

Kemiantekniikan ohjelmointikurssi CHEM-A2600 (5 op)

Antti Karttunen

Kesäkurssin 2021 aloitusluento 2021-05-31

Tervetuloa! Luento alkaa klo 10.15

Alkukysely: premo.aalto.fi/a2600

Osa 1: Kurssin kuvaus



Kurssin opettajat (kesä 2021)



Antti Karttunen
(lomalla 25.6–30.7.)



Riku Holopainen

Kesäkurssin 2021 yhteenveto

- Kurssi kestää kaksitoista viikkoa (31.5.–23.8.)
 - Kurssin suorittaminen on mahdollista myös nopeammassa tahdissa
 - Kurssi järjestetään 100% etäopetuksena
- Kaksi viikkoa = yksi kierros
 - Jokaisella kierroksella on omat aiheensa ja niihin liittyviä ohjelmointitehtäviä
 - Kurssin arviointi perustuu ohjelmointitehtäviin
- Ohjelmointitehtävät on toteutettu MyCoursesin Quiz-toiminnolla
 - CodeRunner-niminen ohjelmisto tarkistaa tehtävät automaattisesti
 - Tehtäviä voi tehdä omalla tietokoneella missä tahansa ja milloin tahansa
- Ohjausta saa kurssin Slack-työtilassa ja Zoom-etäharjoituksissa
 - Ajankohdat päätetään kurssin alussa opiskelijakyselyn perusteella

Kurssikalenteri

Kurssin aloitusluento maanantaina 31.5. klo 10.15 (Zoom)

Kierros	Päättyy (klo 16.00)	Teema
1	14.6.	Pythonin perusteet. Tulostaminen, syötteen lukeminen, muuttujat, operaattorit ja kontrollirakenteet (if-elif-else, while, for)
2	28.6.	Funktiot, moduulit HUOM! MyCourses-huoltotauko ma 21.6. klo 9–16!
3	12.7.	Tietorakenteet (listat, monikot, sanakirjat)
4	26.7.	Numpy ja Matplotlib
5	9.8.	Tiedostojen käsittely, virheen käsittely
6	23.8.	Sovellukset, Scipy, olio-ohjelmoinnin perusteita

Kaikki tehtäväpaketit avautuvat 18.6. mennessä

Slack ja ohjatut harjoitukset

- Kurssilla on käytössä Slack-työtila, jossa on tarjolla ohjausta läpi kesän
 - Ohjeet MyCourses -> Slack-ohjeet.
 - Kysymyksiä voi lähettää milloin tahansa, mutta päivystys keskittyy arkipäiviin klo 9–16.
 - Tehtäväkierrosten deadline-päivät ovat yleensä ruuhkaisia, joten on suositeltavaa kysyä apua hyvissä ajoin ennen deadline-päivää!
- Ensimmäiset Zoom-etäharjoitukset keskiviikkona 2.6. klo 15:00–16:00
 - Linkki kurssin MyCourses-sivulla
 - Ohjausta esimerkiksi kurssilla tarvittavien ohjelmistojen asennukseen ja ohjelmointitehtävien tekemiseen
 - Harjoituksissa on mahdollista saada henkilökohtaista neuvontaa, opettaja voi siirtää keskustelun erilliseen tilaan (Breakout room)
- Muiden etäharjoitusten ajankohdat päätetään opiskelijakyselyn perusteella
 - Jos olet kiinnostunut etäharjoituksista, vastaa 4.6. mennessä kurssin MyCourses-sivulla olevaan Etäharjoitusten aikataulukyselyyn

Tehtävät ja arvostelu

- Kurssin arvostelu perustuu ohjelmointitehtäviin
- Jokainen kierros sisältää kahdentyyppisiä tehtäviä: A- ja B-tehtävät
 - A-tehtävät ovat perustason tehtäviä
 - B-tehtävät ovat syventäviä tehtäviä
- Kurssin arvostelussa sekä A-tehtävien että B-tehtävien painoarvo on 50%
- **Läpikäysraja:** vähintään 50% A-tehtävien kokonaispisteistä **JA** vähintään 25% A-tehtävien pisteistä jokaiselta kierrokselta
- Esimerkkejä siitä, miten tietyn arvosanan voi saavuttaa:
 - 2: vähintään 75% A-tehtävien pisteistä ja noin 25% B-tehtävien pisteistä
 - 3: vähintään 80% A-tehtävien pisteistä ja noin 50% B-tehtävien pisteistä
 - 4: vähintään 90% A-tehtävien pisteistä ja noin 70% B-tehtävien pisteistä
 - 5: vähintään 90% A-tehtävien pisteistä ja noin 90% B-tehtävien pisteistä

Ajankäytöstä

- 5 op = 5 x 27 h = 135 h
- Ohjattu opetus (0–25 h):
 - Aloitusluento (1 h)
 - 12-24 h vapaaehtoisia ohjattuja etäharjoituksia (12 x 1-2 h)
- Oma työskentely (110–135 h):
 - Harjoitustehtävät 60 h (6 x 10 h)
 - Kurssin verkkomateriaaleihin perustuva omatoiminen opiskelu
- Kurssin läpäiseminen vaatii aktiivista omatoimista työskentelyä
 - *Ohjelmoimaan oppii vain ohjelmoimalla!*
 - Aloita ohjelmointitehtävien tekeminen aina ajoissa
- Ohjelmointitehtävien vaativuus ja kurssin vaatima ajankäyttö kasvaa loppua kohti!
 - Aiempien vuosien perusteella vaativimmat kierrokset ovat kierrokset 3 ja 5.

Ohjelmointitehtävistä

- Ohjelmointitehtävät palautetaan yksilösuorituksina
- CodeRunner-ohjelma pisteyttää ohjelmointitehtävien palautukset automaattisesti
- Tehtävistä on mahdollista keskustella opettajien kanssa ohjatuissa harjoituksissa
- Näin isolla kurssilla opettajilla ei ole mahdollisuutta vastata yksittäisiin sähköpostikysymyksiin tehtävistä.
 - Jos kuitenkin löydät mielestäsi virheen tehtävästä, voit laittaa sähköpostia antti.karttunen@aalto.fi
- Tehtävistä saa keskustella myös kaverin kanssa
 - Valmiita ratkaisuja ei saa jakaa (miksi veisit kaverilta oivaltamisen ilon?)
 - Kurssilla on käytössä automaattinen vastausten samankaltaisuuden arviointi, josta lisää seuraavalla sivulla

Palautusten samankaltaisuuden arviointi

- Ohjelmointitehtävät palautetaan yksilösuorituksina
- Kaikki kurssin tehtäväpalautukset tallentuvat tietokantaan
- Opiskelijoiden palautuksia vertaillaan toisiinsa automaattisesti jokaisella kierroksella
 - Palautuksia vertaillaan myös aiempien vuosien palautuksiin
- Yksinkertaisissa tehtävissä ratkaisut ovat luonnollisesti usein samankaltaisia
- Laajemmissa tehtävissä samankaltaisuuksia ei pitäisi esiintyä, koska meillä kaikilla on hieman erilainen tapa ratkaista ohjelmointiongelmia
- Jos automaattinen tarkistus löytää palautuksista samankaltaisuuksia:
 - Käymme palautukset tarkemmin läpi ja selvitämme tilanteen opiskelijoiden kanssa

OHJELMOINTIKURSSIN HUONEENTAULU

Kun kaverisi on jo ratkaissut ohjelmointitehtävän, jota itse olet vielä tekemässä:

**ON SALLITTUA, että näytät koodiasi kaverille, joka neuvoo sinua ongelmatilanteessa.
EI OLE SALLITTUA, että katsot kaverin valmista ratkaisua.**

Kuva MyCourses-profiiliin

- Opettajien kannalta on aina mukavaa, jos nimet yhdistyvät kasvoihin
 - Etäopetuksessa tämä on jopa tavallista tärkeämpää, koska emme voi tutustua teihin lähiopetuksessa.
- Olisi hienoa, jos voisitte lisätä MyCourses-profiiliinne kasvokuvan!

The image illustrates the process of adding a profile picture in MyCourses. It is divided into three main sections:

- Navigation Menu:** On the left, a vertical menu is shown with the user's name "Antti Karttunen" and profile picture at the top. The "Profile" option is highlighted, and a red arrow points from it to the profile page.
- Profile Page:** The middle section shows the "Dashboard / Profile" page. Under "User details", the country is listed as "Finland" and the city/town as "Espoo". A red box highlights the "Edit profile" button, with a red arrow pointing down to the profile picture section.
- Profile Picture Upload:** The bottom section shows the "User picture" area. It displays the "Current picture" and a "Delete picture" button. Below is a "New picture" section with a "Files" area. A red box highlights the "Files" area, which contains a blue arrow pointing down and the text "You can drag and drop files here to add them."

Osa 2:

Yleistä asiaa ohjelmoinnista

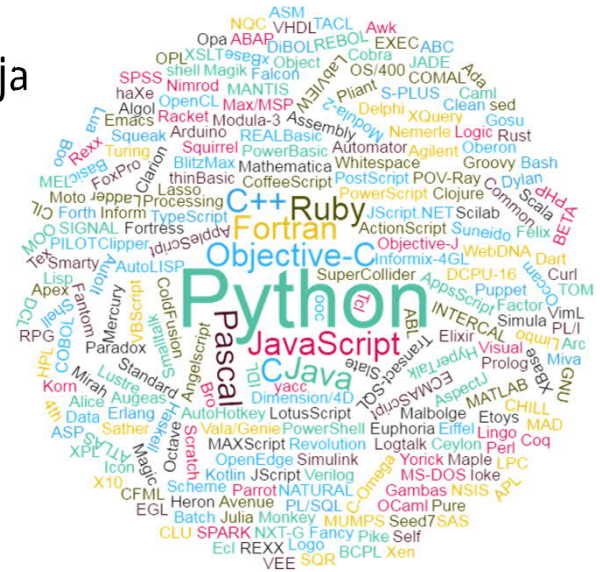


Ohjelmointi ja kemian tekniikka

- Kemian tekniikassa ohjelmoinnilla on merkittävä rooli. Esimerkkejä:
 - Kemian teollisuuden prosessien ohjaus
 - Prosessien simulointi (mallinnus): kemiallinen prosessi mallinnetaan ensin tietokoneella ja siirrytään vasta sitten kalliimpaan kokeelliseen työhön
 - Erilaisten kemiallisten tietokantojen tehokas käyttö vaatii ohjelmointia: materiaali- ja kemikaalitietokannat, rakenne-ominaisuustietokannat
 - Mittausdatan edistynyt käsittely vaatii usein ohjelmointia (tai ainakin se on huomattavasti tehokkaampaa ohjelmoimalla kuin käsityönä Excelillä)
 - *Keksitkö lisää esimerkkejä?*
- Kemiallisissa tieteissä ohjelmoinnilla keskeinen rooli monilla osa-alueilla
 - Laskennallinen kemia, mittausdatan käsittely, spektroskopia, tiedonlouhinta (*data mining*), koneoppiminen (*machine learning*), ...

Mitä ohjelmointi on?

- Ohjelmoinnissa luodaan tarkka toimintaohje, eli "ohjelma"
 - Ohjelman käskyjen avulla tietokone suorittaa tarvittavan tehtävän
- Ohjelma täytyy kirjoittaa *ohjelmointikielellä*, joita on satoja
 - Alkuaikoina ohjelmat kirjoitettiin keskusprosessorin konekielellä (esim. bittijonoina 011001...)
 - Nykyisin käytetään pääasiassa korkeamman tason lausekieliä (kuten Python, Java, C++)
- *Minkä ohjelmointikielen opettelen?*
 - Mikä tahansa moderni lausekieli on hyvä valinta
 - Tärkeintä on oppia ohjelmoinnissa tarvittava *ajattelutapa*.
 - Ajattelutavan opittuasi uusien ohjelmointikielten oppiminen on varsin helppoa
- Tällä kurssilla käytetään Python-ohjelmointikieltä



Kuva: AJK/wordclouds.com

Python-ohjelmointikielen etuja

- Selkeä "kielioppi" (*syntax*). Pythonilla kirjoitettu ohjelma, "koodi", on siis varsin luettavaa, ainakin jos hyviä ohjelmointikäytäntöjä on noudatettu
- Erittäin matala aloituskynnys. Aloittelijakin saa heti aikaan näkyviä tuloksia
- Python-tulkki itsessään on *vapaa (free)* ohjelmisto, eli sen lähdekoodi on kaikkien saatavilla. Tämän seurauksena sen voi hankkia ilmaiseksi ja kuka tahansa voi ehdottaa siihen parannuksia.
- Pythonille on saatavissa valtava määrä vapaita *kirjastoja (library)* joita voi hyödyntää omissa ohjelmissa
- Erittäin vahva ja aktiivinen [yhteisö](#) kehittää Pythonia koko ajan



Mihin Python soveltuu?

- Python soveltuu sekä pieniin että suuriin ohjelmointiprojekteihin
- Pythonilla voi toteuttaa myös suurta laskentatehoa vaativia tieteellisiä ohjelmistoja
 - Tällöin raskas laskenta on syytä hoitaa tieteellisten Python-kirjastojen avulla
 - Esim. NumPy ja SciPy, joita hyödynnetään tällä kurssilla
- Python on ns. *tulkattu* kieli
 - Tulkki (*interpreter*) muuntaa lausekielisen ohjelman ajon aikana konekielelle
 - Tulkatun kielen vastakohta on *käännetty* kieli (kuten C tai C++)



Tietotekniikka sivuaineena?

- Tietotekniikka on erinomainen sivuaine kemian tekniikan opiskelijalle
 - <https://into.aalto.fi/pages/viewpage.action?pageId=23434206>
- CHEM-A2600 vastaa sisällöltään suurelta osin kurssia Ohjelmoinnin peruskurssi Y1 (CS-A1111)
- Seuraava kurssi tietotekniikan sivuaineen “aloittelijan opintopolulla” on Ohjelmoinnin peruskurssi Y2 (CS-A1121)
 - Kurssin CHEM-A2600 suorittaneet voivat aloittaa sivuaineen suoraan kurssista Ohjelmoinnin peruskurssi Y2.
 - Y2-kurssilla on esitietotesti, jolla varmistetaan Y1-kurssin tasoinen osaaminen
 - CHEM-A2600 –kurssilta täytyy olla vähintään arvosana 3, jotta Y2-kurssille jatkamista voi harkita

Osa 3:

Kurssilla käytettävät työkalut



Oppimateriaali

- Kurssin Python-oppimateriaali on toteutettu MyCourses-kirjana
 - Sisältää myös linkkejä myös muihin Python-oppimateriaaleihin
- Oppimateriaali sisältää paljon pieniä harjoittelutehtäviä. Näitä ei pisteytetä, mutta ne tukevat varsinaisten ohjelmointitehtävien tekemistä. Kannattaa kokeilla!

Esimerkki
oppimateriaalin
harjoittelutehtävästä:

Tehtävä 1.6.1

Täydennä if-elif-else -lause vetämällä sanat oikeille paikoilleen

```
paine = float(input("Anna renkaan paine (bar):\n"))
```

```
if paine <= 0.0:  
    print("Virheellinen paine")
```

```
    paine <= 5.0:  
        print("Turvallinen paine")
```

```
    paine <= 7.0:  
        print("Rajoilla ollaan")
```

```
    :  
        print("Rengas räjähti")
```

elif else elif

✓ Check

Anaconda ja Spyder

- Kurssilla hyödynnämme *Anaconda*-nimistä Python-jakelupakettia (*distribution*)
- Anacondan perusversio on ilmainen (Windows, Mac, Linux)
 - Asennusohjeet oppimateriaalissa (luku 8.1)
- Anaconda-jakelupaketti sisältää
 - Python-*tulkin* ohjelmien ajamista varten
 - Valtaavan määrän *kirjastoja* joita voi hyödyntää omissa ohjelmissa
 - Eriyisen kattavan kirjastokokoelman *tieteellistä ohjelmointia* varten
 - *Spyder*-ohjelmointiympäristön, jota suositlemme ohjelmien kirjoittamiseen ja testaamiseen tällä kurssilla
 - Spyderin käyttöohjeet oppimateriaalissa (luku 8.2)
- Kurssilla saa käyttää myös muita ohjelmointiympäristöjä kuten PyCharmia
 - Kurssilla kelpaavat Pythonin versiot 3.6 ja uudemmat

Coderunner-tehtävät

- Voit halutessasi harjoitella CodeRunnerin käyttöä ensin testitehtävien avulla
 - MyCourses -> CodeRunner-testitehtävät

Question 1

Flag question Marked out of 1.00 Not complete

1. Tehtävänanto

Kopioi seuraavat kaksi riviä alla olevaan tekstiruutuun ja paina *Tarkista/Kontrollera/Check* niin näet, miten CodeRunner-tarkastusohjelma toimii.

```
print("Moi!")  
print("Hei!")
```

2. Ensimmäisen kierroksen tehtävissä virheellisistä

Answer: (penalty regime: 0 %) palautuksista ei tule miinusta. Tämä muuttuu myöhemmin.

3. Kopioi Spyderissä kirjoittamasi ja testaamasi koodi tähän tekstiruutuun ja klikkaa *Tarkista / Kontrollera / Check*

Check

4. Vastattuasi kaikkiin tehtäviin klikkaa alalaidasta *Lopeta tehtävä / Avsluta test... / Finish Attempt...* (tämä ei ole pakollista, tehtävät palautetaan automaattisesti kun aikaraja umpeutuu)