaajalle.
- Edistymisestä raportoidaan ohjaajalle viikoittain n. 20 minuutin seurantalpalaverissa. Palavereissa käsitellään työn rajaukseen, toteutustapaan ja toteutustekniikoihin liittyviä kysymyksiä ja sovitaan seuraavista työvaiheista.
- Työn suorittamisessa, aikataulussa ja ohjeistuksessa on pieniä ryhmäkohtaisia vaihteluja; noudata aina ensisijaisesti oman ohjaajasi ohjeita.

Arvostelu

- Arvosteluskala 0 – 60 p

Pisterajat	30	35	40	45	50
Arvosana	1	2	3	4	5
- Työn osien painotus arvostelussa:
toteutus 40%, dokumentaatio 40%, testaus 10%, projektinhallinta 10%

Keskeyttäminen

Työ on saatava valmiiksi työryhmän työskentelyaikana (6 viikkoa) ja sen on sisällettävä kaikki pakolliset osat. Muutoin kurssi katsotaan keskeytyneeksi. Jatkoaikaa voi saada sairauden takia lääkärintodistuksella. Matkat, työkiireet ym. eivät oikeuta jatkoaikaan. **Kurssi katsotaan keskeytyneeksi myös, jos opiskelija ei viikkoon osallistu ohjaukseen tai muulla tavoin raportoi ohjaajalle edistymisestään.**

Työn keskeyttänyt opiskelija voi päästä kurssille uudestaan vain ns. jonotuslistan kautta. Keskeyttäneet ilmoittautuvat erilliseen jonotusryhmään, josta otetaan opiskelijoita kurssille ilmoittautumisjärjestyksessä ja vain, jos ryhmissä on tilaa. Uudet opiskelijat ovat siis etusijalla ryhmiä muodostettaessa.

Linkkejä

- Ohjelmoinnin harjoitustyön Moodle-sivu**
Sisältää mm. aihekuvaukset ja ohjeita työkalujen käyttöön
<https://courses.cs.helsinki.fi/> --> Ohjelmoinnin harjoitustyö
- TTY:n Java-sivut**
Hyvää kertausmateriaalia ja käytännön pikku harjoituksia (automaattitarkistuksella)
<http://javala.cs.tut.fi/>
- Ohj. perusteet / Java-ohjelmointi -kurssien sivut**
Kertaa ruostuneet ja väliin jääneet aihepiirit esim. täältä.
 - <http://www.cs.helsinki.fi/u/wikla/Ohjelmointi/>
 - <http://www.cs.helsinki.fi/u/tapasane/kurssit/Java-Ohjelmointi/kevat2008/>
- Ohjelmointitekniikka (Java) -kurssin sivu**
Syventävää materiaalia Java-ohjelmoinnista, jota peruskursseilla vain raapaistiin.
<http://www.cs.helsinki.fi/u/wikla/OTJ/>
- Sunin Java-materiaalien koostesivu**
Linkit API-kuvaukseen, tutoriaaleihin, koodiesimerkkeihin, sanastoon ym.
<http://java.sun.com/javase/reference/>
- Jaakko Nenosen linkkilista**
Linkkejä tarpeellisiin ohjeisiin yms.
<http://www.cs.helsinki.fi/u/jnenonen/alabra/linkit.html>
- Sami Nikanderin harjoitustyösivusto**
Lisää sekalaisia linkkejä ja materiaalia
<http://www.cs.helsinki.fi/u/nikander/oht/>

Kurssimateriaalin kehittäminen

Jos kurssin aikana jäät kaipaamaan jotain tietynlaista oppimateriaalia, mainitse siitä esim. sähköpostitse sami.nikander@helsinki.fi tai kerro asiasta kurssipalautteessa.

Ohjelmoinnin harjoitustyö – Pohdittavaa

Käsityksesi kurssin sisällöstä

Millainen on tämänhetkinen käsityksesi ohjelmistotuotannosta? Mitä se pitää sisällään?
Millaisia työvaiheita ja menetelmiä ohjelmistoprojekteissa käytetään?
Mitä ohjelmointi on? Entä koodaaminen? Ovatko ne sama vai eri asia?
Mitä ohjelmistotuotannon aihepiireistä arvelet kurssilla käsiteltävän? Mitä ei käsitellä?

Omat oppimistavoitteesi / arvosanatavoitteesi

Mitä odotat oppivasi tällä kurssilla? Tähtäätkö viitoseen vai riittääkö läpipääsy?
Mitkä asiat kurssin aihepiirissä tuntuvat vaikeimmilta? Listaa joitain haasteellisimpia aiheita.
(Tai: jos arvioit jo osaavasi kurssilla opiskeltavia asioita, pohdi mitä osa-alueita hallitset parhaiten? Mitä kautta olet oppinut kurssin asioita aiemmin, onko osaamisesi itseopiskeltua käytännön kokemuksta vai esim. aiempia opintoja? Onko jokin näkökulma kuitenkin uusi?)

Arvio työmäärästä

Arvioi myös omaa työmäärääsi. Miten arvelet laskennallisen työmäärän (n. 80h) sopivan omaan tilanteeseesi? Tuleeko "sakkokierroksia" vai pääsetkö vähemmällä? Joudutko kertaamaan paljon tai muuten preppaamaan keskimääräistä enemmän? Onko sinulla aiempaa kokemusta koodauksesta, teknisestä kirjoittamisesta tms? Aiotko ottaa kunnianhimoisen vai minimalistisen työaiheen? Miten arvosanatavoitteesi vaikuttaa työmäärään?

Kiinnostavat aihepiirit?

Millainen työaihe voisi olla itseäsi kiinnostava ja motivoiva?
Haluatko tehdä ajanviettohjelman? (esim. peli, puppugeneraattori, tietovisa)
Vai hyötyohjelman? (esim. CD-kortisto, budjettilaskuri, sanastoharjoitus) Vai onko sillä väliä?
Mieti myös haluatko tässä työssä opetella esim. graafisen käyttöliittymän tekemistä (ei pakollista) tai uusia ohjelmointityökaluja (ks. myös tarkistuslista jäljempänä).

Ohjelmoinnin harj. työ – Aikataulusuunnitelma

Tarkastele edellisessä kohdassa arvioimaasi työmäärää (oletuksena n. 80h) tarkemmin.

- **Miten aiot jakaa työmäärän kuudelle työviikolle?**
- **Mitä hommia teet milläkin viikolla?**
- **Onko sinulla muita kursseja, tenttejä tms. kiireitä**, jotka vaikuttavat aikatauluun?
- **Miten arvioit työmäärien painottuvan projektissasi eri vaiheisiin?**
(erityisesti suunnittelu / toteutus / testaus)

Tämä ei ole sitova aikataulu. Arvio on vain alustava ja lähinnä oman työskentelysi tueksi. Sovi viikottaisista tavoitteista ja etapeista ohjaajasi kanssa. Voit kurssin aikana seurata työsi edistymistä ja tarkkailla samalla alkuperäisten arvioitteesi osuvuutta.

Vko 1 Yhteisluento I (työkaluihin tutustumista, projektisuunnitelman tekoa)

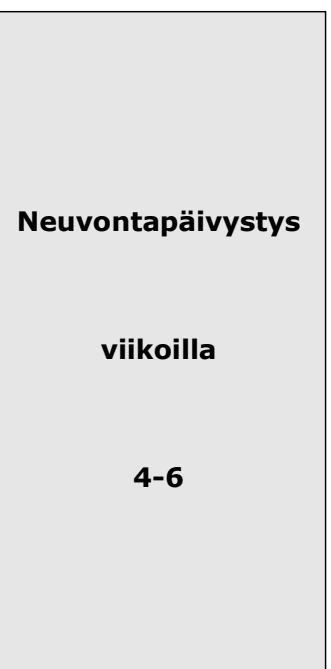
Vko 2

Vko 3 Suunnitelmien esittely omassa ohjausryhmässä

Vko 4 Yhteisluento II (testauksen harjoittelua)

Vko 5

Vko 6



Työn palautus vkon 6 lopussa

Tarkistuslista Ohj. harjoitustyön aloittajalle

Tämän listan tarkoitus on auttaa hahmottamaan kurssilla vastaantulevien asioiden paljoutta. Mitä paremmin osaat ennestään luettelossa mainittuja aihepiirejä, sitä vähemmän joudut nyt käyttämään aikaa kertaukseen ja opetteluun. Vastaavasti jos opeteltavaa kertyy kovin paljon, voit arvioida työmääräsi kasvavan samassa suhteessa. Listan käyttäminen on vapaaehtoista, eikä sen toteutumista seurata; lista on vain oman suunnittelusi tueksi :-)

Esitiedot: Aiemmilta kursseilta tuttua

Nämä asiat sinun oletetaan osaavan jo ennestään, ainakin jossain määrin.

Jos taidot puuttuvat tai ovat ruostuneet, joudut itse opettelemaan tai kertaamaan nämä asiat.

Ohj.perusteet ja Java-ohjelmointi:

Rastita jos tiedät osaavasi tai sitä mukaa kun olet kerrannut:

- Ohjelmoinnin peruskäsitteet (tyypit, lausekkeet, operaatiot, aliohjelmat...)
- Olioiden peruskäsitteet (luokat, kapselointi, periytyminen, rajapinnat...)
- Näkyvyys ja pakkaukset
- Poikkeusten käsittely
- Tiedostojen käsittely: lukeminen, kirjoittaminen
- Kokoelmaluokat, geneeriset tietotyypit (ArrayList<E> ja HashMap<K,V>)

Ohjelmistotekniikan menetelmät:

- Ohjelmistotuotannon vaiheet
- UML-tekniikka, erityisesti
 - luokkakaavio, luokkien yhteydet/moniarvoisuusmääreet
 - sekvenssikaavio
 - kaavioiden yhteys (Java-)toteutukseen
- Yksikkötestaus, JUnit (ainakin käsitteenä)

Johdatus tietojenkäsittelytieteeseen:

Ei pakollinen esitieto, mutta kurssin sisällöstä kannattaa palauttaa mieleen erityisesti nämä ohjelmointiin läheisesti liittyvät ohjenuorat (myös alkuperäisartikkelit saattaisivat avautua nyt paremmin?!).

- Suunnittelun periaatteet: 6.1 Yksinkertaisuus ("ohjelmistohelmet"; suunnittelu ennen toteutusta...)
- Tietojenkäsittelyn käytännöt: 7.1 Ohjelmointi (hyvän ohjelmakoodin ominaisuudet: selkeys, ulkoasu, nimet...)
- Tietojenkäsittelyn käytännöt 7.2 Järjestelmien rakentaminen (ohjelmistotuotannon ongelmat: muuttuvat vaatimukset, rönsyily, olennaisen unohtaminen, määrittelyjen ja projektisuunnitelman puuttuminen...)

Kurssilla opeteltavia uusia asioita

Nämä ovat kurssin keskeistä aihepiiriä, mutta kaikkien ei tarvitse opetella kaikkea juuri nyt. "Oman kurssisi" painotukset muotoutuvat pitkälti työaiheen ja oman kiinnostuksesi pohjalta. Jos osaat näitä jo ennestään, työmääräsi voi olla hieman keskimääräistä pienempi.

Java-tekniikkaa

Osaan jo Opettelen

- | | | |
|--|--------------------------|--------------------------|
| • Javan kokoelmaluokat (Java Collections): java.util | <input type="checkbox"/> | <input type="checkbox"/> |
| • Merkkijonojen, päivämäärien ym. käsittely: java.util | <input type="checkbox"/> | <input type="checkbox"/> |
| • Graafisen käyttöliittymän tekeville: <ul style="list-style-type: none">• Tapahtumaohjelmointi: mm. java.awt.event• Käyttöliittymäkomponentit: mm. javax.swing | <input type="checkbox"/> | <input type="checkbox"/> |
| • Reaaliaikaista vuorovaikutusta (esim. räiskintäpeli) tarvitseville: <ul style="list-style-type: none">• Moniajo, ajastimet, säikeet: mm. java.lang.Thread | <input type="checkbox"/> | <input type="checkbox"/> |
| • Tiedon lataamista verkosta (esim. linkkitarkistin) tarvitseville: <ul style="list-style-type: none">• HTTP-kutsut, URLien käsittely... mm. java.net | <input type="checkbox"/> | <input type="checkbox"/> |
| • Monimutkaisia tiedostoja (esim. asetustiedostot) tarvitseville: <ul style="list-style-type: none">• XML-tiedostojen käsittely mm. javax.xml.* | <input type="checkbox"/> | <input type="checkbox"/> |

Ohjelmistotuotannon taitoja

- | | | |
|--|--------------------------|--------------------------|
| • Teknisten dokumenttien lukutaito (mm. API-kuvaukset!) | <input type="checkbox"/> | <input type="checkbox"/> |
| • Hyvä koodaustyyli | <input type="checkbox"/> | <input type="checkbox"/> |
| • Systemaattinen virheenjäljitys ("debuggaus") | <input type="checkbox"/> | <input type="checkbox"/> |
| • Systemaattinen testaus | <input type="checkbox"/> | <input type="checkbox"/> |
| • Teknisen dokumentin kirjoittaminen | <input type="checkbox"/> | <input type="checkbox"/> |
| • Suunnitelmien ja ohjelmien esittely/demoaminen yleisölle | <input type="checkbox"/> | <input type="checkbox"/> |
| • Ohjelmistotuotteen viimeistely: asennuspaketit, käyttöohjeet | <input type="checkbox"/> | <input type="checkbox"/> |

Omaa työtä helpottavia työkaluja

Näiden käyttöä ei edellytetä kurssilla, mutta opetteluun käytetty vaiva maksaa itsensä moninkertaisesti takaisin viimeistään myöhemmissä opinnoissa – yleensä jo harjoitustyön kuluessa. Hyödyllistä ammatillista tietotaitoa varsinkin pääaineopiskelijoille. Työkaluista mainittu muutamia TKTL:n Linuxeissa toimivia, muitakin on.

Näitä aion opetella:

- | | |
|---|--------------------------|
| • Dokumentaatiotyökalu JavaDoc ("API-generaattori") | <input type="checkbox"/> |
| • Sovelluskehitin eli IDE (Eclipse, NetBeans) | <input type="checkbox"/> |
| • Testikehys JUnit (mm. osana Eclipseä) | <input type="checkbox"/> |
| • UML-piirtotyökalu (Umbrello, Visual Paradigm, ArgoUML...) | <input type="checkbox"/> |
| • Versionhallinta (bzi, svn, cvs) | <input type="checkbox"/> |
| • Käännöksen automatisointityökalu (ant, make) | <input type="checkbox"/> |