

Ohjelmoinnin peruskurssi Y1

CS-A1111

10.9.2018

Esittelyluento: luennon jälkeen

- ▶ Tiedät, miten tämä kurssi on järjestetty ja mitä sinun pitää tehdä, jotta saisit kurssin suoritetuksi.
- ▶ Tiedät, mitä tarkoitetaan ohjelmoinnilla.
- ▶ Osaat kirjoittaa yksinkertaisen tietokoneohjelman.

Mihin teekkari / diplomi-insinööri tarvitsee ohjelmointia?

Mihin teekkari / diplomi-insinööri tarvitsee ohjelmointia?

- ▶ Pienet laskentasovellukset
- ▶ Mittauslaitteen tms. ohjaaminen
- ▶ Matriisilaskenta- ja taulukkolaskentaohjelmia käyttäminen
- ▶ Valmiiden ohjelmien toiminnan opettelu ja ymmärtäminen
- ▶ Työelämässä paljon tehtäviä, joissa vaaditaan sekä jonkin insinöörialan tuntemus että hyvä ohjelmointitaito.

Opiskelijoiden kommentteja kesätöiden jälkeen

Opiskelija 1 (ELEC):

Kesätöissä minulle on ollut apua ohjelmointikursseista. Vaikka käytin töissä toista ohjelmointikieltä, Python- ja C-peruskursseilla opetellut ohjelmoinnin peruskäsitteet ja toimintavat ovat olleet ehdottomasti hyödyllisiä.

Opiskelija 2 (ELEC):

Ohjelmoinnin osaaminen oli työssäni tärkeä esitieto. Vaikka en ollut ennen kesää koodannut lainkaan VBA:lla, oli kielen opetteleminen helppoa, koska olin kursseilla koodannut Pythonilla ja C/C++:lla.

Esimerkki harjoitustyöstä, jossa tarvitaan ohjelmointia

- ▶ Kurssi: ENE-58.129 Rakennusten energiatalous (aikaisemmalta vuodelta)
- ▶ Annettu: Tiedot rakennuksen koosta, rakenteesta, sijainnista, lämmitysjärjestelmästä, eri asioiden kustannuksista jne.
- ▶ Tehtävä (osa harjoitustyöstä): Hae optimaalinen yhdistelmä seuraaville viidelle parametrille
 1. seinien eristyspaksuus
 2. yläpohjan eristyspaksuus
 3. ikkunoiden U-arvo
 4. lämpöpumpun kompressorin nimellisteho
 5. höyrystymislämpötila

siten, että seinämien lämpöeristysten, ikkunoiden ja lämmöntuottojärjestelmän muodostaman investoinnin sekä rakennuksen lämmityskustannusten yhteensä muodostamat elinkaarikustannukset saavat minimiarvon tarkasteluaikana ja rakennus on energiatehokkuusluokassa A.

Tämä kurssi

- ▶ Oppimistavoitteena ohjelmoinnin perusasiat
- ▶ Ohjelmointikieli Python
- ▶ Mahdollisia jatkokursseja:
 - ▶ CS-A1121 Ohjelmoinnin peruskurssi Y2
 - ▶ CS-A1130 Tietotekniikka sovelluksissa
 - ▶ CS-A1141 Tietorakenteet ja algoritmit Y
 - ▶ ELEC-A7100 C-ohjelmoinnin peruskurssi
- ▶ Mahdollista suorittaa myös tietotekniikan sivuaine.

Vaihtoehtona toinen ohjelmointikurssi

- ▶ Tämän kurssin sijaan on mahdollisuus suorittaa selvästi vaativampi kurssi CS-A1110 Ohjelmointi 1.
- ▶ Y1:llä pääpaino on pienten kokonaisten ohjelmien kirjoittamisessa, Ohjelmointi 1:llä taidoissa, joita tarvitaan suurten ohjelmien tekemisessä.
- ▶ Jos haluaa kandidaatin tutkinnon suorittamisen jälkeen hakea tietotekniikan maisteriohjelman, niin vaatimuksena on Ohjelmointi 1:n suoritus.
- ▶ Kandidaatin tutkintoon voi sisällyttää molemmat kurssit CS-A1111 Ohjelmoinnin peruskurssi Y1 ja CS-A1110 Ohjelmointi 1 (toinen menee silloin vapaavalintaisiin opintoihin).

Oletko ohjelmoinut aikaisemmin?

- ▶ Vastaa kyselyyn sivulla <http://presemoo.aalto.fi/y1s2018>

Yleistä

- ▶ Henkilökunta
 - ▶ Luennoija Kerttu Pollari-Malmi, vastaanotot keskiviikkoisin klo 10:00–11:00 tietotekniikan talon huoneessa A141.
 - ▶ 20 tuntiopettajaa.
- ▶ Tiedotus
 - ▶ Kurssin kotisivu MyCourses-järjestelmässä
 - ▶ Keskusteluryhmä Piazzassa
<https://piazza.com/aalto.fi/fall2018/csa1111>
- ▶ Mahdollisuus antaa palautetta harjoitustehtävistä A+-järjestelmässä.

Opetusmuodot

- ▶ Luennot
 - ▶ ma 11.9. klo 12-14 ja sen jälkeen ke 14-16 ensimmäisellä viikolla Aalto-salissa, toisella viikolla B-salissa ja sen jälkeen salissa T1.
 - ▶ Luennoilla ei käsitellä kaikkia asioita.
- ▶ Harjoitukset, joissa opiskelijat ratkaisevat harjoitustehtäviä tietokoneella.
 - ▶ 18 ryhmää viikossa Tietotekniikan talon salissa T7.
 - ▶ Lisäksi kaksi ylimääräistä ryhmää to 13.9. klo 14:15-16 ja 16:15-18 Tietotekniikan talon salissa C106.
 - ▶ Läsnäolo ei ole pakollista eikä siitä saa lisäpisteitä.

Opetusmuodot jatkuu

- ▶ Oppimateriaali
 - ▶ A+-järjestelmässä oleva oppimateriaali (linkki MyCourses-sivulla)
 - ▶ Perusteellisempaa materiaalia haluaville suositellaan *Gaddis: Starting Out with Python, 2nd Ed. tai uudempi*, Pearson.
 - ▶ Lisäksi lisämateriaali (ks. MyCourses, alisivut Materiaalit)

Pakolliset osasuoritukset

- ▶ Tenti
 - ▶ Ensimmäinen mahdollisuus pe 30.11. klo 9:00–12:00.
- ▶ Harjoitustehtävät
 - ▶ 9 kierrosta, joista 8 ensimmäistä + esitietotesti pakollisia.
 - ▶ Palautetaan A+-järjestelmässä automaattisille tarkastajalle.
 - ▶ Jos haluaa harjoitustehtävistä vähintään arvosanan 3, on myös viimeisestä kierroksesta saatava minimipisteet.
- ▶ Lisätehtävät (Mastery Grids – MG)
 - ▶ Näiden avulla voit harjoitella lisää kurssin asioita 20.9. alkaen. Mahdollisuus saada 600 lisäpistettä harjoitustehtäviin.
- ▶ Kurssiarvosana on tenttiarvosanan ja harjoitusarvosanan keskiarvo, jos molemmat osasuoritukset on hyväksytty.

Pakolliset osasuoritukset, jatkuu

- ▶ Paikkausmahdollisuudet
 - ▶ Rästitentit (**Pakollinen ilmoittautuminen viimeistään viikko ennen tenttiä**).
 - ▶ Harjoitustehtävien rästikierros (vain kierrosten minimipisteiden saavuttamiseen).

Tärkeä tieto rästikierroksesta

Rästikierroksella voi saada vain vaaditut minimipisteet sellaisesta kierroksesta, josta ei ole saanut riittävästi pisteitä. Rästikierroksella ei voi korottaa hyväksytyjen kierrosten pistemäärää.

Tarvittavat ATK-tunnukset

- ▶ A+-järjestelmään kirjautumiseen tarvitaan Aalto-tunnus.
- ▶ Kurssin Piazza-keskusteluryhmään luodaan tunnus Aalto-sähköpostiosoitteen avulla.

Yhteistyö harjoitustehtävissä

- ▶ Tehtävien ratkaisusta saa keskustella muiden opiskelijoiden kanssa, mutta jokainen kirjoittaa oman koodinsa itse.
- ▶ Kaveria saa neuvoa, mutta valmista tai lähes valmista ratkaisua ei saa luovuttaa kaverille.
- ▶ Keskustelu tai neuvonta ei saa olla niin yksityiskohtaista, että molemmilla ohjelmissa täsmälleen samat käskyt samassa järjestyksessä.
- ▶ A+-materiaalissa, luennoilla ja oppikirjassa esitettyä koodia saa käyttää apuna.

Arvio kurssin työmäärästä

- ▶ Kurssin laajuus on 5 op, joka vastaa noin 133 tuntia työtä.
- ▶ Arvioitu työmäärä yhden kierroksen suorittamiseen on noin 12–14 tuntia:
 - ▶ Kierroksen aihepiiriä käsittelevä luento 2 h
 - ▶ Kierroksen materiaalin lukeminen 2 h
 - ▶ Kierroksen tehtävien tekeminen 8–10 h
- ▶ Lisäksi
 - ▶ Aloitusluento 2 h.
 - ▶ Kertaustenttiin 10 h.
 - ▶ Tenttiin osallistuminen 3 h.

Mikä on tietokoneohjelma?

Keittokirjan resepti

1. Vatkaa munat ja sokeri vaahdoksi.
2. Lisää jauhot ja sekoita.
3. Kaada vuokaan ja paista 20 minuuttia.

Tietokoneohjelma

1. Pyydä rahasumma markkoina ja lue se.
2. Jaa luettu rahasumma 5.94573:lla.
3. Tulosta jakolaskun tulos kuvaruudulle.

Erilaisia tapoja esittää ohjelma tietokoneelle

- ▶ Käskyt esitetään konekielellä (bittijonoina).
- ▶ Symbolinen konekieli
- ▶ Lausekieli, esim. Python, Java, C, C++, Scala, Pascal jne.

Kääntäjä ja tulkki

- ▶ Tietokoneen prosessori ymmärtää vain konekielellä annettuja käskyjä. Muussa muodossa annettu ohjelma pitää muuttaa konekielelle.
- ▶ Muuttaminen voidaan tehdä toisella tietokoneohjelmalla, kääntäjällä tai tulkilla.
- ▶ Kääntäjä ottaa koko ohjelman, kääntää sen konekielelle ja tallentaa konekielisen ohjelman tiedostoon.
- ▶ Tulkki muuttaa käskyn kerrallaan konekielelle ja suorittaa käskyn heti. Sen jälkeen se siirtyy ohjelman seuraavaan käskyyn.

Kolme eri tapaa kirjoittaa ja ajaa Python-ohjelmia

- ▶ Käynnistä Python-tulkki ja anna sille käsky kerrallaan.
- ▶ Kirjoita ohjelma tiedostoon ja aja se komentotulkissa käskyllä `python3 tiedosto.py`
- ▶ Käytä työkalua, jolla voi sekä kirjoittaa että ajaa Python-ohjelmia, tällä kurssilla Eclipse.

Ensimmäinen Python-ohjelma

```
print("Hyvaa paivaa")
```

Muuttujat

- ▶ Muuttujia käytetään ohjelman käsittelemien arvojen tallentamiseen ja käsittelyyn.
- ▶ Muuttujalle voi antaa arvon sijoituskäskyllä.

```
nimi = "Matti"  
print("Hei,", nimi)  
print("Tervetuloa Python-kurssille!")
```

Esimerkkiajo

```
Hei, Matti  
Tervetuloa Python-kurssille!
```

Arvon pyytäminen käyttäjältä

- ▶ Käyttäjän antaman arvon voi lukea `input`-käskyllä.
- ▶ Käskyn sulkujen sisään kirjoitetaan käyttäjälle annettava kehoite.
- ▶ Luetun arvon voi antaa arvoksi muuttujalle sijoituskäskyllä.

```
nimi = input("Kerro nimesi: ")  
print("Hei,", nimi)  
print("Tervetuloa Python-kurssille!")
```

Ajoesimerkki

```
Kerro nimesi: Minna  
Hei, Minna  
Tervetuloa Python-kurssille!
```


Lukuarvon lukeminen käyttäjältä

- ▶ Käsky `input` antaa aina luetun arvon tekstinä, merkkijonona.
- ▶ Jotta luettua arvoa voitaisiin käsitellä lukuna, se pitää ensin muuttaa kokonais- tai desimaaliluvuksi.
- ▶ Jos luettu merkkijono on muuttujan `rivi` arvona, saadaan sitä vastaava kokonaisluku muuttujan `luku1` kokonaislukutyypiseksi arvoksi seuraavasti:

```
luku1 = int(rivi)
```

- ▶ Vastaava desimaaliluku saadaan muuttujan `luku2` float-tyypiseksi arvoksi seuraavasti:

```
luku2 = float(rivi)
```

Esimerkki: eurolaskuri

```
print("Muutan markat euroiksi.")
rivi = input("Anna rahasumma markkoina: ")
markat = float(rivi)
eurot = markat / 5.94573
print("Se on", eurot, "euroa.")
```

Esimerkkiajo

```
Muutan markat euroiksi.
Anna rahasumma markkoina: 543.50
Se on 91.4101380318 euroa.
```