

L08 Tehtävät

- Valmiiden aliohjelmakirjastojen käyttö, erityisesti `math`, `random` ja `datetime`
- Omien aliohjelmakirjastojen teko ja käyttö
- Tehtävän 5 rakenne vastaa harjoitustyön rakennetta ja CodeGraden hyväksymät palautukset tarkastetaan rakenteen osalta assistenttien toimesta, ks. tarkemmin alla
- **Kaikissa tehtävissä pitää olla mukana otsikkotiedot, ks. L07 tehtäväksianto. Otsikkotietojen puuttuminen voi johtaa tehtävän hylkäämiseen.**

Lue oppaan tämän viikon asioita käsittelevä luku 8. Lisäksi tehtävien suorittamiseen tarvitset aiempien lukujen tietoja.

L08T1: Python-kirjastojen käyttö, <code>math</code> ja <code>random</code>	1
L08T2: Oman kirjaston teko ja käyttö	3
L08T3: Python-kirjastojen käyttö, <code>datetime</code>	5
L08T4: Laajeneva valikkopohjainen ohjelma useassa tiedostossa	7
L08T5: Useasta tiedostosta muodostuva perusohjelma	8
CodeGraden hyväksymien L08T5 ohjelmien tarkastus	10

L08T1: Python-kirjastojen käyttö, `math` ja `random`

Harjoittele Pythonin kirjastojen käyttöä `math` ja `random` kirjastojen avulla. Tee normaali valikkopohjainen ohjelma valikko-aliohjelmalla ja pääohjelmalla, ja kutsu pääohjelman valintarakenteesta alla olevia funktioita sekä tulosta niiden arvot näytölle. Esimerkkiajon mukaisesti ohjelma kysyy tarvittavat syötteet aina operaation valitsemisen jälkeen. Tulosta desimaaliluvut eli absoluuttinen arvo, potenssi ja neliöjuuri pyöristettynä yhden desimaalin tarkkuudelle.

1. **Absoluuttinen arvo `fabs(x)`.** Funktio palauttaa parametrin absoluuttisen arvon.
2. **Kertoma `factorial(x)`.** Funktio laskee parametrin kertoman.
3. **Potenssiin korotus `pow(x, y)`.** Funktio palauttaa ensimmäisen parametrin korotettuna toiseen parametriin.
4. **Neliöjuuri `sqrt(x)`.** Funktio palauttaa parametrin neliöjuuren.
5. **Satunnaisluku `randint(x, y)` ja `seed()`.** `randint`-funktio palauttaa satunnaisen kokonaisluvun lukujen `x` ja `y` välistä ml. nämä rajat. Funktio voidaan alustaa `seed()`-funktioilla ja automaattitarkastuksen yhteydessä näin on pakko tehdä, joten kutsu `seed()` -funktioita pääohjelman ensimmäisenä käskynä parametrin arvolla 1.

Katso alla olevasta esimerkkiajosta ohjelman tarkempi toiminta. Kirjastofunktioiden toimintaan kannattaa tutustua IDLen helpin eli F1 avulla ja katsoa, mitä sieltä löytyy ja miten kirjastofunktiot toimivat. Myös tooltip'iä kannattaa hyödyntää tässä tehtävässä.

Ohjelman esimerkkiajo:

Mitä haluat tehdä?

- 1) Laske absoluuttinen arvo
- 2) Laske kertoma
- 3) Laske potenssi
- 4) Laske neliöjuuri
- 5) Arvo satunnaisluku
- 0) Lopeta

Anna valintasi: 1

Minkä luvun absoluuttinen arvo selvitetään: -12.345

Luvun absoluuttinen arvo on 12.3

Mitä haluat tehdä?

- 1) Laske absoluuttinen arvo
- 2) Laske kertoma
- 3) Laske potenssi
- 4) Laske neliöjuuri
- 5) Arvo satunnaisluku
- 0) Lopeta

Anna valintasi: 2

Minkä luvun kertoma lasketaan: 7

Luvun kertoma on 5040

Mitä haluat tehdä?

- 1) Laske absoluuttinen arvo
- 2) Laske kertoma
- 3) Laske potenssi
- 4) Laske neliöjuuri
- 5) Arvo satunnaisluku
- 0) Lopeta

Anna valintasi: 3

Mikä luku korotetaan potenssiin: 2

Mitä eksponenttia käytetään: 3

Luku korotettuna eksponenttiin on 8.0

Mitä haluat tehdä?

- 1) Laske absoluuttinen arvo
- 2) Laske kertoma
- 3) Laske potenssi
- 4) Laske neliöjuuri
- 5) Arvo satunnaisluku
- 0) Lopeta

Anna valintasi: 4

Mikä luvun neliöjuuri lasketaan: 16

Luvun neliöjuuri on 4.0

Mitä haluat tehdä?

- 1) Laske absoluuttinen arvo
- 2) Laske kertoma
- 3) Laske potenssi
- 4) Laske neliöjuuri
- 5) Arvo satunnaisluku
- 0) Lopeta

Anna valintasi: 5

Arvotaan satunnaisluku, anna rajat kokonaislukuina.

Anna minimi (otetaan mukaan): 1

Anna maksimi (otetaan mukaan): 9

Arvottiin satunnaisluku 3

```
Mitä haluat tehdä?
1) Laske absoluuttinen arvo
2) Laske kertoma
3) Laske potenssi
4) Laske neliöjuuri
5) Arvo satunnaisluku
0) Lopeta
Anna valintasi: 0
Lopetetaan

Kiitos ohjelman käytöstä.
```

L08T2: Oman kirjaston teko ja käyttö

Tee Python-ohjelma, joilla voi muuntaa yhdysvaltalaisia nesteen vetomittoja SI-yksiköiksi ja takaisin. Käytä muuntokertoimina Wikipedian tilavuuksia (https://fi.wikipedia.org/wiki/Yhdysvaltalaiset_mittayksik%C3%B6t). Toteuta muunnokset litrasta gallon'ksi, pint'ksi, cup'ksi ja fluid ounce'ksi sekä takaisin. Toteuta kukin muunnos omana aliohjelmana, joka saa parametrina muunnettavan arvon ja palauttaa paluuarvona tuloksen. Laita nämä muunnosfunktiot omaan tiedostoon aliohjelmakirjastoksi, lisää kirjastoon kiintoarvona kirjaston versionumero, nyt 1.0, ja tallenna tiedosto nimellä L08T2Kirjasto.py.

Tee toinen tiedosto, L08T2.py, jossa on pääohjelma eli alla olevan esimerkkiohjelman mukainen valikko sekä siihen liittyvä valintarakenne. Käyttäjä aloittaa valitsemalla haluamansa muunnoksen ja antaa muunnettavan arvon desimaalilukuna, jonka jälkeen pääohjelma kutsuu kirjastofunktiota suorittamaan tämän muunnoksen sekä tulostaa tuloksen näytölle pyöristettynä kahden desimaalin tarkkuuteen round()-funktiolla.

Kirjastojen kanssa on oleellista, että rajanpinta ja toiminnallisuus on ymmärretty oikein. Tässä ohjelmassa on muistettava, että jokainen funktio saa parametrina luvun ja palauttaa muunnoksen jälkeen luvun pyöristämättä sitä. Koska kyseessä on yksikkömuunnoskirjasto, on kirjastossa vain ko. muunnosfunktiot ja kirjaston version kertova numero (kiintoarvo). Pääohjelma ja valikko muodostavat tässä tehtävässä ohjelman käyttöliittymän ja siksi ne ovat toisessa tiedostossa.

Ohjelman esimerkkiajo:

```
Käytetään kirjaston versiota 1.0
Minkä muunnoksen haluat tehdä?
1) Anna muunnettava tilavuus
2) Muunna litra gallon'ksi
3) Muunna litra pint'ksi
4) Muunna litra cup'ksi
5) Muunna litra fluid ounce'ksi
6) Muunna gallon litroiksi
7) Muunna pint litroiksi
8) Muunna cup litroiksi
9) Muunna fluid ounce litroiksi
0) Lopeta
Anna valintasi: 1
Anna muunnettava tilavuus desimaalilukuna: 1

Minkä muunnoksen haluat tehdä?
1) Anna muunnettava tilavuus
2) Muunna litra gallon'ksi
```

3) Muunna litra pint'ksi
4) Muunna litra cup'ksi
5) Muunna litra fluid ounce'ksi
6) Muunna gallon litroiksi
7) Muunna pint litroiksi
8) Muunna cup litroiksi
9) Muunna fluid ounce litroiksi
0) Lopeta
Anna valintasi: 2
Litrat muutettuina gallon'ksi: 0.26

Minkä muunnoksen haluat tehdä?
1) Anna muunnettava tilavuus
2) Muunna litra gallon'ksi
3) Muunna litra pint'ksi
4) Muunna litra cup'ksi
5) Muunna litra fluid ounce'ksi
6) Muunna gallon litroiksi
7) Muunna pint litroiksi
8) Muunna cup litroiksi
9) Muunna fluid ounce litroiksi
0) Lopeta
Anna valintasi: 3
Litrat muutettuina pint'ksi: 2.13

Minkä muunnoksen haluat tehdä?
1) Anna muunnettava tilavuus
2) Muunna litra gallon'ksi
3) Muunna litra pint'ksi
4) Muunna litra cup'ksi
5) Muunna litra fluid ounce'ksi
6) Muunna gallon litroiksi
7) Muunna pint litroiksi
8) Muunna cup litroiksi
9) Muunna fluid ounce litroiksi
0) Lopeta
Anna valintasi: 9
Fluid ounce't muutettuina litroiksi: 0.03

Minkä muunnoksen haluat tehdä?
1) Anna muunnettava tilavuus
2) Muunna litra gallon'ksi
3) Muunna litra pint'ksi
4) Muunna litra cup'ksi
5) Muunna litra fluid ounce'ksi
6) Muunna gallon litroiksi
7) Muunna pint litroiksi
8) Muunna cup litroiksi
9) Muunna fluid ounce litroiksi
0) Lopeta
Anna valintasi: 0
Lopetetaan

Kiitos ohjelman käytöstä.

L08T3: Python-kirjastojen käyttö, datetime

Harjoittele Pythonin `datetime` kirjaston käyttöä toteuttamalla seuraavat toiminnot omissa aliohjelmissaan ja tee näitä kutsuva valikkopohjainen ohjelma eli toteuta esimerkkiajon mukainen valikko ja sitä vastaava pääohjelma. Huomaa, että rakenne muistuttaa edellisten tehtävien ohjelmien rakennetta, joten niitä kannattaa hyödyntää mahdollisuuksien mukaan.

1. **`datetime`-olion jäsenmuuttujat.** Ohjelma pyytää käyttäjältä päivämäärän ja kellonajan merkkijonona formaatissa ”pp.kk.vvvv hh:mm” ja muuttaa merkkijonon `datetime`-olioksi. Tästä oliosta voi käyttää jäsenmuuttujia – nyt päivämäärä, kuukausi, vuosi, tunnit ja minuutit – ja tulostaa ne näytölle esimerkkiajon mukaisesti.
2. **Ajanjakson pituuden laskenta.** Ohjelma kysyy käyttäjältä syntymäajan muodossa pp.kk.vvvv ja laskee kuinka vanha tämä oli 1.1.2000 sekä tulostaa vastauksen päivinä. Käyttämällä `datetime`-olioita iän voi laskea muuttujien erotuksena.
3. **Viikontähtäimien nimien tulostaminen.** `strftime`-funktiolla voi tulostaa `datetime`-olion sisältämiä tietoja monipuolisesti. Käytä tätä funktiota hyväksi ja tulosta viikontähtäimien nimet näytölle. Aloita luomalla `datetime`-olio käyttäen minkä tahansa maanantain päivämäärää, esim. 31.10.2022. Käy sen jälkeen viikon 7 päivää läpi toistorakenteella ja siirtymällä seuraavaan päivään `timedelta`-jäsenfunktiolla. Tietokoneen kieliasetukset vaikuttavat tulostuksen kieleen, mutta CodeGradessa viikontähtäimet tulostuvat englanniksi `strftime`-funktiolla.
4. **Yhden vuoden kuukausien nimien tulostaminen lyhenteinä.** Tämä tehtävä onnistuu edellisen kohdan ideoilla, joten mieti tähän tehtävään sopivat alkupäivämäärä ja siirtymä. Tehtävä on tulostaa 12 kuukauden nimet oikein eli siirtymän pitää osua aina seuraavalle kuukaudelle, muttei välttämättä samalle päivälle.

Katso alla olevasta esimerkkiajosta ohjelman tarkempi toiminta. `datetime`-moduulin tärkeimmät toiminnallisuudet löytyvät ohjelmointioppaassa ja Pythonin dokumentaatiosta (F1). Toteuta kukin yllä olevista toiminnoista omana aliohjelmanaan, tyypillisesti alle 10 riviä per aliohjelma.

Ohjelman esimerkkiajo:

Tämä ohjelma käyttää `datetime`-kirjastoa tehtävien ratkaisemiseen.

Mitä haluat tehdä:

- 1) Tunnista aika-olion komponentit
- 2) Laske ikä päivinä
- 3) Tulosta viikontähtäimet
- 4) Tulosta kuukaudet
- 0) Lopeta

Anna valintasi: 1

Anna päivämäärä ja kello muodossa 'pp.kk.vvvv hh:mm': 27.10.2022 21:46

Annoit vuoden 2022

Annoit kuukauden 10

Annoit päivän 27

Annoit tunnin 21

Annoit minuutin 46

Mitä haluat tehdä:

- 1) Tunnista aika-olion komponentit
- 2) Laske ikä päivinä
- 3) Tulosta viikontähtäimet
- 4) Tulosta kuukaudet
- 0) Lopeta

Anna valintasi: 2

Anna syntymäpäivä muodossa pp.kk.vvvv: 1.1.1999
1.1.2000 henkilö oli 365 päivää vanha.

Mitä haluat tehdä:

- 1) Tunnista aika-olion komponentit
- 2) Laske ikä päivinä
- 3) Tulosta viikonpäivät
- 4) Tulosta kuukaudet
- 0) Lopeta

Anna valintasi: 3

Monday

Tuesday

Wednesday

Thursday

Friday

Saturday

Sunday

Mitä haluat tehdä:

- 1) Tunnista aika-olion komponentit
- 2) Laske ikä päivinä
- 3) Tulosta viikonpäivät
- 4) Tulosta kuukaudet
- 0) Lopeta

Anna valintasi: 4

Jan

Feb

Mar

Apr

May

Jun

Jul

Aug

Sep

Oct

Nov

Dec

Mitä haluat tehdä:

- 1) Tunnista aika-olion komponentit
- 2) Laske ikä päivinä
- 3) Tulosta viikonpäivät
- 4) Tulosta kuukaudet
- 0) Lopeta

Anna valintasi: 0

Lopetetaan.

Kiitos ohjelman käytöstä.

L08T4: Laajeneva valikkopohjainen ohjelma useassa tiedostossa

Tällä viikolla valikkopohjainen ohjelma jaetaan kahteen tiedostoon. Laita pääohjelma ja siihen kiinteästi liittyvä valikko yhteen tiedostoon ja kaikki muut aliohjelmat toiseen tiedostoon eli kirjastoon. Yhtenäisyyden nimissä muokataan myös tiedostonimien kysymistä, jotta ohjelman rakenne vastaa harjoitustyön ja tehtävän L08T5 rakennetta. Tämän tehtävän lähtökohtana kannattaa käyttää itse tekemääsi ohjelmaa L07T4.

Toimivan ohjelman jako useisiin tiedostoihin ei edellytä juurikaan uutta koodia, vaikka luonnollisesti jotain on muutettava ohjelman rakenteen muuttuessa. Jaa toimiva ohjelma ensin yllä olevien ohjeiden mukaan kahteen eri tiedostoon ja nimeä ne L08T4.py sekä L08T4Kirjasto.py. Tee sen jälkeen ohjelmakoodiin tarvittavat muutokset oppimateriaalien mukaisesti. Lopuksi siirrä tehtävän L07T5 valikon valintana 1 olleet tiedostonimien kysymiset tiedoston lukemisen ja tiedoston kirjoittamisen yhteyteen eli niitä edeltäville riveille sekä järjestä valikko ja sitä vastaava valintarakenne uutta rakennetta vastaavaksi esimerkkiajon mukaisesti.

Useaan tiedostoon jaettu ohjelma toimii samalla tavalla kuin kaikki koodi yhdessä tiedostossa oleva ohjelma, joten tiedostot ovat samoja kuin aiemmin ja esimerkkiajo poikkeaa vain valikon rakenteen muutoksen osalta. Testaa ohjelma samoilla tiedostoilla kuin L07T4 ja muuta tallennettavan tiedoston nimi tehtävän mukaisesti esim. L08T4T1.txt. Moodlessa on testausta varten tiedostot L07T4D1.txt ja L07T4D2.txt.

Luettavan tiedoston muoto, L07T4D1.txt:

```
10000
12345
```

Kirjoitettavan tiedoston muoto, L08T4T1.txt:

Analyysin tulokset ovat seuraavat:

Datan pienin arvo on 10000.

Datan suurin arvo on 12345.

Datan summa on 22345.

Datan keskiarvo on 11172.0.

Ohjelman esimerkkiajo:

Tämä on valikkopohjainen ohjelma, jossa voit valita haluamasi toiminnon.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 0) Lopeta

Anna valintasi: 1

Luettavan tiedoston nimi on ''.

Anna uusi nimi, enter säilyttää nykyisen: L07T4D1.txt

Tiedosto 'L07T4D1.txt' luettu.

Valitse haluamasi toiminto:

- 1) Lue tiedosto
- 2) Analysoi
- 3) Kirjoita tiedosto
- 0) Lopeta

Anna valintasi: 2

```
Analyysi suoritettu.

Valitse haluamasi toiminto:
1) Lue tiedosto
2) Analysoi
3) Kirjoita tiedosto
0) Lopeta
Anna valintasi: 3
Kirjoitettavan tiedoston nimi on ''.
Anna uusi nimi, enter säilyttää nykyisen: L08T4T1.txt
Tulokset kirjoitettu tiedostoon.

Valitse haluamasi toiminto:
1) Lue tiedosto
2) Analysoi
3) Kirjoita tiedosto
0) Lopeta
Anna valintasi: 0
Lopetetaan

Kiitos ohjelman käytöstä.
```

L08T5: Useasta tiedostosta muodostuva perusohjelma

Tee valikkopohjainen ohjelma, joka pystyy lukemaan tiedoston, analysoimaan luetut tiedot, tallentamaan tulokset tiedostoon sekä lopettamaan ohjelman suorituksen. Tässä tehtävässä harjoitellaan itse tehdyn kirjaston käyttämistä, joten laita pääohjelma valikon kanssa yhteen tiedostoon, L08T5.py, ja kaikki muut aliohjelmat toiseen tiedostoon eli kirjastoon, L08T5Kirjasto.py.

Pääohjelma on minimaalinen ja keskittyy ikisilmukan pyörittämiseen sekä kutsuu kirjaston aliohjelmia käyttäjän valintojen mukaisesti. Koska tieto kulkee ohjelmien välillä parametreina ja paluuarvoina, määrittele pääohjelmassa kaksi listaa, yksi tiedostosta luettuja rivejä varten ja toinen analysoitujen tietojen tallettamista varten. Kysy tiedostojen nimet samalla aliohjelmalla, jota kutsutaan pääohjelmasta ennen tiedoston lukemista ja kirjoittamista ja välitä aliohjelmalle tilanteeseen sopiva kehote. Tiedostonimen kysymisen yhteydessä tulostetaan ensin nykyinen tiedostonimi ja käyttäjä voi jatkaa sen käyttämistä antamalla tyhjän tiedostonimen eli painamalla enter:iä. Alusta molemmat tiedostonimet tyhjiksi, jotta käyttäjä antaa niille ensimmäisellä kerralla sopivat nimet.

Kirjastossa tulee määritellä luokka tietoja varten ja aliohjelmat toimintojen tekemiseen. Tässä tehtävässä tärkeintä on ohjelman järkevä rakenne, joten pidä aliohjelmat lyhyinä ja niiden rooli on osoittaa ohjelman järkevä eteneminen, sillä yksittäisiä aliohjelmia voidaan myöhemmin laajentaa ja kehittää.

Alla on ohjelman lukema tiedosto, L08T5D1.txt, josta näkyy siinä olevat tiedot: tuotetunniste, tuotteiden lukumäärä ja yhden tuotteen hinta. Tee ohjelmaan luokka, jossa on jäsenmuuttuja jokaista tietoa varten, alusta ne None:lla ja laita aina yhden rivin tiedot yhden olion jäsenmuuttujiin ja olio listaan eli tee oliolista.

Tietojen analyysi tarkoittaa varaston eri tuotteiden arvojen laskemista tuloslistaan niin, että jokaisen tuotteen varaston arvo laitetaan tuloslistan alkioksi. Arvo saadaan kertomalla tuotteiden määrä niiden arvolla. Analyysi-aliohjelman päätteeksi tulostetaan koko varaston arvo.

Tietojen tallennusvaiheessa kaikki tuloslistassa olevat alkiot kirjoitetaan yksi alkio aina yhdelle riville ja rivinvaihto viimeisenä merkinä. Tulosta kaikki desimaaliluvut kahden desimaalin tarkkuudella. Katso valikon ja ilmoitusten esitysasut ohjelman esimerkkiajosta.

Tämän ohjelman rakenne muistuttaa harjoitustyön rakennetta, joten tämä tehtävä kannattaa tehdä itse. Harjoitustyötä tehdessä kannattaa ottaa tämä sama rakenne lähtökohdaksi.

Ohjelman lukema tiedosto L08T5D1.txt

```
A01-128;3;12.30
A01-987;10;1.15
B12-1;1;123.45
B54-182;111;13.50
C21-321;32;321.10
```

Ohjelman kirjoittaman tiedosto L08T5T1.txt

```
36.90
11.50
123.45
1498.50
10275.20
```

Ohjelman esimerkkiajo:

Mitä haluat tehdä:

- 1) Lue tiedosto
- 2) Analysoi tiedot
- 3) Tallenna tulokset
- 0) Lopeta

Valintasi: 1

Anna luettavan tiedoston nimi, nykyinen on ''.

Anna uusi nimi, enter säilyttää nykyisen: L08T5D1.txt

Tiedosto 'L08T5D1.txt' luettu, 5 riviä.

Mitä haluat tehdä:

- 1) Lue tiedosto
- 2) Analysoi tiedot
- 3) Tallenna tulokset
- 0) Lopeta

Valintasi: 2

Tiedot analysoitu, varaston arvo on 11945.55 EUR.

Mitä haluat tehdä:

- 1) Lue tiedosto
- 2) Analysoi tiedot
- 3) Tallenna tulokset
- 0) Lopeta

Valintasi: 3

Anna kirjoitettavan tiedoston nimi, nykyinen on ''.

Anna uusi nimi, enter säilyttää nykyisen: L08T5T1.txt

Tulokset tallennettu tiedostoon 'L08T5T1.txt'.

Mitä haluat tehdä:

- 1) Lue tiedosto

```
2) Analysoi tiedot
3) Tallenna tulokset
0) Lopeta
Valintasi: 0
Lopetetaan.
```

Kiitos ohjelman käytöstä.

CodeGraden hyväksymien L08T5 ohjelmien tarkastus

Kaikki CodeGraden hyväksymät palautukset tehtävään L08T5 tullaan tarkastamaan assistenttien toimesta ja niistä annetaan palautetta luennoilla nimettyjen asioiden osalta. Mikäli kaikki kohdat on toteutettu kurssilla annettujen ohjeiden mukaisesti, saa tarkastuksesta yhden lisäpisteen viikkotehtäväpisteisiin. Tarkastuksen palaute on jokaisen nimetyn asian osalta ”OK/Kesken/EiOk”. Nämä samat asiat sisältyvät harjoitustyön arviointiin eli tekemällä tämän ohjelman ja katsomalla tarkastuksen palautteet pitäisi pystyä välttämään harjoitustyön korjauskierrokset.

Mikäli tarkastuksen palauteen perusteella ohjelmassa ei ole kaikki kunnossa, kannattaa katsoa mitä ko. asiasta on puhuttu luennolla. Samoin neuvontatilaisuuksissa ja keskustelupalstoilla voi kysyä neuvoja assistenteilta. ASPAa kannattaa käyttää ohjelman tarkastamiseen jo ennen sen palautusta, sillä se tarkistaa pitkälti samat asiat kuin assistenttikin.

Huomaa, että tekemällä tehtävän L08T5

- voit saada yhden ylimääräisen pisteen viikkotehtäviin
- saat hyvän lähtökohdan harjoitustyön oikean rakenteen tekemiseen. Tämän tehtävän ja harjoitustyön perusversion rakenne ovat hyvin samanlaisia ja assistentin palautteella saat varmuutta siihen, että olet toteuttanut ohjelman rakenteet oikein.