

# CS-A1111 Ohjelmoinnin peruskurssi Y1, Tentti 22.10.2019

**Tärkeitä ohjeita vastausten kirjoittamiseen:** Kun kirjoitat ohjelmakoodia, käytä kahden ruudun levyisiä sisennyksiä. Jos sisennyksiä ei ole käytetty tai niistä ei saa selvää, vähennetään siitä pisteitä. Kirjoitettavaan ohjelmakoodiin ei tarvi lisätä kommentteja. Missään tehtävässä tulostusta ei tarvitse muotoilla. Voit myös olettaa, että käyttäjän antama syöte on virheetöntä, ellei tehtävässä erikseen käsketä käsittelemään virhetilanteita.

Opiskelijat, joiden äidinkieli ei ole suomi, saavat käyttää sanakirjaa, jos siinä ei ole merkintöjä (tentinvalvoja tarkistaa sanakirjan). Nämä opiskelijat saavat halutessaan sekä suomen- että englanninkielisen kysymyspaperin.

**Tentissä ei saa käyttää laskimia eikä lisämateriaalia.**

## 1. Funktiot ja tulostus (25 p)

a) Mitä tämä ohjelma tulostaa? (2p)

```
def main():
    matka = 3100
    if matka > 2800:
        print("Cooperin testin tulos on erinomainen.")
    if matka > 2200 and matka <= 2800:
        print("Cooperin testin tulos on hyva.")
    else:
        print("Cooperin testin tulos on huono.")
main()
```

b) Mitä tämä ohjelma tulostaa? (2p)

```
def funktio1(muuttuja):
    arvo = 5
    return 2 * muuttuja

def funktio2(arvo):
    return 3 * arvo

def main():
    arvo = 2
    muuttuja = 11
    tulos = funktio1(arvo)
    print(tulos)
main()
```

c) Mitä tämä ohjelma tulostaa? (3p)

```
def main():
    lista = [10, 20, 30, 10, 40]
    tulos = 100
    i = 0
    while i < len(lista):
        if lista[i] < 30:
            tulos = tulos - lista[i]
            i = i + 1
    print(tulos)
main()
```

d) Mitä tämä ohjelma tulostaa? (4p)

```
def vaihda(luku, arvot):
    arvot[2] = 10 * arvot[0]
    luku = 17
    return luku

def main():
    lukumaara = 4
    lista = [3, 12, 8]
    vaihda(lukumaara, lista)
    print(lukumaara)
    for alkio in lista:
        print(alkio)

main()
```

**Mitä seuraavat funktiot (kohdat e-f) tekevät?** Älä selitä funktion toimintaa käsky käskyltä, vaan selitä parella lauseella, mikä on funktion tarkoitus (esim. “funktio laskee ja palauttaa parametrina annetussa listassa olevien lukujen summan”). Funktiolle annettavien parametrien luonne on selitetty kunkin kohdan yhteydessä.

e) Funktiolle annetaan parametreina kokonaislukuja sisältävä lista ja kokonaisluku (4p)

```
def mystery_e(list1, raja):
    i = 0
    while i < len(list1):
        if list1[i] < raja:
            return list1[i]
        i += 1
    return 10000
```

f) Funktiolle annetaan ensimmäisenä parametrina positiivisia kokonaislukuja sisältävä lista, jossa on vähintään kaksi alkioita, ja toisena parametrina positiivinen kokonaisluku (5 p)

```
def mystery_f(list1, luku):
    i = 1
    while i < len(list1):
        if list1[i] > list1[i - 1] + luku:
            return False
        i += 1
    return True
```

g) **Korjaa seuraava funktio**, jolle annetaan ensimmäisenä parametrina positiivisia kokonaislukuja sisältävä lista. Funktion tarkoitus on tutkia, ovatko kaikki listan luvut suurempia kuin toisena parametrina annettu positiivinen kokonaisluku. Funktio ei kuitenkaan toimi tarkoitetulla tavalla. Selitä parilla lauseella, mikä virhe funktiossa on ja miten virhe pitäisi korjata. (5p)

```
def mystery_g(list1, figure):
    i = 0
    while i < len(list1):
        if list1[i] > figure:
            return False
        else:
            return True
    i = i + 1
```

## 2. Pieni ohjelma (10 p)

Eräällä kurssilla opiskelija saa arvosanan tentistä ja harjoitustyöstä. Jos vähintään toinen arvosanoista on nolla, opiskelijan kurssisuoritus on hylätty ja hän saa kurssista arvosanan nolla (0). Muussa tapauksessa opiskelijan kurssiarvosanaksi tulee yleensä tenttiarvosana. Jos kuitenkin harjoitustyön arvosana on viisi (5), niin hyväksyttyä tenttiarvosanaa (eli arvosanoja yhdestä neljään) kasvatetaan yhdellä. Kirjoita python-ohjelma, joka pyytää käyttäjältä opiskelijan tentin ja harjoitustyön arvosanat, ja tulostaa kurssiarvosanan.

## 3. Yksittäinen funktio (15 p)

Muuttoja järjestävä yritys käyttää ohjelmistoa, jonka avulla voidaan arvioidaan muuttokustannuksia. Kirjoita tätä ohjelmaa varten funktio `laatikoiden_maara`, joka laskee tarvittavien laatikoiden määrän. Se saa parametrina listan, jossa on pakattavien tavaroiden koko litroina. Muuttolaatikkoon voi laittaa tavaroita maksimissaan 69 litran verran. Tavarat pakataan listan antamassa järjestyksessä. Joskus laatikkoon saattaa siis jäädä jonkin verran tyhjää tilaa, jos seuraava tavara on liian iso laitettavaksi laatikkoon. Funktio palauttaa tarvittavien laatikoiden määrän kokonaislukuna. Jos jonkin tavaran koko on yli 69 litraa, funktio kuitenkin palauttaa -1 eikä siten kerro laatikoiden määrää. Voit olettaa, että funktiolle annetaan järkevät parametrin ja kaikki listassa olevat luvut ovat kokonaislukuja, ja että muu ohjelma huolehtii kaikesta muusta. Kirjoita vain pyydetty funktio, ei muuta ohjelman osia (esim. pääohjelmaa).

## 4. Tekstitiedosto ja virheiden käsittely (20 p)

Eräällä liikkeellä on tekstitiedostossa tieto sen myymistä tuotteista, niiden normaalihinnoista ja alennusprosentteista. Yhden tuotteen tiedot on tiedoston yhdellä rivillä niin, että tuotteen nimi, normaalihinta ja alennusprosentti on erotettu toisistaan puolipisteellä. Tiedoston rivit voisivat näyttää esimerkiksi seuraavilta:

```
Kattila Hackman;40.0;15.0
Hiusharja;8.0;5.0
Uusi Cola;2.0;20.0
```

Kirjoita Python-ohjelma, joka pyytää käyttäjältä tiedot sisältävän tiedoston nimen. Ohjelma lukee tämän tiedoston ja tulostaa niiden tuotteiden nimet ja alennetut hinnat, joiden alennusprosentti on vähintään 10. Jos tuotteen alennusprosentti on alle 10, tuotteesta ei tulosteta mitään tietoja. Yllä olevassa esimerkkitapauksessa ohjelman tulostus olisi

```
Kattila Hackman 34.0
Uusi Cola 1.6
```

Ohjelman on käsiteltävä seuraavat virhetilanteet:

- Annetun nimistä tiedostoa ei ole olemassa tai tiedoston lukeminen ei onnistu jostain muusta syystä
- Tiedoston jollain rivillä tuotteen hinnan tai alennusprosentin paikalla ei ole desimaaliluku.

Näissä tapauksissa ohjelma ilmoittaa käyttäjälle, millainen virhe on sattunut, ja lopettaa toimintansa. Ohjelman ei siis tarvitse jatkaa rivien lukemista virheellisen rivin jälkeen. Voit myös olettaa, että tiedoston jokaisella rivillä on täsmälleen kolme toisistaan puolipisteellä erotettua osaa. Ohjelman ei tarvitse osata käsitellä esimerkiksi sellaisia virhetilanteita, joissa rivi on tyhjä tai ei sisällä tuotteen nimen lisäksi muuta tekstiä.

## 5. Olio-ohjelmointi (25 p)

Toimit killan fuksivastaavaana. Yksi tehtävistäsi on pitää kirjaa fuksipisteistä, joita pitää kerätä tarpeeksi teekkarilakin ja teekkari-arvonimen saamiseksi Vappuna. Kirjoita Python-kielellä luokka `Opiskelija` alla olevan määritelmän mukaan ja laadi lisäksi **seuraavalla sivulla** oleva pääohjelma.

- `__nimi` opiskelijan nimi
- `__pisteet` opiskelijan keräämien oman killan antamien fuksipisteiden määrä (kokonaisluku)
- `__laulupiste` onko opiskelija suorittanut laulukokeen (totuusarvo `True` tai `False`)
- `__lisapisteet` opiskelijan keräämien lisäpisteiden määrä (kokonaisluku).

Määrittele luokkaan seuraavat metodit. Jos metodin kuvauksessa ei ole kerrottu mitään metodin palauttamasta arvosta, metodin ei tarvitse palauttaa mitään. (Muitakin metodeja olisi järkevä määrittellä, mutta ne eivät ole tässä tenttitehtävässä tarpeen.)

- `__init__(self, opiskelijan_nimi)` luo uuden `Opiskelija`-olion. Luotavan opiskelijan nimi annetaan metodin parametrina, ja uuden opiskelijan pisteet ovat 0 eikä hän ole suorittanut laulupistettä.
- `kerro_nimi(self)` palauttaa opiskelijan nimen.
- `kerro_fuksipisteet(self)` palauttaa opiskelijan kilta-fuksipisteiden määrän.
- `lisaa_pisteita(self, pisteet)` lisää opiskelijalle parametrina annetun määrän killan omia fuksipisteitä.
- `lisaa_laulupiste(self)` lisää laulupisteen.
- `lisaa_lisapisteita(self, pisteet)` lisää opiskelijalle parametrina annetun määrän muita fuksipisteitä.
- `poista_pisteita(self, tyyppi, pisteet)` on virhekirjauksien poistoa varten. Sille annetaan merkkijonona poistettavan pisteen tyyppi, ja kuinka monta pistettä poistetaan. Opiskelijan pisteet eivät kuitenkaan mene alle nollan, vaikka niitä yritettäisiin poistaa enemmän. Metodilla voi poistaa myös laulupisteen. Tällöin metodikutsussa pisteet-parametri on 1. Metodi palauttaa `True`, jos pisteiden poisto onnistui, ja `False`, jos pisteitä ei ollut poistettavaksi alkuunkaan.
- `kerro_fuksipisteet(self)` palauttaa fuksipisteiden määrän kokonaislukuna.
- `kerro_pisteet(self)` palauttaa fuksipisteiden ja lisäpisteiden yhteismäärän kokonaislukuna.
- `onko_laulupiste(self)` kertoo onko laulupiste suoritettu vai ei.
- `onko_tekkari(self)` palauttaa totuusarvon `True` tai `False` sen mukaan, onko opiskelija saanut tarvittavat pisteet tekkari-arvonimeä varten. Arvonimeä varten laulupisteen pitää olla suoritettu, ja fuksipisteitä pitää olla vähintään viisi (5) ja pisteitä (ilman laulupistettä) yhteensä vähintään 21.
- `__str__(self)` palauttaa merkkijonon, joka sisältää opiskelijan nimen, fuksipisteet, laulupisteet ja lisäpisteet pilkuilla eroteltuina.

Kirjoita lisäksi joko samaan tai toiseen moduuliin pääohjelma, joka luo kaksi `Opiskelija`-oliota, ja sen jälkeen lisää kummallekin valitsemasi määrän fuksipisteitä ja lisäpisteitä, sekä toiselle opiskelijalle myös laulupisteen. Sen jälkeen ohjelma poistaa toiselta opiskelijalta yhden lisäpisteen, ja tulostaa onnistuiko tämä vai ei. Tämän jälkeen ohjelman pitää kysyä, onko opiskelijat teekkari-arvonimen arvoisia. Lopuksi ohjelman on tulostettava molempien opiskelijoiden tiedot. Voit päättää opiskelijoiden nimet ja lisättävät pistemäärät itse, niitä ei tarvitse kysyä käyttäjältä.

## 6. Epäselväksi tai vaikeaksi koettua ohjelmonnissa (5 p)

Koitko kurssilla jonkun asian kohdalla oivalluksen, että näinhän tämä toimii! Mikä tuo asia oli? Mitä konseptia, käskyä tai käsitettä oivalluksesi koski? Miten ymmärsit asian aluksi väärin ja mikä auttoi asian ymmärtämisessä?

Jos mikään ei ollut vaikeaa esim. koska olet ohjelmoinut jo paljon, niin mieti, mikä asia voi olla hankalaa ymmärtää, kun ensimmäistä kertaa kirjoittaa ohjelmakoodia. Voit myös kertoa, mikä pythonissa yllätti, jos olet ohjelmoinut muilla ohjelmointikielillä.