

Python-ohjelmointi

Harjoitus 16

TAVOITTEET

- Kerrataan *datetime*- ja *time*-kirjastojen käyttö.
- Kerrataan konnagrafiikan perustoiminnot sekä ehto- ja vertailurakenteet.
- Opitaan käyttämään näppäimistöä konnan ohjaamiseen sekä muodostamaan erillinen suoritusruutu piirtämiselle.
- Vahvistetaan aliohjelmien suunnittelua, tekemistä ja käyttämistä osana omaa ohjelmaa.

Matematiikan sisällöt

Tehtävissä harjoitellaan ja syvennetään polynomilaskennan taitoja sekä yhtälöparien ratkaisemista.

Ohjelmointiin käytettävä alusta löytyy osoitteesta <http://sade-oppiminen.herokuapp.com/>

KERTAUSTA HARJOITUKSESTA 15

TEHTÄVÄ 1

Tavoitteena on palauttaa mieleen *datetime*-kirjaston komentoja.

Run

Tee ohjelma, jolla voit laskea kahden päivämäärän erotuksen. Ohjelma tulostaa erotuksen ensin päivinä ja sen jälkeen vielä muodossa ____ v, ____ kk, ____ pp aliohjelmaa apuna käyttäen (katso harjoituskokoelma 15, tehtävä7).

Päivämäärät pyydetään käyttäjältä. Päivämäärät kannattaa pyytää siten, että kysytään erikseen molemmista päivämääristä vuosiluku, sitten kuukausi ja sen jälkeen päiväys.

TEHTÄVÄ 2

Tavoitteena on palauttaa mieleen *time*- kirjaston ajanmittausmahdollisuus sekä ehtorakenteen ja vertailun käyttäminen.

Run

Tee ohjelma, jonka avulla käyttäjä voi tutkia ajankulumisen arviointikykyään.

Ohjelmassa

- Käyttäjä saa valita, haluaako hän arvioida 1 s, 2 s, 5 s vai 10 s kestoaa.
- Valinnan jälkeen tallennetaan aloitusajanhetki.
- Kun käyttäjän mielestä aikaa on kulunut riittävästi, hän kirjoittaa vastausruutuun x .
- Otetaan lopetusajanhetki ja verrataan käyttäjän arviota todelliseen kestoan. Arvioinnissa sallitaan 0,05 sekunnin heitto eli esimerkiksi yhden sekunnin keston oikeellisuutta tarkasteltaessa:

- Koska vertailuehtoon ei voi suoraan määrittää, että arvioitu aika tulee olla tietyllä välillä, kirjoitetaan vertailuehto muodossa: *t on suurempi tai yhtä suuri kuin 0,95 ja t on pienempi tai yhtä suuri kuin 1,05*.
- Käyttäjä saa arviointikyvystään palautteen ohjelman päätteeksi.

TEHTÄVÄ 3

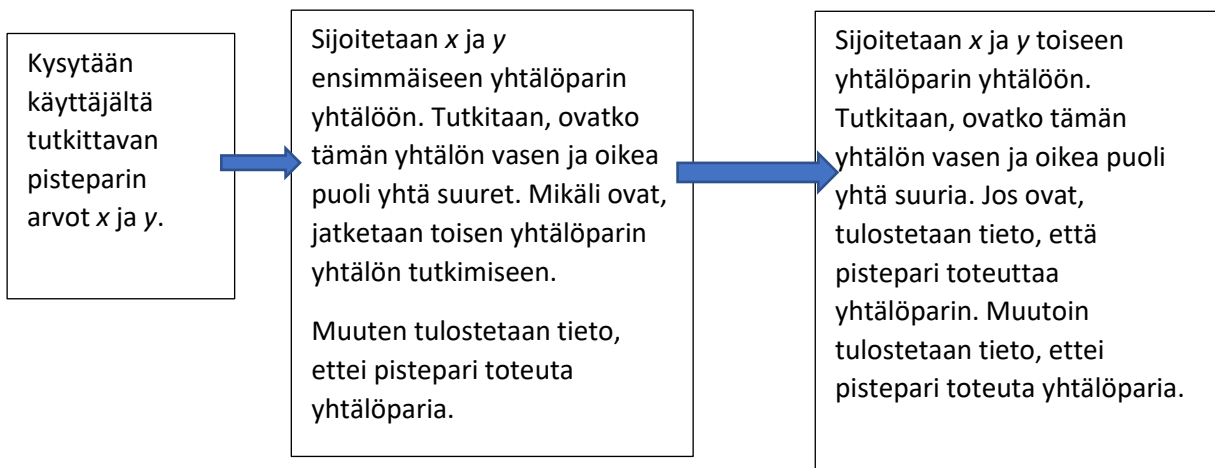
Tavoitteena on palauttaa mieleen ehtorakenne ja vertailuehtojen käyttäminen sekä muuttujien käyttäminen lausekkeissa.

Run

Tee ohjelma, jonka avulla voidaan tutkia, toteutuuko yhtälöpari annetulla pisteparilla (x, y) .

Sijoita annetut arvot yhtälöparin yhtälöihin ja tee vertailut, joiden perusteella tulostuu tieto, toteuttaako annettu pistepari yhtälöparin vai ei.

Tee ohjelma seuraavan suunnitelman perusteella:



KONNAN LIKUTTAMINEN PIIRTOALUEELLA JA ERILLINEN SUORITUSALUE PIIRTÄMISELLE

Tähän asti konnaa on liikutettu koodin avulla, mutta konnaa voidaan liikuttaa myös näppäimistöä käyttäen.

Ensin täytyy määritellä jokaiselle käytettävälle näppäimelle oma aliohjelma. Aliohjelmassa kerrotaan, mitä konna tekee kyseisellä näppäimen painalluksella. Ohjelman suorittamisen jälkeen klikataan hiirellä tehtyä suoritusaluetta ja sen jälkeen konnaa voidaan liikuttaa tuolla alueella.

LIKKUMISEEN LIITTYVÄT KOMENNOT

<code>alue.onkey(funktio, "näppäimen_nimi")</code>	Tämä komento yhdistää näppäimen ja aliohjelman toisiinsa. Näppäimen nimet kirjoitetaan englanniksi, esimerkiksi "nuoli ylös" on "Up".
<code>alue.listen()</code>	Ohjelma tunnistaa näppäinkomennot.

Konnaohjelmoinnissa on mahdollista myös määritellä erillinen alue, joka toimii piirto- ja toimintoalueena. Erillisen alueen keskipiste vastaa origoa ja koordinaattipisteet täytyy suhteuttaa määritellyn alueen kokoon.

ERILLINEN SUORITUSRUUTU

<code>alue.setup(200, 200)</code>	<code>Screen()</code> on määritelty alueeksi ja alueelle määritellään koko pikseleinä.
<code>alue.bgcolor('väri')</code>	Määritetään erillisen alueen väri

Esimerkki 1

Tehdään ohjelma, jossa konnaa liikutetaan piirtoalueella. Samalla konnan kulkema reitti piirtyy ja välillä konnan sijainti leimataan. Konna lähtee liikkeelle alueen vasemmasta alanurkasta. Konnaa liikutetaan oikealle ja ylös näppäimistön vastaavien nuolinäppäimien avulla. Konna liikkuma reitti piirtyy suoritusalueelle, lisäksi konna leimaa välillä sijaintiaan. Leimaamiseen käytetään välilyöntipainiketta.

Ratkaisu

Valitaan aloituspisteeksi (-300, -200).

Määritellään yhden näppäinpainalluksen etenemäksi 20 yksikköä.

```
from turtle import*
naytto = Screen()

konna = Turtle()
#Siirretään konna aloituspisteeseen ilman piirrosviivaa.
konna.shape('turtle')
konna.penup()
konna.goto(-300, -200)
#Lasketaan kynä, jotta reitti piirtyy.
konna.pendown()

#Tehdään aliohjelmat näppäinkomennoille.
def oikea():
    konna.seth(0)           #konna osoittaa oikeaan
    konna.forward(20)      #yhden näppäinpainalluksen etenemä

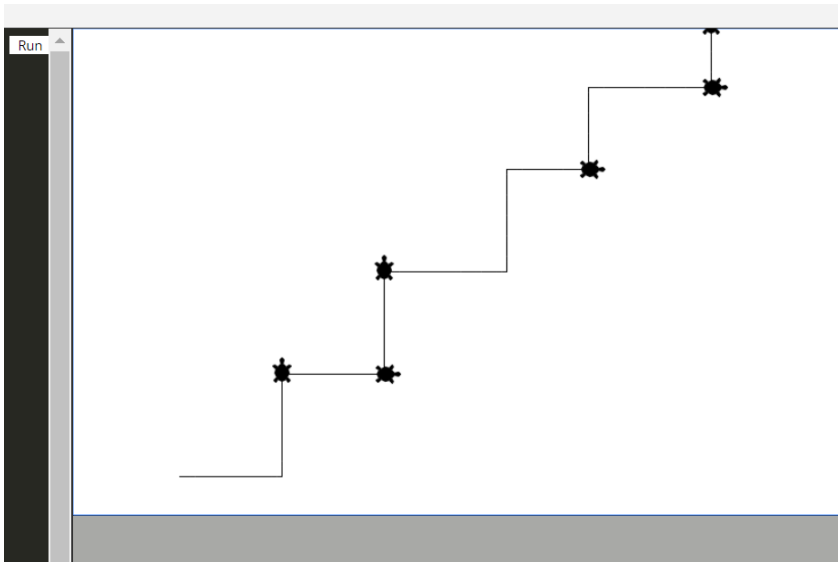
def ylos():
    konna.seth(90)        #konna osoittaa ylös
    konna.forward(20)

def leimaus():
    konna.stamp()

#Yhdistetään aliohjelmat ja näppäinpainallukset.
naytto.onkey(oikea, "Right")
naytto.onkey(ylos, "Up")
naytto.onkey(leimaus, "Space")

#Kuunnellaan näppäimistön painalluksia.
naytto.listen()
```

Ohjelman suorittamisen jälkeen klikataan piirtoaluetta ja käytetään määriteltyjä nuolinäppäimiä liikuttamiseen ja välilyöntipainikkeella tehdään sijainnin leimaus.

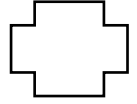


TEHTÄVÄ 4

Tavoitteena on harjoitella konnan liikuttamista.

Run

Tee ohjelma, jossa liikuttamalla konnaa nuolinäppäimillä saat piirrettyä viereisen kuvan kaltaisen kuvion.



Voit palauttaa mieleen myös konnaohjelmoinnin komennoista, miten saat vaihdettua muodostuvan viivan väriä.

Esimerkki 2

Tehdään ohjelma, joka määrittää erillisen piirtoalueen kooltaan 200 x 200. Alueelle määritetään taustaväri sekä kirjoitetaan vasempaan ylänurkkaan teksti ” Erillinen piirtoalue”.

Ratkaisu

```
from turtle import*  
  
alue = Screen()  
alue.setup(200, 200)  
alue.bgcolor('pink')  
  
konna = Turtle()  
konna.penup()  
konna.goto(-75, 75)  
konna.write('Erillinen piirtoalue')
```



Yllä näkyy koodi vasemmalla ja ohjelman suorituksen jälkeinen tulos näkyy oikealla. Eli erillinen suoritusalue tulee alkuperäisen piirtoalueen päälle.

TEHTÄVÄ 5

Run

Tee ohjelma, jossa määritetään erillinen piirtoalue kooltaan 300 x 400. Määritä piirtoalueelle haluamasi taustaväri.

Piirrä konnaa liikuttamalla alueelle 50 x 100 kokoinen suorakulmio.

TEHTÄVÄ 6

Tavoitteena on vahvistaa näppäimistön käyttöä konnan toiminnallisuuden ohjaamisessa ja harjoitella sijainnin arviointia koordinaatistossa. Palautetaan myös mieleen konnaohjelmoinnin peruskomentoja.

Run

Tee ohjelma, jossa piirretään kaksi suoraa erilliselle piirtoalueelle kooltaan 400 x 300. Kun suorat on piirretty, liikutetaan konna suorien leikkauspisteeseen. Konna liikkuu yhdellä näppäimen painalluksella 1 - 10 yksikköä. Mitä pienempi on liikkumisaskel, sitä tarkemmin saat konnan sijoitettua. Konnan liikkumisnopeutta voi myös tarvittaessa nostaa. Kun konna on leikkauspisteessä, leikkauspiste voidaan ensin leimata ja sen jälkeen tulostetaan leikkauspisteen koordinaatit.

Tee liikuttelualiohjelmien lisäksi seuraavat näppäimistöä käynnistyvät toiminnot:

- s-kirjaimella saat piirrettyä ensimmäisen suoran
 - o Konna komennetaan siirtymään suoran aloituspisteeseen.
 - o Suoran aloituspisteen
 - x -koordinaatti saa arvon -120
 - y -koordinaatti saa arvon y :n suhteen ratkaistusta muodosta, ellei se ole valmiiksi siinä muodossa. Jos kyseessä on viereisen laatikon esimerkin suora, niin y -koordinaatti kirjoitetaan muodossa $2^*x + 4$. Mikäli et käytä x -koordinaatin määrittelyyn muuttujaa, korvaa y -koordinaatin lausekkeessa x arvolla -120.
 - o Konna komennetaan siirtymään suoran lopetuspisteeseen.
 - o Suoran lopetuspisteen
 - x -koordinaatti saa arvon 120
 - y -koordinaatin kohdalle kirjoitetaan suoran lauseke y :n suhteen ratkaistussa muodossa.
 - o Kun lopetat suoran piirtämisen, muista nostaa kynä ylös.
- a-kirjaimella saat piirrettyä toisen suoran
 - o Suoran aloitus ja lopetuspisteet tehdään kuten nousevallakin suoralla.
- välilyönnillä saat leimattua sijainnin
- p-kirjaimella saat tulostettua sijainnin koordinaattipisteet
 - o sijainti saadaan tulostettua komennolla *print konna.pos()*

Alkuperäinen

$$-4x + 2y = 8$$

 y :n suhteen ratkaistu

$$y = 2x + 4$$

Piirrä näppäinkomentoja käyttäen nouseva suora ja laskeva suora. Siirrä konna niiden leikkauspisteeseen, leimaa leikkauspiste ja tulosta leikkauspisteen koordinaatit.

RISTINOLLA-PELI

Seuraavien tehtävien avulla saadaan toteutettua ristinollan ideaa jäljittelevä peli. Pelialueena on 3 x 3 - ruudukko, jossa kahdella eri konnahahmolla liikkuvat pelaajat liikkuvat haluamaansa ruutuun ja leimaavat sijaintinsa. Tavoitteena on saada kolme samaa hahmoa vaakariville, pystyriville tai lävistäjille.

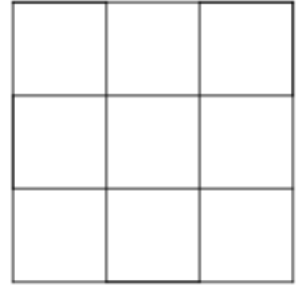
TEHTÄVÄ 7

Tavoitteena on palauttaa mieleen piirtäminen konnagrafiikalla.

Run

Tee ohjelma, joka piirtää 3x3-ruudukon erilliselle piirtoalueelle.

Suunnittele ohjelmasta sellainen, että sitä voidaan hyödyntää myöhemmin aliohjelmana.



TEHTÄVÄ 8

Tavoitteena on vahvistaa konnien liikuttamista näppäimistöä hyödyntäen.

Run

Käytä edellisen harjoituksen koodia aliohjelman pelipohjan luomiseen. Lisää kaksi konnaa, joilla molemmilla on eri ulkoasu. Konnat eivät piirrä liikuessa viivaa. Lisää konnille liikkumiskomennot. Huomaa, että molemmat konnat tarvitsevat omat liikuttamisnäppäimensä. Konna merkitsee ruudun leimaamalla. Molemmille konnille tarvitsee valita omat leimausnäppäimensä.