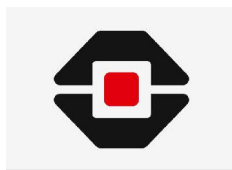




## VIIVAN SEURAAMINEN

Jos robotin on tarkoitus seurata enimmäkseen suoria viivoja, lineaarista (proportionaali) säätöä pidetään yleisesti riittävänä.

Heijastuvan valon mittausohjelma on hyödyksi viivanseurannan virittämisessä. Muista myös sivuvalosuojaus.



LEGO education

EV3-G



LEGO education

EV3 Classroom



LEGO education

Spike Scratch



LEGO education

Spike Python

# TAUSTAA

## VIIVAN SEURAAMINEN

Suoriutuakseen tehtävistään / ansaitakseen pisteitä robotin pitää aina ensin löytää perille, joka kerta!

Värianturin toimintaan yleisesti liittyviä virhelähteitä tyypeittäin:

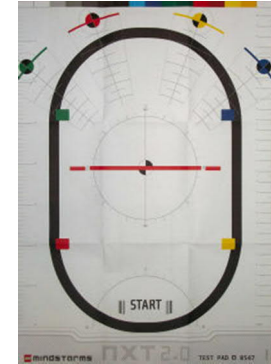
- Satunnaisia, kuten kennon ja mitattavan pinnan pienet asentojen poikkeamat. Kenno ja mitattava pinta eivät ole aina optimaalisesti keskenään kohtisuorassa asennossa.
- Systemaattisia, kuten kennon asentaminen väärälle etäisyydelle tarkasteltavasta pinnasta, tai alentuneesta käyttöjännitteestä johtuva apuvalon heikkeneminen (akku on lähes tyhjä).
- Ulkopuolisia, kuten sivusta tuleva kirkas valo, tai epäselvä ratamerkintä.
- Sisäisiä, kuten mittaustiedon tarkasteluväli. Ohjain kykenisi lukemaan valokennon mittaustiedon jopa 1000 kertaa sekunnissa, mutta ohjelman suoritusnopeus on vain murto-osa tästä.

Esimerkiksi kulkusuunnan kanssa risteävän viivan havaitseminen riippuu silmukan pituudesta, eli montako komentoa se pitää sisällään ja miten nopeasti robotti kulkee. Liian suurella nopeudella isokin kohde saattaa jäädä robotilta välillä huomaamatta.



# TEHTÄVÄ

## VIIVAN SEURAAMINEN

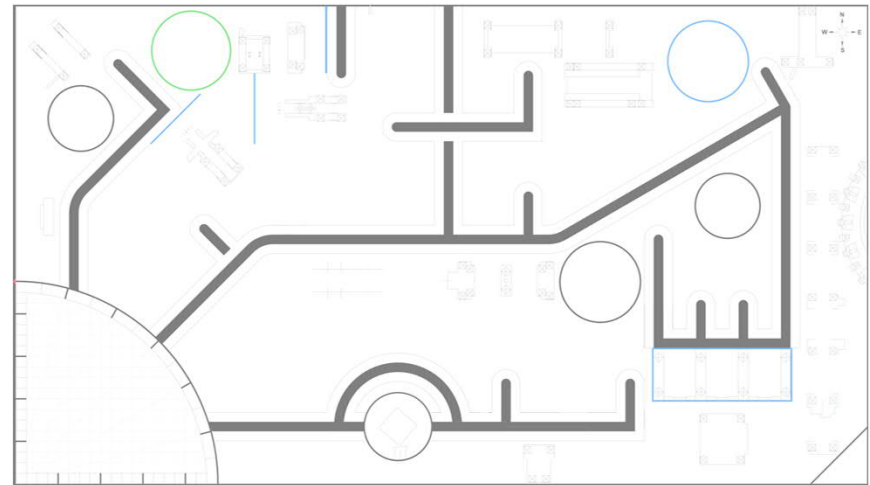


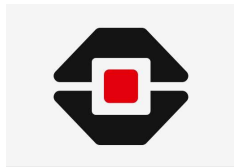
Kokeillaan lineaarisen viivanseurannan ohjelmoimista, virittämistä ja käyttämistä.

Tehtävässä voidaan hyödyntää valmiita alustoja tai laatia oma testirata mustalla teipillä vaalealle alustalle (tai valkoisella teipillä tummalle alustalle).

Tämän jakson lopussa on kaksi viivanseurannan harjoitusta.

Yllä ohjelmien testauksessa käytetty alusta (Paper Test Mat for NXT 2.0) Alla esimerkki viivanseurannan harjoitteluun soveltuvasta alustasta.



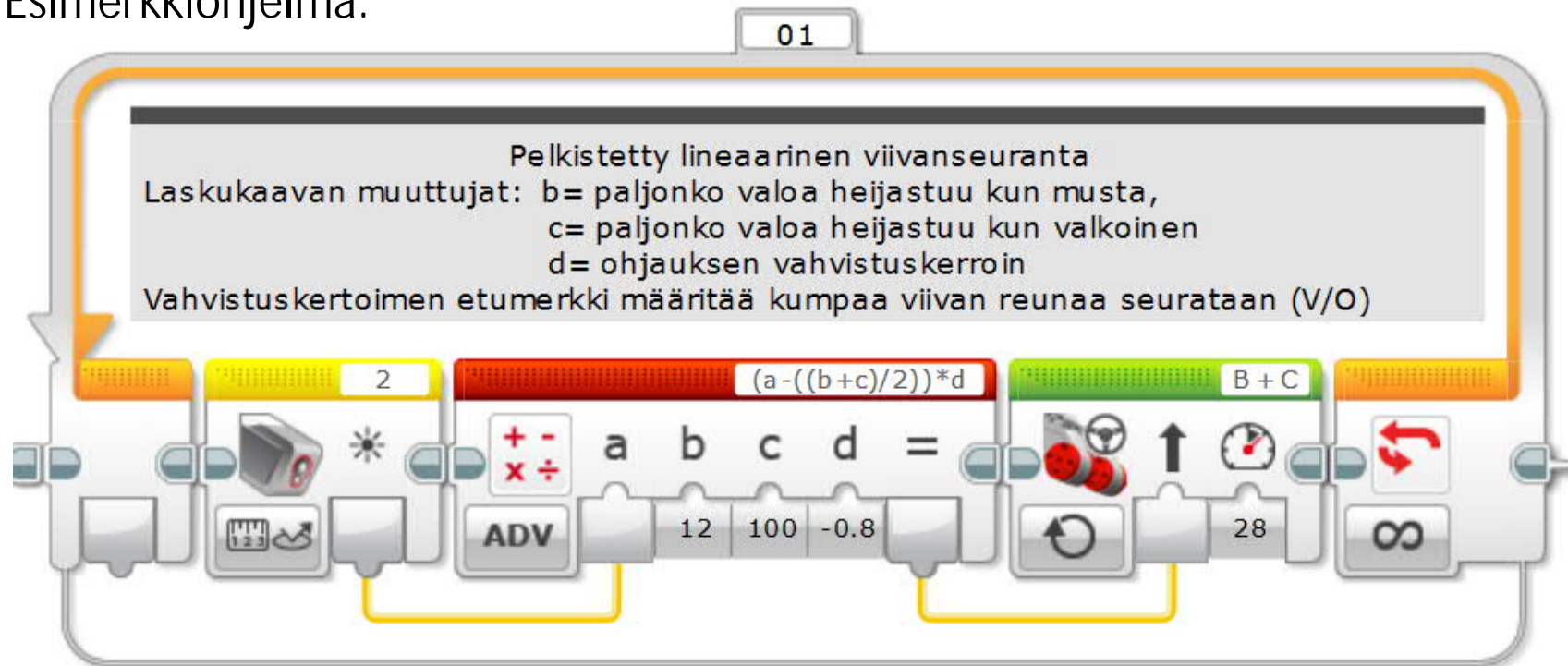


LEGO education

# VIIVAN SEURAAMINEN

EV3-G

Esimerkkiohjelma:



- Viritys: Mitataan valon heijastuminen mustalla ja valkoisella, muuttujat b ja c.  
Kokeillaan mitkä ovat sopivat arvot
- Vahvistuskertoimelle, muuttuja d.
  - Ajonopeudelle.





# VIIVAN SEURAAMINEN

## EV3 Classroom

### Esimerkkiohjelma:

```
when program starts
  set movement speed to 28 %
  set Vahvistuskerroin to -0.6
  set Musta to 12
  set Valkoinen to 100
  forever loop
    start moving (2 reflected light intensity - Musta + Valkoinen / 2 * Vahvistuskerroin)
```

### Muuttujien viritys:

- Mitataan valon heijastuminen mustalla ja valkoisella
- Kokeillaan mitkä ovat sopivat arvot
  - Vahvistuskertoimelle
  - Ajonopeudelle.

Kuten esimerkkiohjelmistakin käy ilmi, EV3 Classroom:in ja EV3-G:n ohjaus-toimilohkojen toiminnassa on pientä eroa (vahvistuskerroin).

Huomaa että jos ajomoottoreina on jokin muu kuin oletus-yhdistelmä B ja C, ajomoottorit pitää määritellä ohjelmassa.

Ohjaus = (heijastunut valo -((musta + valkoinen)/2)) \* vahvistuskerroin

Ole tarkka kaavaa ohjelmoidessasi, laskujärjestys riippuu siitä.

Lisää moottorilohkoon ensin kertolasku ja vahvistuskerroin, sitten vähennyslasku ja värianturi. Seuraavaksi tulee kahdella jakaminen ja kaikkein sisimmäiseksi mustan ja valkoisen arvojen yhteenlaskeminen.

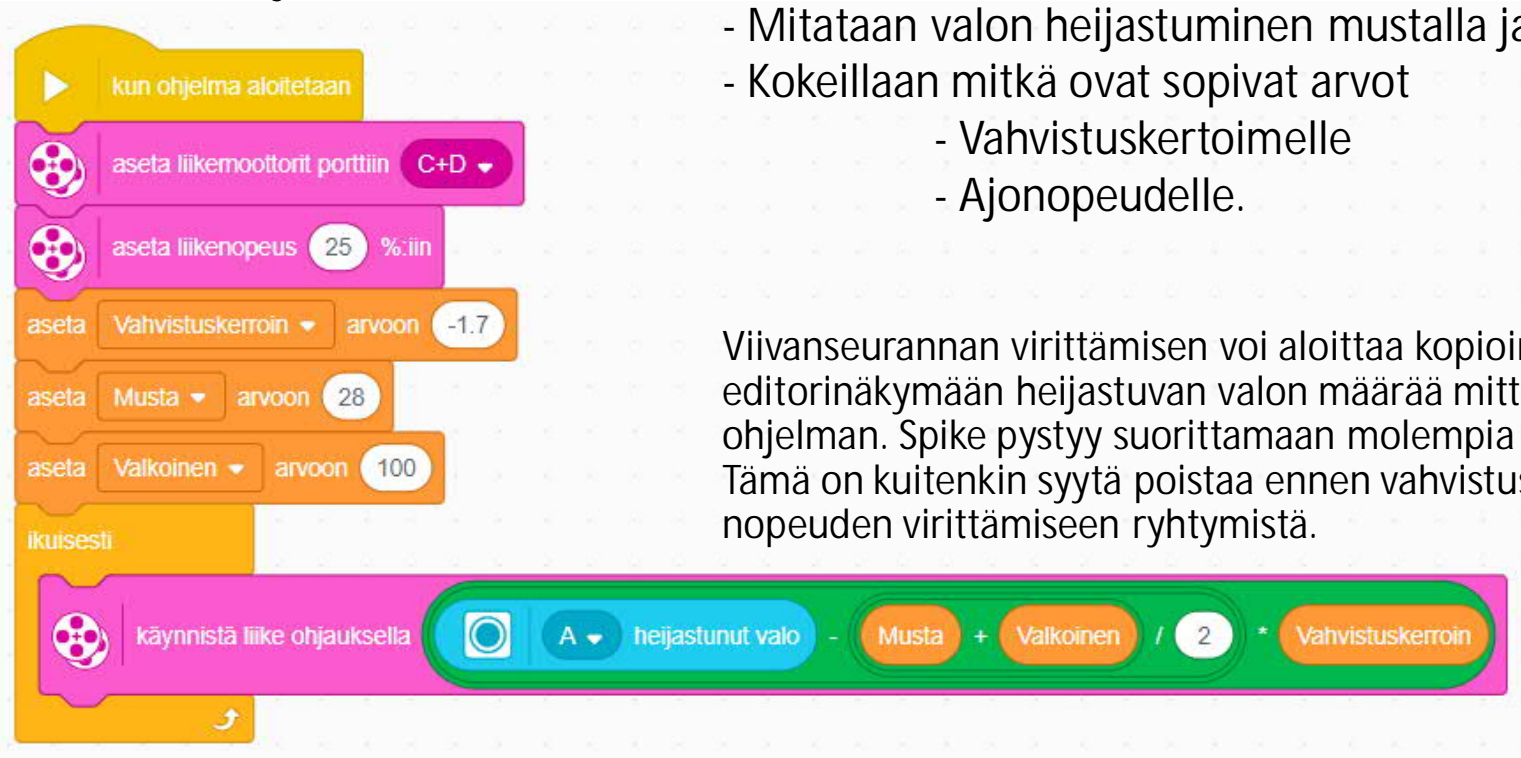




# VIIVAN SEURAAMINEN

## Spike Scratch

### Esimerkkiohjelma:



### Muuttujien viritys:

- Mitataan valon heijastuminen mustalla ja valkoisella
- Kokeillaan mitkä ovat sopivat arvot
  - Vahvistuskertoimelle
  - Ajonopeudelle.

Viivanseurannan virittämisen voi aloittaa kopioimalla samaan editorinäkymään heijastuvan valon määrää mittaavan ohjelman. Spike pystyy suorittamaan molempia yhtä aikaa. Tämä on kuitenkin syytä poistaa ennen vahvistuskertoimen ja nopeuden virittämiseen ryhtymistä.

Ohjaus = (heijastunut valo - ((musta + valkoinen)/2)) \* vahvistuskerroin

Ole tarkka kaavaa ohjelmoidessasi, laskujärjestys riippuu siitä.

Lisää moottorilohkoon ensin kertolasku ja vahvistuskerroin, sitten vähennyslasku ja värianturi. Seuraavaksi tulee kahdella jakaminen ja kaikkein sisimmäiseksi mustan ja valkoisen arvojen yhteenlaskeminen.





LEGO education

# VIIVAN SEURAAMINEN

## Spike Python



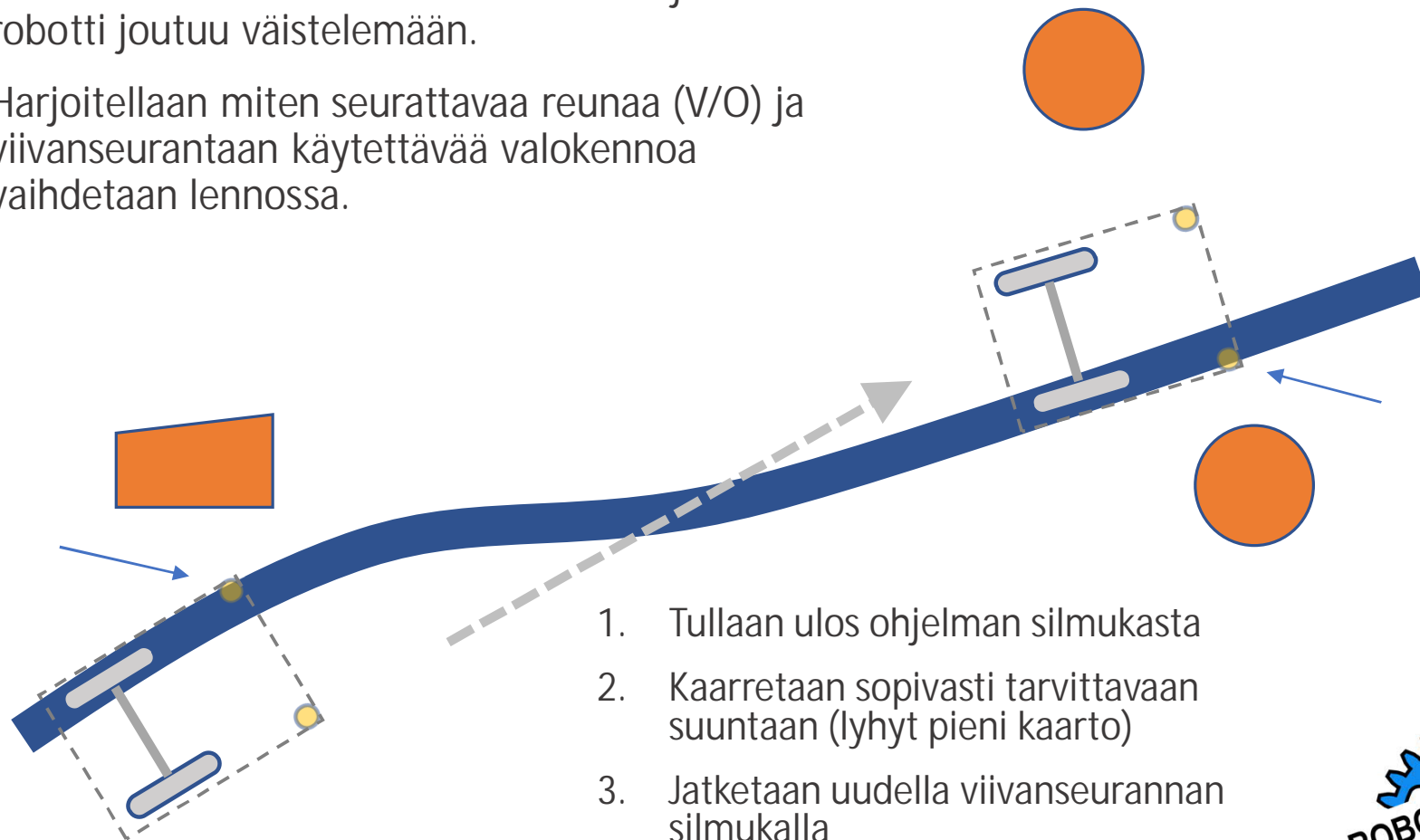
```
1 # Ladataan kirjastot
2 from spike import PrimeHub, ColorSensor, MotorPair
3 from math import *
4 # Tehdään alustukset
5 hub = PrimeHub()
6 moottorit = MotorPair('C', 'D')
7 värianturi = ColorSensor('A')
8 nopeus = 25
9 vahvistuskerroin = 1.5
10 musta = 28
11 valkoinen = 100
12 # Ohjelma
13 while True:
14     # Ohjaus = (heijastunut valo - ((musta + valkoinen)/2)) * vahvistuskerroin
15     valo = värianturi.get_reflected_light()
16     ohjaus = trunc((valo - ((musta + valkoinen)/2)) * vahvistuskerroin)
17     moottorit.start(ohjaus, nopeus)
18
19     # trunc(), pyöristetään nolaa kohti kokonaisluvuksi
20
```



# HARJOITUS 1 VIIVAN SEURAAMINEN

Toisinaan kulkureitin varrella on esteitä joita robotti joutuu väistelemään.

Harjoitellaan miten seurattavaa reunaa (V/O) ja viivanseurantaan käytettävää valokennoa vaihdetaan lennossa.



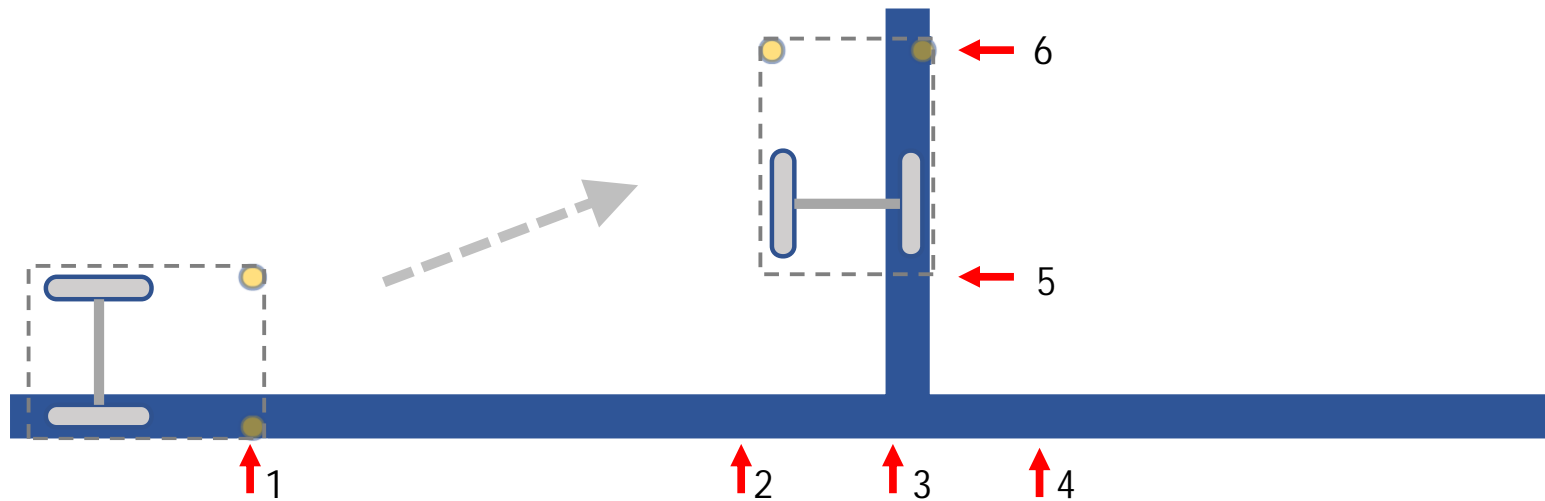
1. Tullaan ulos ohjelman silmukasta
2. Kaarretaan sopivasti tarvittavaan suuntaan (lyhyt pieni kaartto)
3. Jatketaan uudella viivanseurannan silmukalla





## Harjoitus 2

1. Seurataan viivaa käyttäen oikean puoleista anturia, silmukka.
2. Silmukasta ulos kun on kuljettu sopiva matka (lasketaan moottorin pyörimisiä kierroksia / asteita).
3. Ajetaan hyvin pienellä nopeudella aavistuksen vasemmalle kaartaen esim. 7 cm eteenpäin.



4. Vasen moottori peruuttaa hitaasti kunnes oikean puoleinen anturi näkee ensin valkoista (moottori käyntiin + odota) ja sitten mustaa (moottori käyntiin + odota), moottori seis.
5. Seurataan viivaa 10 cm käyttäen oikean puoleista anturia, uusi silmukka.
6. Sitten seis.

