

Kirjalle nimi!

Ensimmäiset robottini

Ohjelmointia 3-9

Pekka Pihola

Kirjoittanut Pekka Pihola

Kuvat Pekka Pihola

Sisältö

Esipuhe

Opettajalle

Sanasto

Mitä ohjelmointi on

Mindstorms ohjelmoinnin peruskurssi

1. Tutustutaan laitteisiin ja rakennetaan harjoitusrobotti
2. Moottorien ohjaaminen, ohjelman silmukat ja haarat, ääni
3. Valokenno, paikoitus, viivan seuraaminen, If-Then ja viive
4. Etäisyyden mittaaminen, paikoitus, kohteen etsiminen
5. Labyrintti
6. Esineen noutaminen, tarttujan rakentaminen, 3. moottori
7. Kosketusanturi, paranneltu tarttuja
8. Soft Sensors – virtuaaliset anturit, tehdään tarttujalle tuntoaisti
9. Gyro suunnistus
10. Muistipaikkojen käyttäminen, näyttö- ja ääni funktiot

Jatko1, Tutustutaan kilparobotteihin

11. Nopeusrampit
12. My Blocks, kapseloidut aliohjelmat
13. Pienien esteiden ylittäminen, esterata
14. Esteiden kiertäminen
15. Sumo
16. Tanssi
17. Pelastus
18. First Lego League
19. Kilpatehtävien analysointi
20. Voittoisat strategiat

Jatko2, Robotit ja arki

21. Robotti-autot ja liikennekaupunki
22. Vihivaunu
23. Robotti-imuri
24. 8-akselinen teollisuusrobotti
25. "Jätteiden" lajittelu

Jatko3, Kohti vaativampia haasteita

26. Portaavat / ylitys
27. Joy-stick kauko-ohjattu laite
28. Opetettava robotti
29. "Seg-Way"
30. Useampia tehtäviä yhdelle moottorille

Expert1, Java ohjelmointia

- 31.
- 32.
- 33.
- 34.
- 35.

Expert2, C-ohjelmointia

- 36.
- 37.
- 38.
- 39.
- 40.

Expert3, Projektityöt

- 41.
- 42.
- 43.
- 44.
- 45.

Mindstorms EV3

Ultraääni ja Gyro-anturit

Tekniikkavinkit rakentamiseen

Lähdeluettelo

Liitteet

- Käyttöopas EV3
- Robokisojen sääntöjä

Esipuhe

Fyysisen robotin rakentaminen on motivoiva ja hauska tapa tutustua ohjelmointiin. Tässä kirjassa tutustutaan ohjelmoinnin saloihin LEGO-Mindstorms EV3-robottien välityksellä. Harjoitusten robotit on pääsääntöisesti rakennettu raken-nussarjojen perusosista. Mikäli tästä on poikettu, siitä mainitaan harjoituksen esittelyn yhteydessä.

Luokilla 3-9 robottien ohjelmoinnissa käytetään kaikkien saatavilla olevaa ilmaista EV3:n Lab View -pohjaista EV3-G ohjelmointi kieltä. Juotoskolvia, työkaluja, tai mitään erityisiä ohjelmointi-ohjelmistoja, tai -taitoja ei siis tarvita. Luokille 7-9 tarkoitettuihin harjoituksiin on sisällytetty mahdollisuus käyttää myös tekstimuotoisia ohjelmointikieliä, kuten esimerkiksi C-kieltä, tai Javaa.

Harjoitukset on laadittu käytettäväksi perusopetuksessa, tai kerhoissa (alk.9v). Useimmat niistä soveltuvat myös itsenäiseen ohjelmointiin ja robotiikkaan tutustumiseen vaikka kotona. Luokille 3-4 laadittujen kurssien laajuus on 10 kertaa kahden oppitunnin kokonaisuus (10x1.5h). Luokille 5-9 tarkoitettut yksittäiset tehtävät ovat laajempia ja niiden tekemiseen on syytä varata enemmän aikaa. Harjoituksiin liittyy aina myös asiapitoinen alustus sekä kahta lajia kotitehtäviä, tehtyä harjoitusta tarkastelevia ja seuraavaan harjoitukseen valmistavia.

Tämä kirja sisältää myös joukon vinkkejä kilparobottien rakentajille – lajista riippumatta. Nämä vinkit ovat suuntaa antavia. Niitä ei pidä ottaa liian vakavasti, ehdottomina totuuksina. Mutta ne voivat olla hyödyllisiä myös harjoituksia tehtäessä.

Lajista riippumatta hyvän kilparobotin rakentaminen on joukkueelle aina ison innostuksen lähde. Mutta se on myös iso projekti, johon tullaan käyttämään aika paljon aikaa. Siksi on varmasti perusteltua miettiä ensin yhdessä, paitsi kuka haluaa tehdä mitäkin, myös sitä, miten tätä projektia lähdetään tekemään.

Ja sitten itse asiaan.

Opettajalle

Ohjelmoinnin opetuksessa lähtökohtina luovuus ja oppiainerajat ylittävä oppimiskokonaisuus

Ohjelmoinnin ja ohjelmoitujen laitteiden, kuten robottien toimintaperiaatteiden ymmärtämisestä on tullut uimataitoon verrattavissa oleva kansalaisten perustaito. Melkein kaikki suomalaiset osaavat uida, mikä antaa vahvan syyn uskoa myös ohjelmoinnin perustaitojen omaksumiseen.

Mihin ohjelmointia ja robotiikkaa tarvitaan?

Ohjelmointi on keino, jolla voidaan vaikuttaa johonkin, lähes poikkeuksetta reaali maailmaan liittyvään asiaan. Robotit puolestaan ovat helppo ja konkreettinen keino tuoda opetukseen ohjelman, laitteiden ja niiden välisen yhteyden merkitys ohjelmoinnin tavoitteiden toteutumiselle. Eli, jos laite tms. itsessään on susi, paraskaan ohjelmointityö ei sitä pelasta, mutta hyvästä laitteesta voidaan saada irti monin monin verroin enemmän hyötyjä korkeatasoisella ohjelmointiosaamisella ja -työllä.

Opetussuunnitelmassa

Uudessa Perusopetuksen opetussuunnitelmassa painotetaan monialaisia oppimiskokonaisuuksia, sekä pelillisyyden hyödyntämisestä oppimisen edistämisessä. Ohjelmointiin ja robotiikkaan viitataan matematiikan ja käsityön yhteydessä, ja näitä aiheita voi sisällyttää miltei kaikkiin oppiaineisiin. Nämä tavoitteet on otettu huomioon tämän aineiston laadinnan lähtökohtia määriteltäessä. Haaste on otettu vastaan.

Tähän aineistoon liittyen ja tukeutuen

Teknisissä käsitöissä voidaan mm. valmistaa erilaisia harjoituksiin liittyviä harjoituslustoja ja kilpalajeihin liittyviä tehtäväratoja. Teknisissä töissä

voidaan lisäksi valmistaa kehittyneimpiin kilparobotteihin omia antureita ja muita tarvittavia osia.

Tekstilikäsityön opetukseen voidaan liittää kilpalajeista Tanssi, kuvaamataitoon Freestyle. Musiikin tunnilla voidaan suunnitella uusia äänitiedostoja ja kuvaamataidossa kuvatiedostoja. Selkeiden toimintakuvauksien kirjoittaminen sivuaa äidinkieltä ja monet robotiikasta kertovat liitteet ovat englanniksi.

Geometria, mittayksiköt, koordinaatisto, erilaiset laskutoimitukset, sekä ongelmanratkaisutaitojen kehittyminen tukevat matematiikan opetusta. Valooppia, kitkaa, voimaa jne. sivuavat harjoitukset puolestaan tukevat fysiikan opetusta.

Toteutus

Alemmille luokille laadituissa tehtävissä käytetään graafista, erittäin laajasti teollisuudessa käytettyyn LabView kieleen perustuvaa LEGO® MINDSTORMS® EV3-G ohjelmointikieltä. Samaa laitealustaa hyödynnetään edelleen ylemmillä luokilla tehtävissä, joissa voidaan hyödyntää rivimuotoisien ohjelmointikielten, kuten Java:n ja C:n vahvuuksia.

Koulun kerhotoiminta ja erilaiset robottikilpailut

tähän tekstiä

Sanasto

Laite on laaja käsite. Se voi olla esim, näppäimistö, näyttö, hiiri, verkkokortti, tai tulostin. Roboteissa myös esim. anturi, painike, moottori, valo, jne.

Laiteohjain tunnistaa ja/tai muodostaa sähköisiä signaaleja. Siihen liittyy usein myös tähän kulloinkin liitettyyn laitteeseen liittyvä, sen kanssa käytävän vuoropuhelun säännöt (kommunikoinnin protokolla) sisältävä sisältävä pieni ohjelman pätkä, **laiteajuri**.

Murphyn laki

R. Daneel Olivaw

Picker

Ganrty

Portaali

Synkronoitu

samasto löytää muotonsa ja täydentyy lopuksi

Mitä ohjelmointi on

"Istuessani aamulla syömässä aamupalaa ojensin maidon tyttärelleni tämän pyynnöstä." Tilanne on kaikille tuttu, mutta mitä siinä oikeastaan tapahtui?

Sama Robottien maailmassa

1. C3PO esittää puhekielisen pyynnön "Saisinko maitoa?".
2. R2-D2 vastaanottaa pyynnön ääni-antureillaan, tulkitsee sen sisällön ja
3. keskeyttää muut askareensa.
4. Seuraavaksi R2-D2 tunnistaa ja 3D-paikantaa kohteen optisesti.
5. Nopean mutta tarkan laskelman perusteella R2-D2 tekee pienen käännöksen ja ojentaa tarttujan kohti kohdetta,
6. Tarttuminen kohteeseen pitää tehdä juuri oikealla voimalla, pakkausta murskaamatta, mutta riittävän lujasti.
7. Seuraavaksi R2-D2 nostaa maidon ja kääntyy, paikallistaen samalla C3PO:n.
8. Kohteen siirto päättyy sen varovaiseen laskemiseen C3PO:n ulottuville.
9. Nyt pitää irroittaa tarttuja ja loitontaa sitä varovasti kohteesta.
10. Lopuksi R2-D2 piipailee C3PO:lle ääniviestin "Tässä, ole hyvä", ennen
11. paluutaan keskeytyneiden arkareidensa pariin.

(toimintakuvaus kertoo vaihe vaiheelta suoritettavasta tehtävästä)

Keskimäärin jokainen meistä tulee opettelemaan eläessään noin kolmea puhuttua kieltä, ja ohjelmoimaan erilaisia laitteita kymmenillä eri ohjelmointikielillä/tavoilla. Ohjelmoinnin opiskelua voidaan verrata vieraan kielen opiskeluun, sillä erotuksella että puhutut kielet sisältävät paljon käsitteitä ja kuvaavia ilmauksia. Laajan sanaston oppimisessa on oma työnsä, mutta sen avulla asia kuin asia voidaan ilmaista lyhyesti ja tehokkaasti. Sen sijaan koneiden, kuten tietokoneiden ja robottien ohjelmoinnissa käytetty sanasto on olematon, josta syystä yksinkertaiseltakin tuntuvat asiat pitää ensin purkaa pienen pieniin osiin. Kukin osa, kuten vaikka robotin käden asentoa muuttava moottorin käynnistäminen, nopeus ja pysäyttäminen määritellään sitten ohjelmaan.

Jos osaat osaat ohjelmoida televisiosta tulevan ohjelman tallennuksen, osaat ohjelmoida jo yhdellä kielellä.

Ohjelmoitaessa vaikkapa robottia suorittamaan jotakin tehtävää, sen suorittaminen määritellään ohjelmaan vaihe vaiheelta käyttäen valitun ohjelmointikielen komentoja. Tässä tulee tietysti ottaa huomioon myös k.o. kielen rajoitteet, sekä ennen kaikkea itse kone. Eli mitä kyseinen kone voi/saa tehdä, ja mitä ei. Hyvässä ohjelmassa on lisäksi usein mietitty myös, mitkä kaikki asiat voivat mennä vikaan ja mitä koneen pitäisi sitten tehdä. Ulkoa oppimisen sijaan ohjelmoinnissa onkin kyse paljon enemmän tutkimisesta, oivaltamisesta ja hyviksi havaittujen ratkaisujen soveltamisesta.

Vaikka ohjelmointikieliä on tuhansia, ellei satoja tuhansia, niiden peruskäsitteistö ja -sanasto ovat hyvin vakiintuneita. Eroja kielten välille, eli miksi tarvitaan useita kieliä, on syntynyt lähinnä siksi, että on haluttu kehittää johonkin tiettyyn käyttötarkoitukseen yleisiä ohjelmointikieliä paremmin sopivia kieliä. Käytännössä näihin kieliin on lisätty joukko tätä tiettyä käyttötarkoitusta tukevia komentoja ja ominaisuuksia. Esimerkiksi joillakin kielillä on muita helpompaa ohjelmoida nettisivuja ja toisilla vaikka pelejä.

Peruskomennot

- Matemaattiset perustoiminnot,
- Loogiset perustoiminnot
- Aikaan liittyvät komennot,
- ohjelmaan syötettävään / ohjelmaan muuta kautta luettavaan tietoon liittyviä komentoja Input
- Ohjelman suorittamiseen liittyvät
- Muistin käsittelyyn liittyvät
- Aikaansaanoksen käsittely, Output, laiteohjaimien komennot

Harjoituksissa käyttämällämme ohjelmointikielellä on muutamia ominaisuuksia, joiden ansiosta se soveltuu erityisen hyvin ohjelmoinnin alkeiden opiskeluun. Ensimmäinen ja merkittävin etu on sen havainnollinen tapa tuoda esille ohjelman kulku, suoritusjärjestys, rinnastus vuokaavioon

toinen on sen käyttäjäystävällisyys, perustuuhan se ihan oikeaan ja laajasti käytössä olevaan ohjelmointikieleen. kokemuspohjaa ja ...

Mahdollisuus seurata reaaliaikaisesti ohjelman suorituksen etenemistä on erittäin havainnollinen tapa tutustua ohjelmoitujen laitteiden tapaan toimia. Se myös nopeuttaa ohjelman bugien löytymistä ja viimeistelyä

Yleiset ohjeet harjoitusten tekemiseen

Tässä kirjassa ohjelmointia lähestytään tekemisen kautta, kokeilemalla ja havainnoimalla. Nämä ovat yleiset ja yhteiset ohjeet kaikkien tämän kirjan harjoitusten tekemistä varten.

Etkot

Harjoitukset edellyttävät monasti pientä valmistautumista, jotta ne voidaan viedä läpi sujuvasti. Näihin etukäteis valmistautumisiin kuuluu aina myös robottien akkujen lataaminen. Tästä ei mainita erikseen enää myöhempanä.

Intro

Intro on tärkeä opettajan / ohjaajan pitämä 10-15 minuutin mittainen asiapitoinen alustus harjoitukselle.

Tehtävät

Tehtävät on tarkoitettu tehtäväksi parityöskentelynä. Niitä on varattu pääsääntöisesti 3kpl kutakin harjoitusta kohden. Lisäksi on aina varattu 1-2 syventävää lisätehtävää nopeimmille ryhmille.

Jos rakennustehtävästä on esitetty kuvallisia ohjeita, niiden noudattaminen on yleensä tärkeää myös seuraavien tehtävien onnistumisen kannalta. Myöhemmin, rakentamisen perustekniikoiden jo tultua tutuiksi, ohjeita on vähemmän. Näin saadaan tilaa luovalle suunnittelulle ja ongelmanratkaisulle, omien ideoiden kehittämiseksi ja jo opittujen käytäntöjen soveltamiselle, eli innovoinnille.

Ohjelmointitehtävissä kokonaisuus on syytä purkakaakaan ensin pieniin osiin ja laatia ohjelmasta toimintakuvaus vihkoon. Toimintakuvauksessa käytetään tuttuja, yksinkertaisia tekemistä täsmällisesti kuvaavia sanoja ja ilmaisuja, kuten: mene, käänny, etsi, nosta, työnnä, vedä, paina, tartu, irroita, käännä, pyöritä, ääni, poimi, kerää, kuljeta, etsi mustan viivan reuna, seuraa viivaa, miten pitkään, käänny, miten paljon, paikallaan vai kaartaen, laske tarttuja, nosta tarttuja jne.

Kiinnitäkää samalla huomiota siihen, jos jokin toiminto esiintyy toistuvasti (silmukat/aliohjelmat), vaihtoehtoisin tapahtumien kulkuihin (ehdollinen ohjelman haarautuminen), sekä samanaikaisesti suoritettaviin toimintoihin.

Kirjoittakaa samalla robotin rakeenteen suunnittelua silmällä pitäen luettelo niistä taidoista, joita robotin tulee osata.

Harjoituskerran lopuksi, sen mukaan miten halutaan edetä, sekä muut opetus-, tai kerhoryhmät huomioiden laittet joko puretaan, tai jätetään purkamatta. Laaditut ohjelmat sensijaan kannattaa aina taltioida, sillä niistä voi käydä myöhemmin kopioimassa uusiin ohjelmiin hyviä, jo toimiviksi havaittuja osuuksia.

Jatkot

Jatkoihin kuuluu kotitehtäviä ja vinkkejä seuraavan harjoituskerran aiheeseen ennalta tutustumiseen.

Mindstorms ohjelmoinnin peruskurssi

Seuraavien kymmenen harjoituskerran aikana tutustutaan ohjelmoinnin alkeisiin ja robotin rakentamisen, sekä joihinkin robotiikassa yleisesti käytettyihin antureihin.

Tehtävissä lähdetään ihan perusasioista. Tietoteknistä osaamista ei käytännössä edellytetä alussa. Taidot kyllä kehittyvät harjoitusten myötä. Alussa ohjelmointiin tutustutaan malliohjelmien kautta ja taitojen karttumisen myötä siirrytään vähitellen itsenäiseen ongelmanratkaisuun ja ohjelmointiin.

Tässä harjoituksessa tutustutaan tavallista perusteellisemmin jopa mittauksien ja kokeiden kautta robotin antureihin. Oppilaat oppivat toimimaan mitatun tiedon pohjalta ja selvittämään puuttuvat tiedot.

1. Tutustutaan laitteisiin ja rakennetaan harjoitusrobotti

Etkot

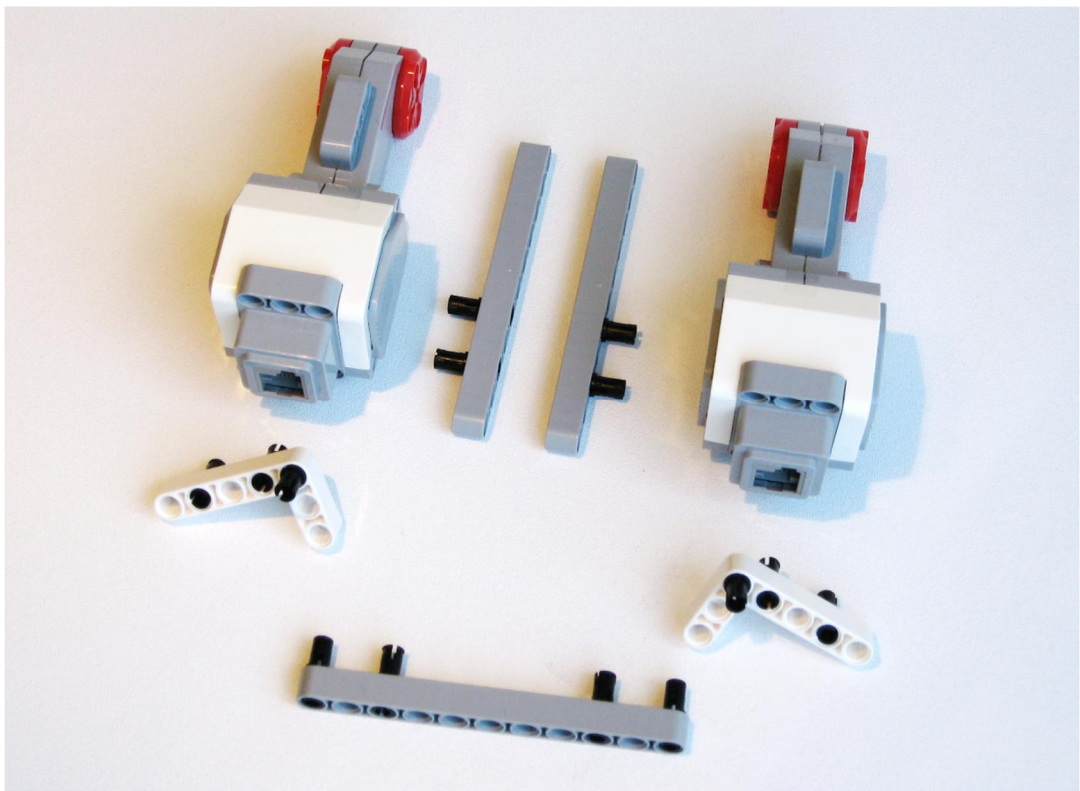
Opetuksessa käytettäville tietokoneille tulee olla asennettuna EV3:n ohjelmointi-ohjelman uusin versio (www.lego.com, Tuotteet, Mindstorms ja Downloads).

Intro

- o Käydään läpi EV3 ohjaimen käynnistäminen, sammuttaminen ja valikot (käyttöohje, liitteenä, sivut 7, 9, 10 ja 16)
- o Ohjelmointikäyttöliittymän pääalueiden esittely, laitesivu käydään läpi tarkemmin (käyttöohje, liitteenä, sivut 35-39)

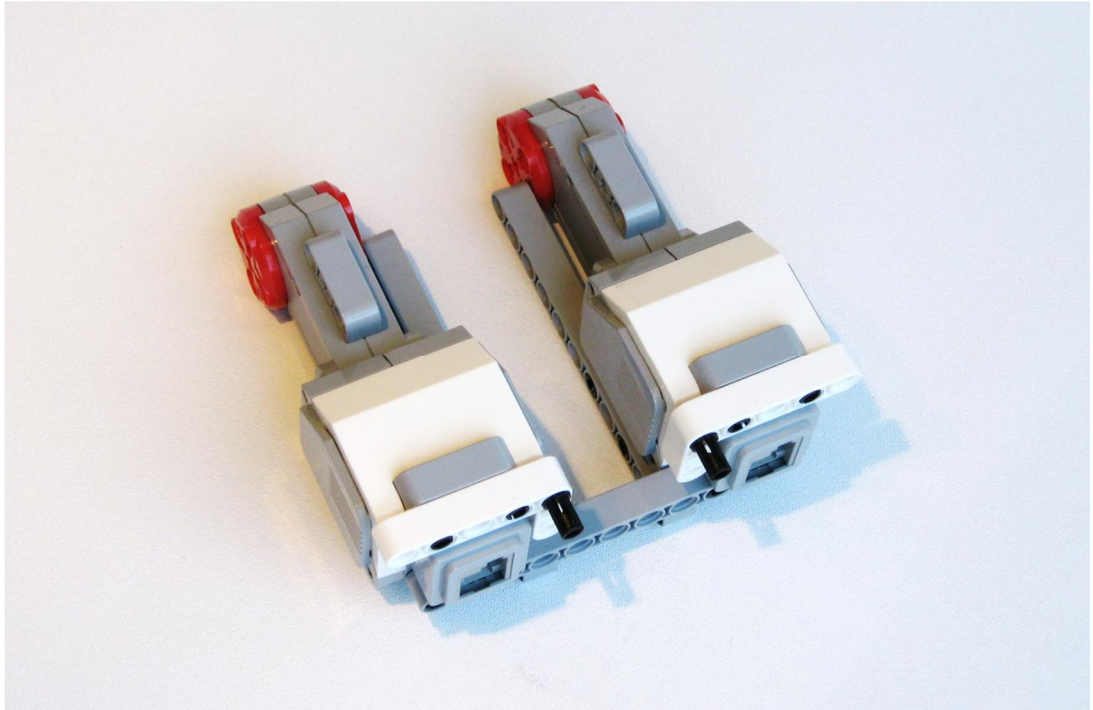
Tehtävä1

Rakenna harjoitusrobotti seuraavan ohjeen mukaan

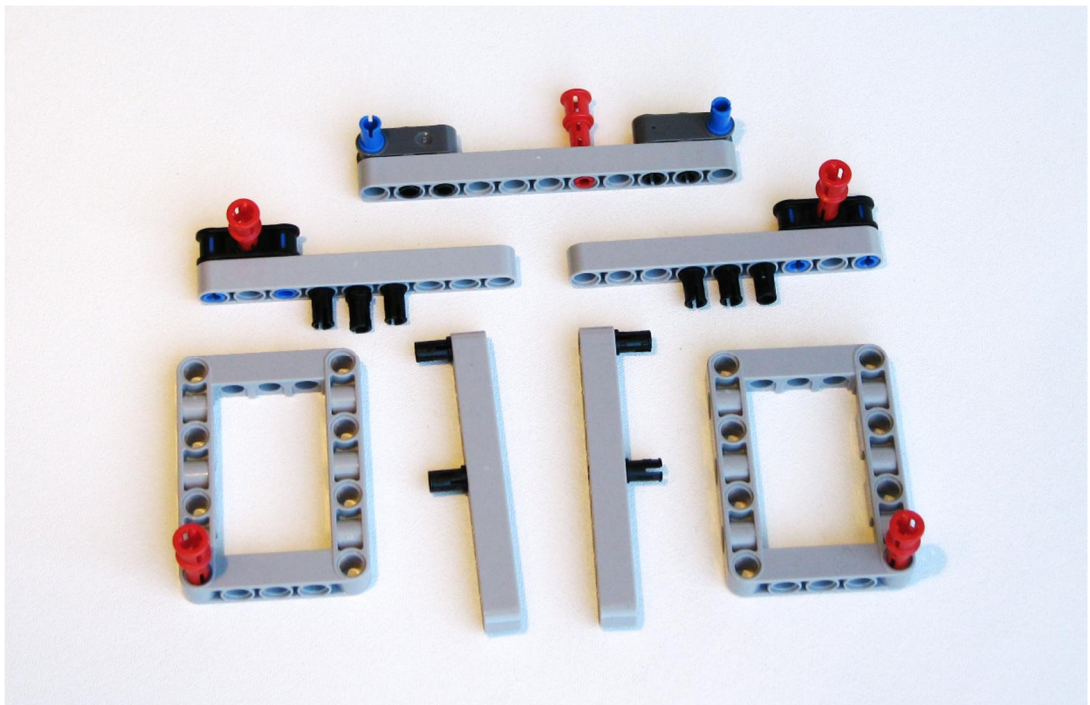


(kuva)

LEGO-järjestelmän perusmitta on "nuppi" = 8mm. Tässä kuvassa esimerkiksi kaikki harmaat palkit ovat 11 nuppia pitkiä. Kerää osat ja rakenna kuvan mukaan.

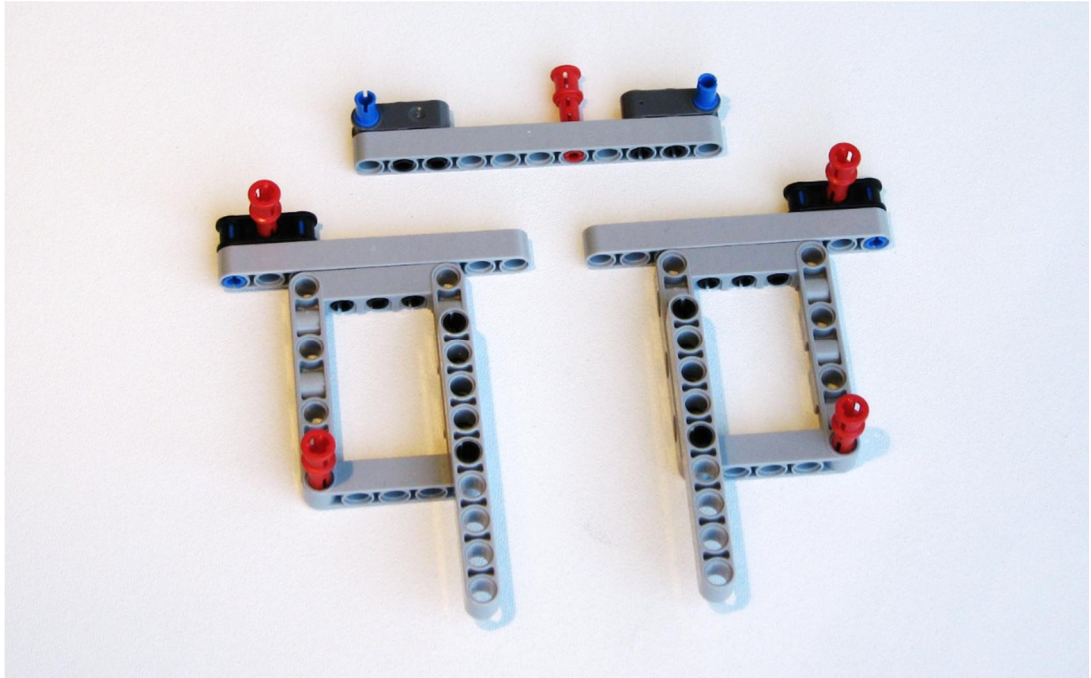


(kuva)

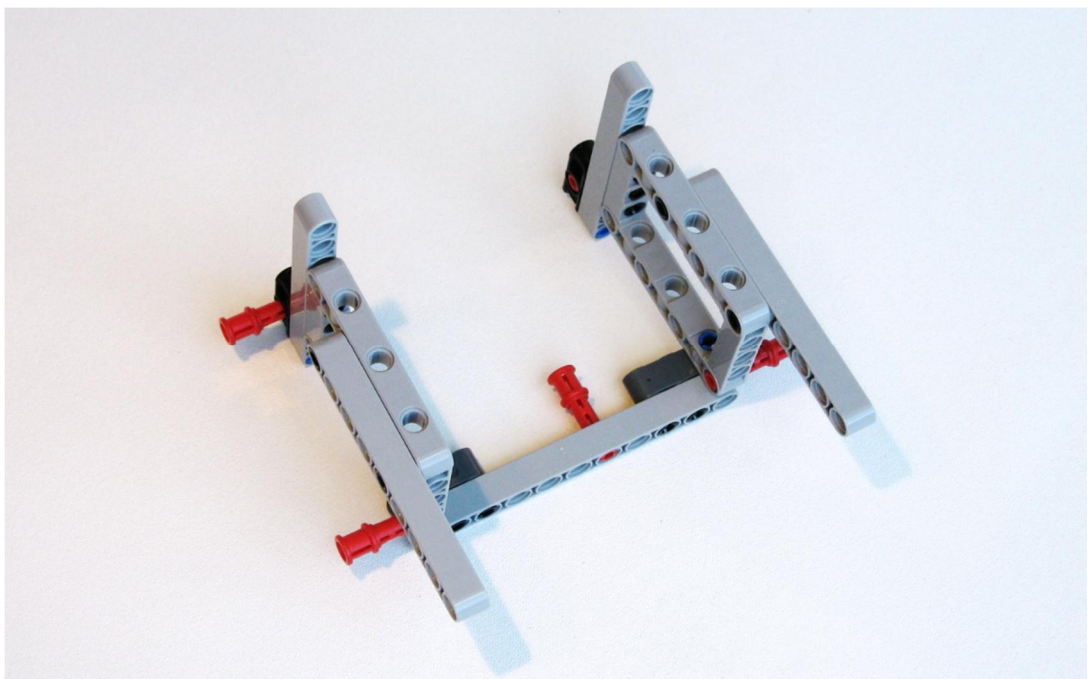


(kuva)

Pieniä liittostappeja kutsutaan "pinneiksi". Pinnejä on kahta perustyyppiä ja montaa eri mallia, jotka tunnistaa toisistaan helpoiten värin perusteella. Harmaat ja vaalean ruskeat (tan) edustavat vapaasti pyörivien pinnien ryhmää, joita käytetään kun halutaan jonkin osan liikkuvan herkästi. Useimmat pinnit kuuluvat kitka-elementeillä varustettujen pinnien ryhmään, kuten mustat, siniset ja kuvan punaiset vetopinnit. Joissakin pinneissä, kuten kuvan lyhyissä sinisissä pinneissä (6kpl), toinen pääty on muotoiltu LEGO-akselin muotoiseksi.



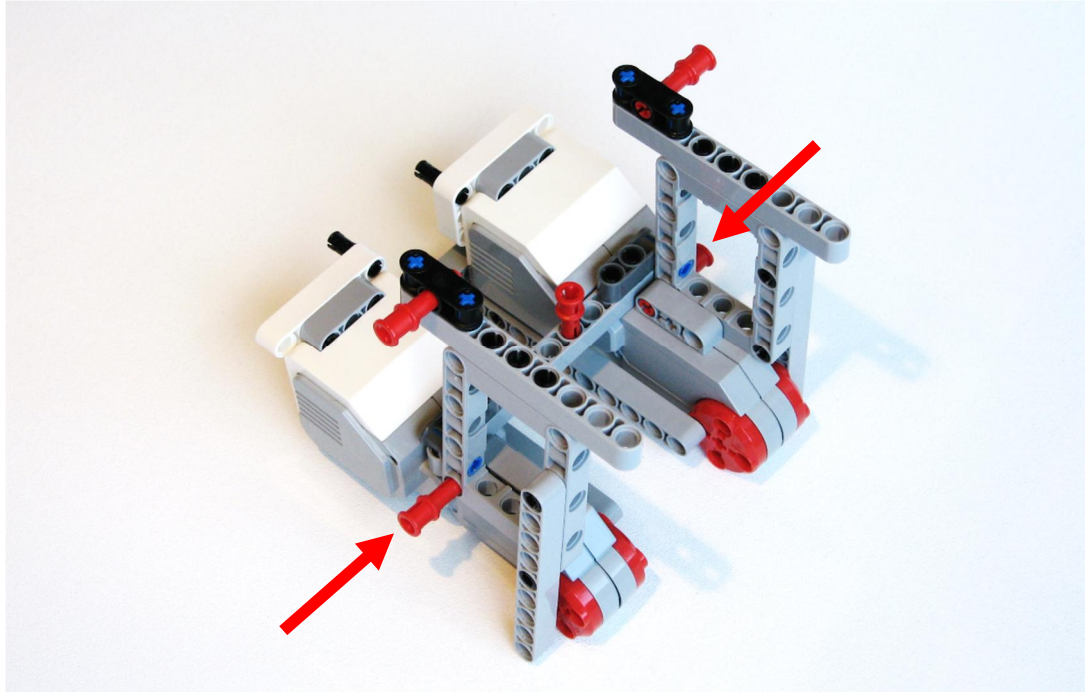
(kuva)



(kuva)

Runko alkaa tässä jo hahmotua, joskaan se ei vielä ole kovin tukeva. Vetopinnejä käytetään tyypillisesti tilanteissa joissa joko halutaan liittää jotakin rakenteeseen vasta myöhemmin, tai kun halutaan tehdä rakenne josta jokin osa voidaan irroittaa helposti. Tai molempia.

Jos vetopinni on kuvassa ulommassa asennossa, kuten kaikki tämän kuvan 5 vetopinniä, se on myös tarkoitus.



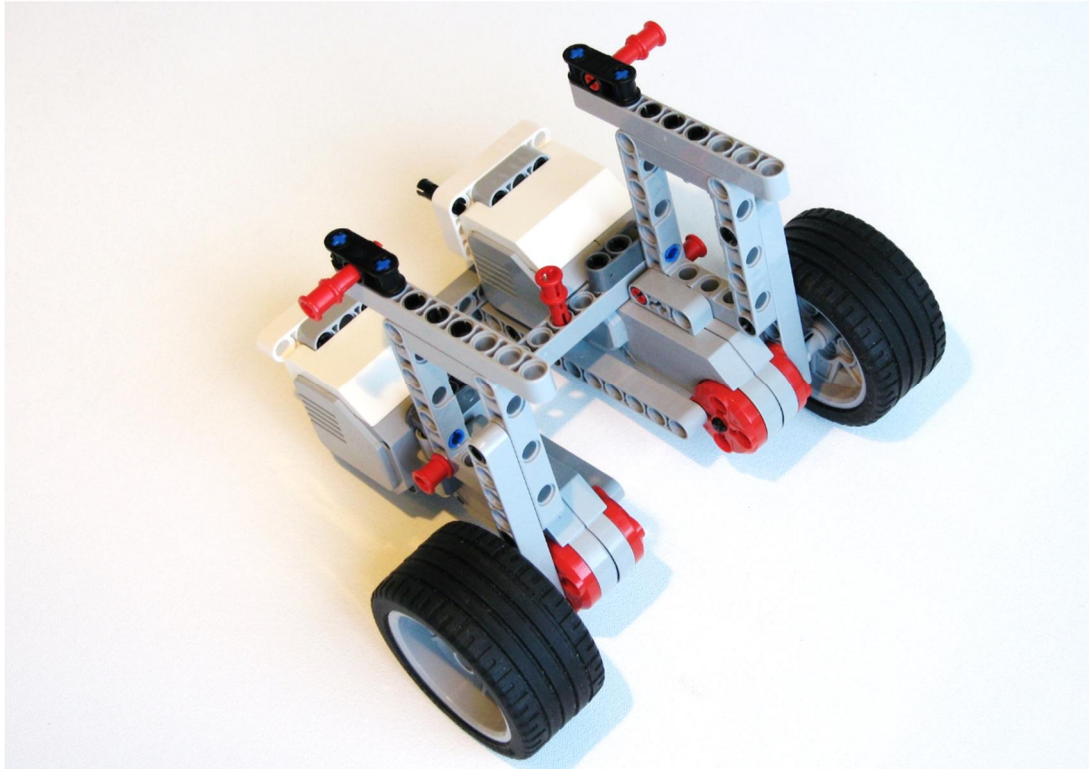
(kuva)

Liitä nyt runko moottoreihin työntämällä kuvassa osoitetut kaksi vetopinniä sisään.



(kuva)

Kokoa renkaat, akselit ovat 6 nuppia pitkiä. Näiden EV3-Education sarjan mukana tulevien renkaiden ulkohalkaisija on n.56mm. Tästä saadaan yhtä moottorin pyörähdystä vastaavan matkan pituudeksi laskemalla $56 \times 3.14 \approx 176\text{mm}$



(kuva)

Renkaat asennettuina. Robotin painavin osa on ohjainyksikkö. Se liitetään myöhemmin rungon päälle. Moottorien toiminnan ja kulumisen kannalta on edullista, mitä vähemmän se joutuu kantamaan robotin painosta. Tässä rakennettavan robotin painosta enin osa välittyy nyt ohi moottorien, suoraan pyörien akseliin. Robotin ollessa painavampi on sen pyörille syytä rakentaa tuenta molemmille puolille.



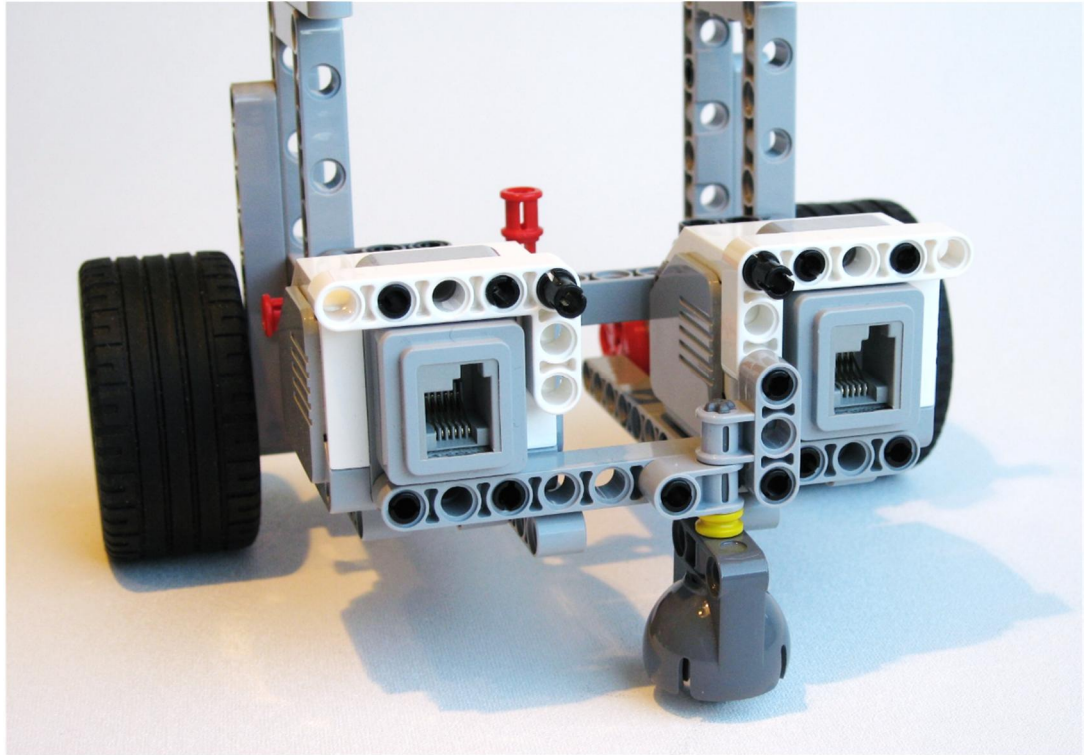
(kuva)



(kuva)

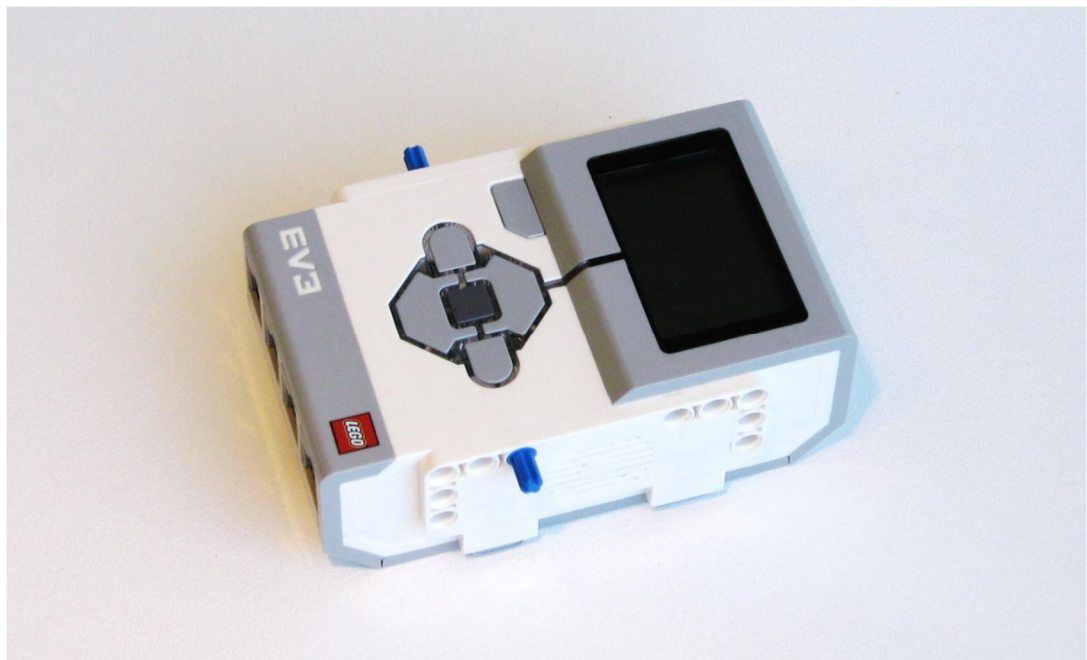
Robotissa on nyt kaksi pyörää, mutta koska emme ole (vielä) tekemässä tasapainoilijaa, tarvitaan vielä ainakin yksi pyörä tms. Hyviä vaihtoehtoja on useita. Tällä kertaa käytetään pieneen tilaan mahtuvaa Caster Ball:ia, eli kaikkiin suuntiin vapaasti pyörivää palloa.

Caster Ball:in voi kiinnittää monella tavalla. Tämä ratkaisu mahdollistaa korkeuden säätämisen lisäämällä, tai vähentämällä keltaisia holkkeja.



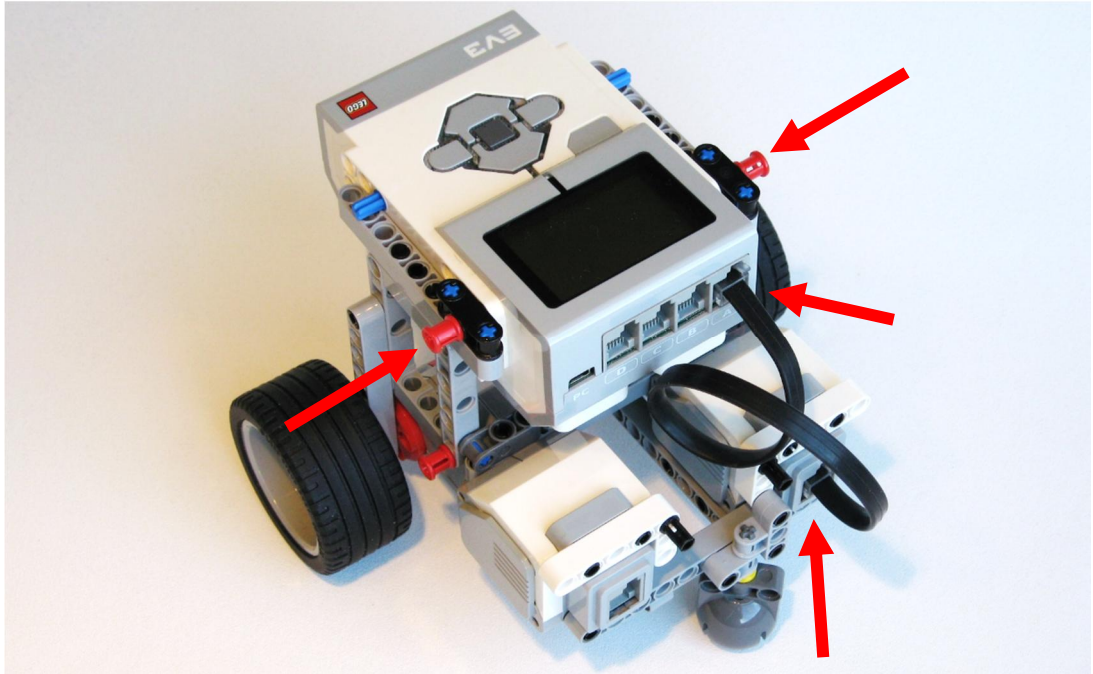
(kuva)

Caster Ball asennetaan hiukan epäkeskeisesti, jotta tähän päätyyn saadaan myöhemmin mahtumaan muitakin laitteita. Robotin tulisi kulkea nyt jotakuinkin vaakasuorassa asennossa.



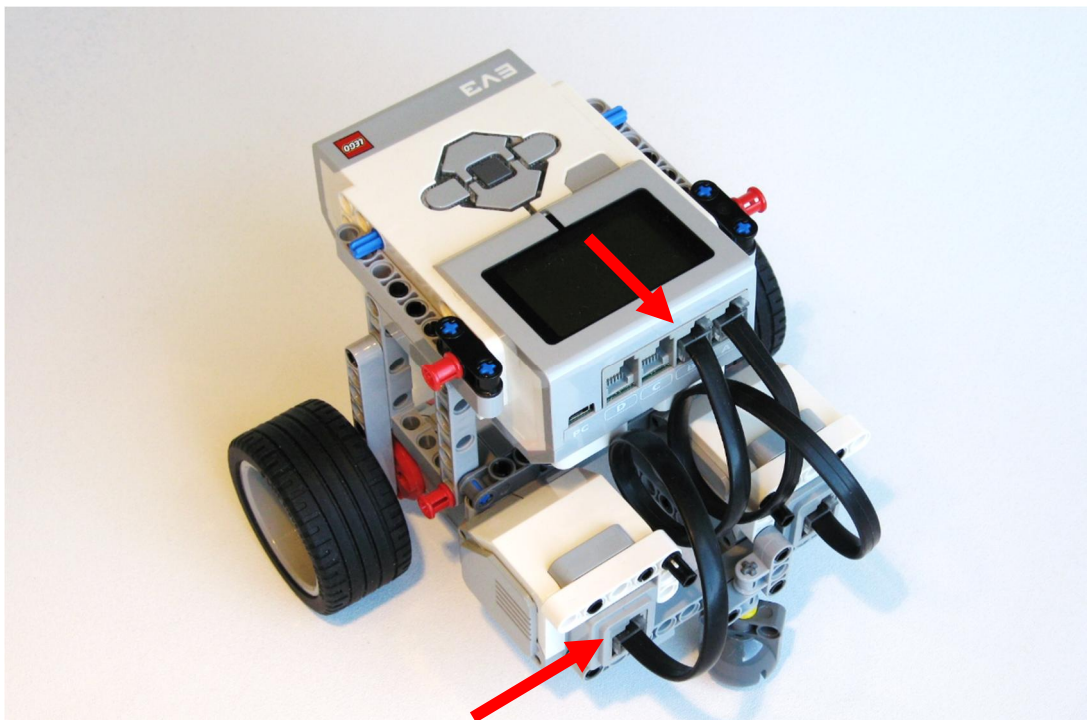
(kuva)

Keskusyksikkö, älypalikka, ohjain, EV3-brick, sillä on monta nimeä. Tärkeimmät painikkeet ovat keskimmäinen (käynnistys/valitse) ja näytön alla, vasemmalla oleva (peruuta/sammuta) -painike. Muita painikkeita käytetään valikoissa liikkumiseen. Keskialueen viidelle painikkeelle voi myös ohjelmoida vapaasti suoritettavalle ohjelmalle välitettäviä komentoja.



(kuva)

Kiinnitetään ohjain robottiin ja kytketään A-moottori ohjaimen A-porttiin. Ohjelmointikaapelin mini USB portti on saman rivin vasemmassa reunassa (pieni). Robottia rakennettaessa on hyvä ottaa huomioon ohjelmointikaapelin ja akun latauskaapelin liittämisen / paristojen vaihtotyön helppous.

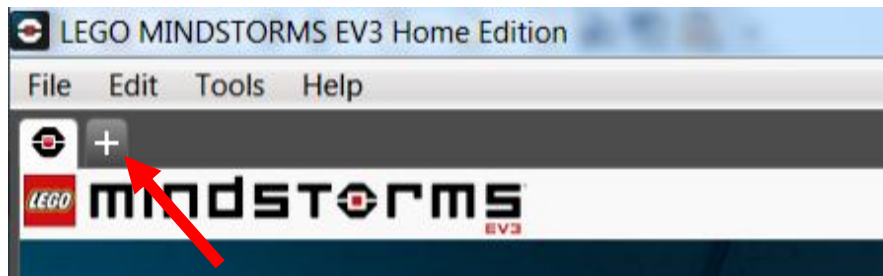


(kuva)

Kytetään vastaavasti B-moottori ohjaimen B-porttiin. Johtoja voi pyöräyttää siististi pienille lenkeille ja sovitella rakosiin, etteivät ne olisi tiellä. Ensimmäinen harjoitusrobotti on nyt valmis.

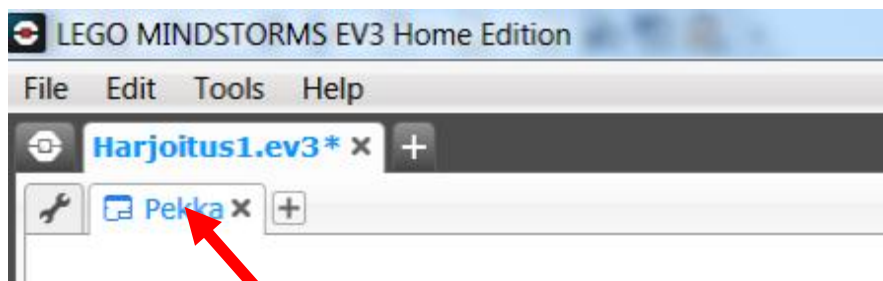
Tehtävä2

Aloita avaamalla ohjelmointiohjelma ja uusi projekti.



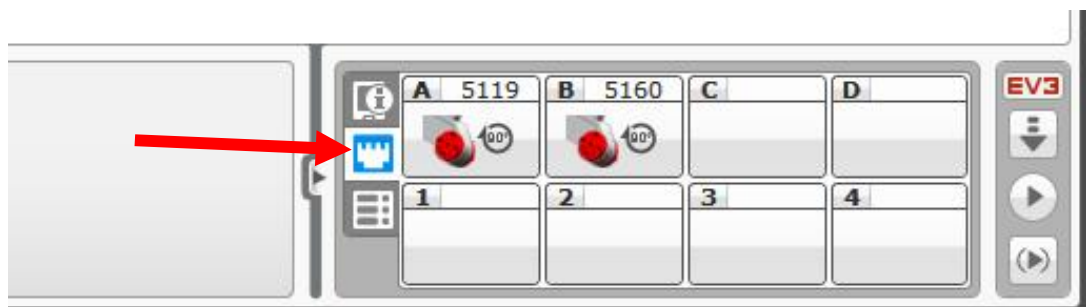
(kuva)

- Annetaan projektille nimi tallentamalla se nimellä Harjoitus1 (File->Save As)
- Nimeä ohjelma omalla nimelläsi, (tuplaklikkaa ohjelman nimeä)



(kuva)

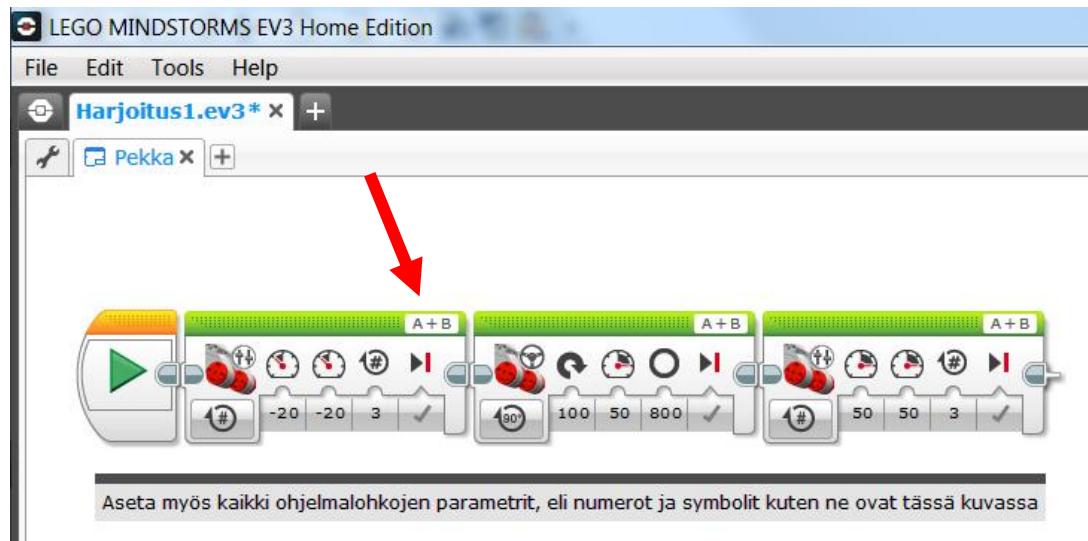
Tarkastetaan vielä kytkennät, molempien moottorien tulisi nyt näkyä myös ohjelmointiohjelman laitemonitorin portti-näkymässä (klikkaa punaisen nuolen osoittamaa kuvaketta)



(kuva)

Tee tarkasti alla olevan mallikuvan mukainen ohjelma. Huomaa laittaa ohjattavien moottorilähtöjen tunnuksat kohdilleen joka kohtaan.

(kuva)

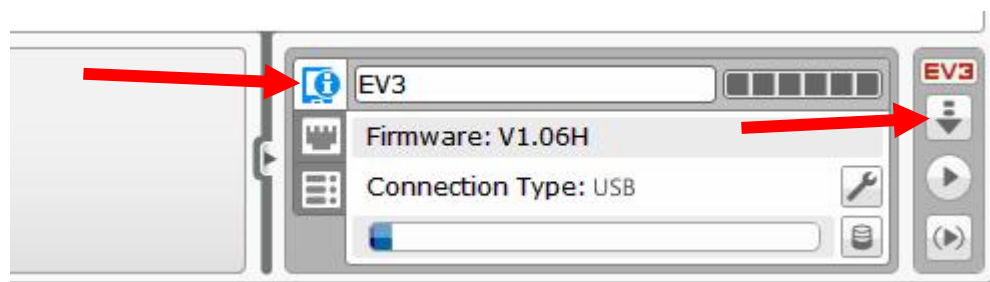


(kuva)

HUOM! Laske robotti nyt lattialle siten että sille jää tilaa ympärilleen.

Käynnistä ohjain käynnistyspainikkeesta ja odota merkkiääntä. Tähän menee tovi, n. 40s.

Siirrä ohjelma robottiin käyttäen USB-kaapelia.



(kuva)

Irroita USB-kaapeli robotista, hae robotin ohjelmakansioista Harjoitus 1 ja käynnistä ohjelmasi.

Mitä robotti teki?

Tehtävä3

Ohjelmoi robotti

- kulkemaan ensin 40cm suoraan,
- kääntymään kohtisuoraan oikealle,
- kulkemaan 30cm suoraan,
- kääntymään oikealle niin paljon, että robotti osoittaa nyt kohti lähtöpaikkaansa.
- Kulkemaan 50cm suoraan ja
- lopuksi antamaan vapaavalintaisen äänimerkin.
- Tutustu samalla mahdollisuuteen seurata ohjelman kulkua reaaliajassa tietokoneen näytöltä. Tämä on hyödyllinen ominaisuus ongelmakohtia paikallistettaessa, jos jonkin ohjelman suorittaminen esimerkiksi jumittaa joskus.

Lisätehtävä 1

Miten tarkasti robotti tuli takaisin lähtöpisteeseensä?

Kokeile onnistutko parantamaan tarkkuutta. Robotin servomoottorit lähettävät ohjaimelle pulssitietoa aina niiden akselin pyöriessä. Ohjain laskee nämä pulssit ja tätä tietoa voidaan käyttää esimerkiksi robotin pysäyttämiseen juuri oikeaan kohtaan. Voitko vaikuttaa pysähtymisen tarkkuuteen muuttamalla robotin nopeutta? Kokeile.

Jatkot

Tehtävä: Tulet tarvitsemaan laajempaa tietoa EV3:n moottorien ohjaamisesta tulevia harjoituksia tehdessäsi. Etsi Internetissä videoita käyttäen hakusanoina esimerkiksi seuraavia:

- EV3 programming motor
- EV3 ohjelmointi moottori

Vastaukset, Tehtävä2

Robotti kulkee ensin rauhallisesti noin 53cm eteenpäin ja pyörähtää paikallaan kaksi täyttä kierrosta vastapäivään. Lopuksi robotti palaa lähtöpaikalle.

Vastaukset, Lisätehtävä 1

Robotin tarkkuuten vaikuttaa moni tekijä, tässä joitakin tunnettuja. Moottorien pulssilaskennassa voi etenkin suurilla nopeuksilla, kiihtyvyyksillä ja massoilla syntyä pieniä virheitä. Kovissa kiihdytyksissä ja jarruruksissa pyörät liukuvat, ja on käynyt niinkin että rengas on pyörähtänyt vanteella. Nopeuden pitäminen aluella 20-75% auttaa asiaan, pienempi parempi.

Käännöksiä tarkkuuteen vaikuttaa myös raideleveys ja renkaiden leveys siten, että suurempi raideleveys ja pienempi renkaiden leveys yleensä parantavat tarkkuutta.

2. Moottorien ohjaaminen, ohjelman silmukat ja haarat

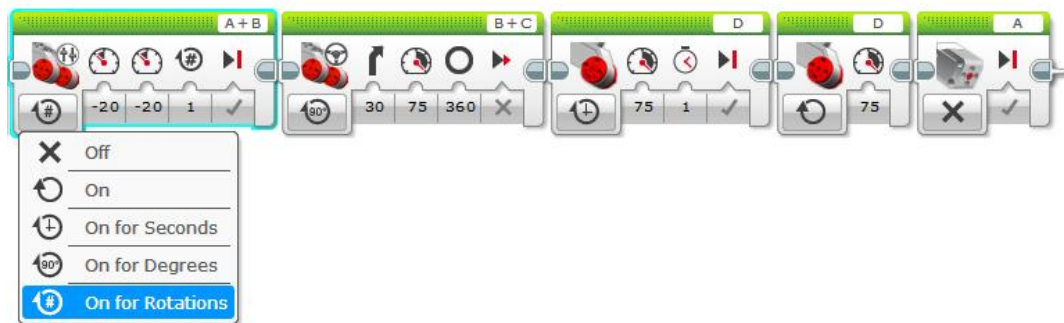
Etkot

Harjoitukset jatkuvat käyttäen ensimmäisessä harjoituksessa rakennettua robottia. Kerrataan moottori-, viive-, sekä silmukkalohkojen parametointi.

Intro

Ensimmäisessä harjoituksessa kokeilimme kahta erilaista kahden moottorin ohjaamiseen tarkoitettua ohjelmalohkoa. Tämän harjoituksen aluksi käydään läpi moottorilohkojen asetukset.

Seuraavassa kuvassa on esitetty viisi moottorilohkoa.



(kuva)

1. ensimmäisenä on Move Tank moottorilohko, jolla käsketään tässä

(moodi = On for Rotations)

- a. moottoreita A ja B pyörimään synkronoidusti
- b. taaksepäin 20% nopeudella (-20)
- c. yhden kokonaisen kierroksen (1) verran ja
- d. asettamaan lopuksi sähköisen jarrunsa päälle (✓),
- e. ennen ohjelman suorituksen siirtymistä eteenpäin.



2. toisena on Move Steering moottorilohko, jolla käsketään tässä
(moodi = On for Degrees)

- robottia kaartamaan 30% jyrkästi (30)
- käyttäen moottoreita B ja C synkronoidusti,
- suhteellisella nopeudella 75% (75),
- nimellisesti 360 asteen (360) verran ja
- jäämään lopuksi vapaapyörintä -tilaan (X),
- ennen ohjelman suorituksen siirtymistä eteenpäin.



3. Kolmantena on Large Motor moottorilohko, jolla käsketään tässä
(moodi = On)

- moottoria D pyörimään
- eteenpäin 75% nopeudella (75)
- yhden sekunnin (1) ajan ja
- asettamaan lopuksi sähköisen jarrunsa päälle (✓),
- ohjelman suoritus siirtyä välittömästi eteenpäin.



4. Neljäntenä on Large Motor moottorilohko, jolla käsketään tässä
(moodi = On for Rotations)

- moottoria D pyörimään
- eteenpäin 75% nopeudella (75)
- määräämättömän ajan verran.
- Ohjelman suorituksen siirtymistä eteenpäin.



5. Viidentenä on Medium Motor moottorilohko (pienempi moottori),
jolla käsketään tässä (moodi = Off)

- moottoria A pysähtymään välittömästi
- asettamaan lopuksi sähköisen jarrunsa päälle (✓),
- ohjelman suoritus siirtyä välittömästi eteenpäin.



Lisää tietoa ja vinkkejä löydät klikkaamalla ohjelmointi ohjelman ylälaidasta
Help -> Show EV3 Help -> Programming Blocks -> ja valitse tyyppi valikosta.

Polku

”Polku vie suunnistajaa rastilta rastille,
ohjelmaa käskyltä käskylle.”

Intron loppuksi opetellaan vielä ohjelmalohkojen kytkemistä toisiinsa polulla. Tietotekniikan yleiskielisissä (englanti) kirjoituksissa ohjelmalohkoja toisiinsa kytkevästä viivoista käytetään usein sanaa wire, joka voidaan tulkita langaksi, johtimeksi, tai vaijeriksi. Suomenkielistä sanaa polku voidaan kuitenkin pitää tässä asiayhteydessä vähintään yhtä kuvaavana.

Polku on ohjelmassa yksisuuntainen reitti, jolla voidaan yhdistää kaksi ohjelmalohkoa, jota pitkin ohjelman suoritus kulkee. Joskus se haarautuu, mutta aina sillä on alku ja loppu.

Polun piirtäminen

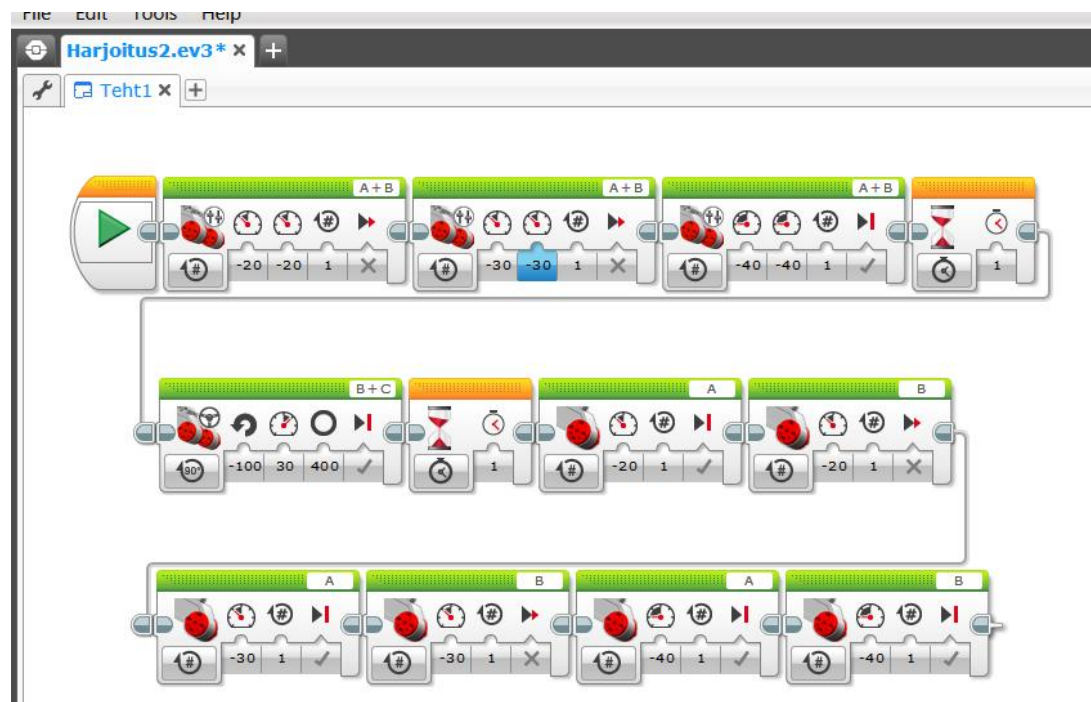
Vie hiiren osoitin ohjelmalohkon oikeassa reunassa olevan ”liittimen” päälle (hiiren osoitin vaihtuu lankarullaksi). Paina hiiren vasen painike alas ja vedä lankarulla ohjelmassa seuraavaksi suoritettavan lohkon vasemman reunan liitimeen. Tämä vaatii tarkkaa kättä. Liitos syntyy vasta kun liitin vaihtaa värinsä siniseksi.

- Polun saa pois klikkaamalla sen loppua.
- Ohjelmalohkot saa siirtymään yhteen ja niiden välissä olleen polun katoamaan klikkaamalla polun alkua.

Tehtävä1

Tutustutaan moottorilohkojen toimintaan. Harjoitusta varten lähdemme tekemään ohjelmaa, jossa robotti ajaa ensin asteittain kiihdyttäen noin 50cm, kääntyy ja palaa asteittain kiihdyttäen takaisin. Menomatka ja kääntyminen tehdään kahden moottorin moottorilohkoilla, paluu yhden moottorin moottorilohkoilla.

Tee alla olevan mallin mukainen ohjelma.



(kuva)

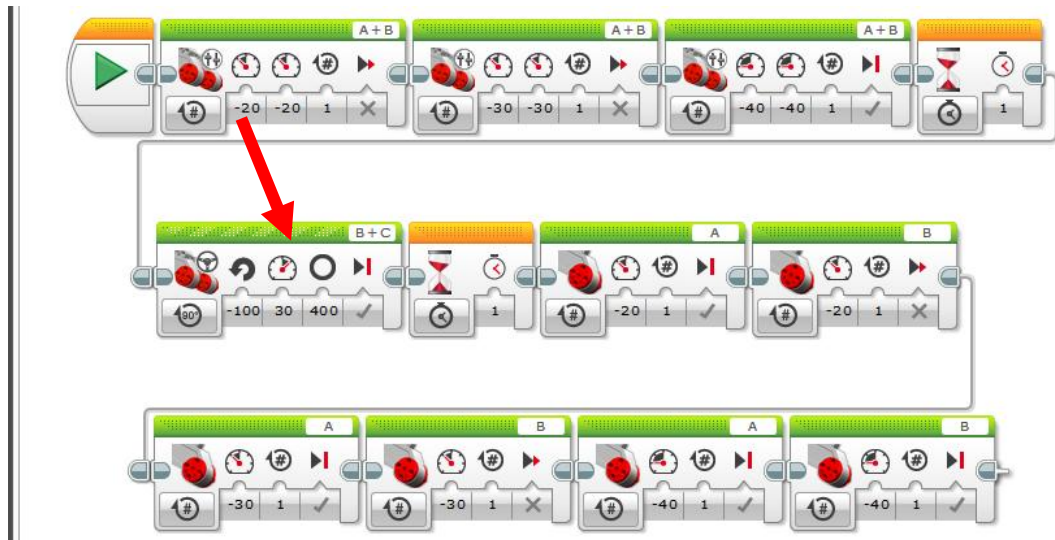
Laske robotti lattialle, lataa ohjelma ja kokeile sitä.

Robotti ei tullut takaisin lähtöpisteeseensä. Ohjelmassa on nyt virhe.

EV3:een liittyvä mahdollisuus tarkastella sen toimintoja reaaliaikaisesti on osoittautunut käteväksi avuksi ohjelmien vikojen etsimisessä ja muutenkin ohjelmien kehittämisessä.

Pysäytetään ohjelman suoritus, kytketään ohjelmointikaapeli takaisin kiinni robottiin ja käynnistetään ohjelma uudestaan. Seuraa nyt ohjelman kulkua tietokoneen näytöltä. Ohjelmalohkon, jota kulloinkin suoritetaan, tunnistat sen yläpalkissa juoksevasta raidoituksesta.

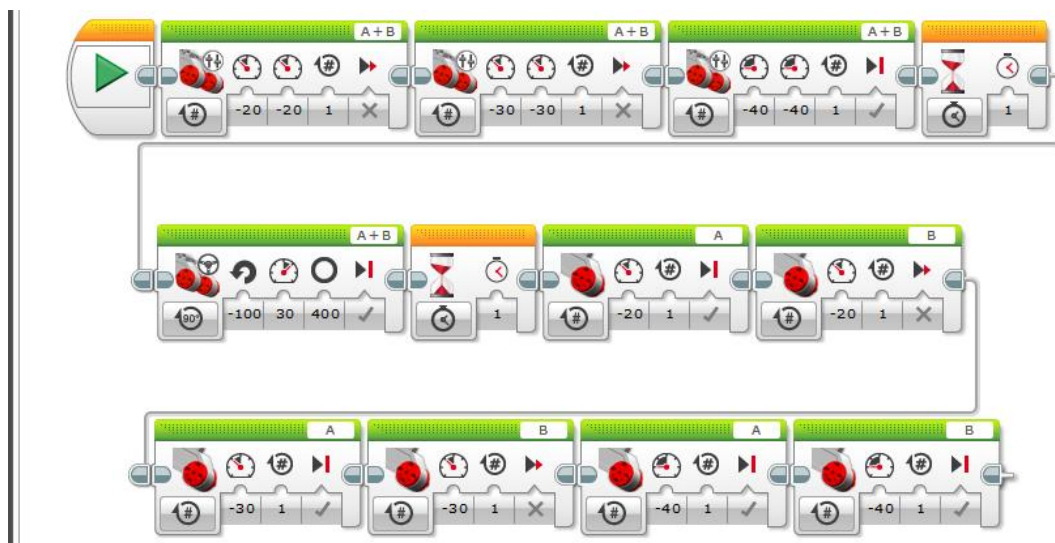
Ohjelman suorituksen pitäisi jumittaa nyt viidennen ohjelmalohkon kohdalla.



(kuva)

Kääntymiseen käytetyn kahden moottorin moottorilohkon suoritus jäi kesken. Se vertailee moottorien pyörimisen keskinäistä suhdetta, ja nyt se ei saanut minkäänlaista pyörimisestä kertovaa tietoa lähtöportista C (käytettävien moottorien määrittely oli tehty virheellisesti).

Korjaa käytettävien moottorien määrittely nyt oikeaksi (A ja B).

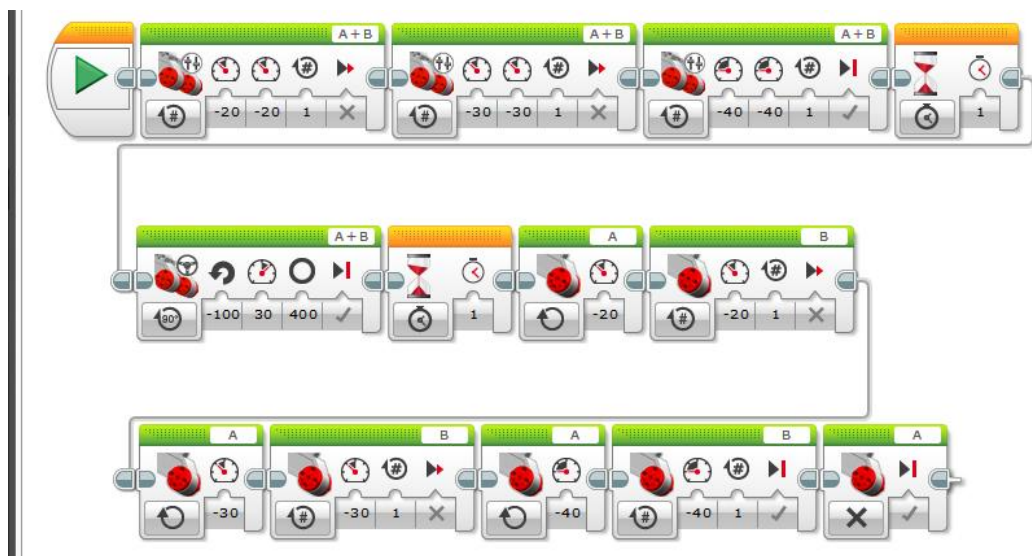


(kuva)

Kokeillaan jälleen, palaako robotti nyt takaisin lähtöpaikalleen?

Nyt robotti kääntyi, mutta kierturtelee paluumatkalla, mikä ei ollut tarkoituksena. Kun seurataan ohjelman kulkua, voidaan havaita suorittavan aina A-moottoria koskevan komennon ensin loppuun asti, ennen B-moottorin vastaavaa komentoa.

Muutetaan seuraavaksi ohjelmaa siten, että ohjelma vain käynnistää A-moottorin ja antaa välittömästi B-moottorille käskyn, joka pitää sisällään tiedon myös jakson kestosta. Lopuksi pitää A-moottori vielä pysäyttää erikseen.



(kuva)

Kokeillaan jälleen ohjelman toimintaa. Seuraa robotin toimintaa tarkasti. Miten sen toiminta menomatkalla poikkeaa paluumatkasta?

Hyvät ohjelmointikäytännöt

Näissä harjoituksissa moottorin pyörimissuunta on tarvittaessa käännetty käyttämällä miinusmerkkiä nopeustiedon edessä. Pyörimissuuntaan voi kääntää myös laittamalla miinusmerkin moottorin pyörimistä asteina tai kokonaisina kierroksina määrittävän luvun eteen.

Tässä piilee kuitenkin pieni pulma. Ohjelmoitaessa tulee aina välillä vastaan tilanteita, joissa esimerkiksi robotin kulkema matka on kätevintä määritellä sekunteina, esim. 2.7s (huom. voit käyttää myös sekunnin osia). Mutta aika kulkee vain yhteen suuntaan, eteenpäin.

Pyörimissuunnan kääntäminen käyttämällä miinusmerkkistä nopeustietoa on osoittautunut yhdeksi monista hyvistä ohjelmointikäytännöistä. Yhteistä hyville ohjelmointikäytännöille on niiden taipumus auttaa meitä kirjoittamaan helpolukuisempia ohjelmia ja välttämään virheitä.

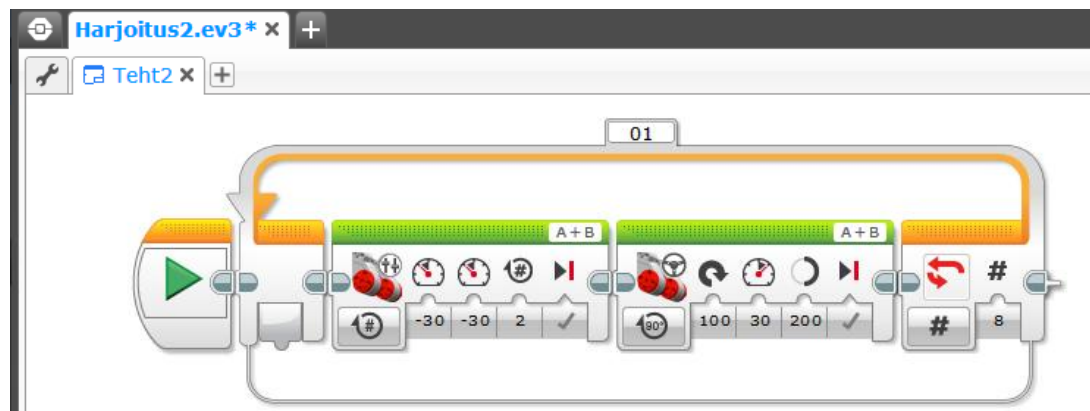
"Tiesitkö että kuvakkeiden käyttö ohjelmoinnissa on yksi tärkeimmistä hyvistä ohjelmointikäytännöistä."

Tehtävä2

Silmukka (Loop)

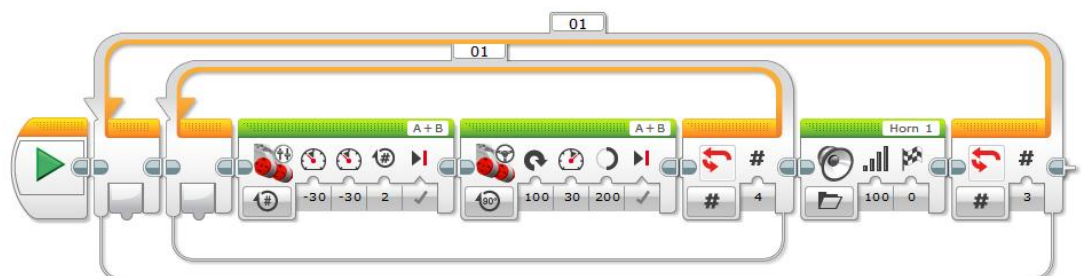
Ohjelmoitaessa kohdataan usein tilanteita, joissa jokin käsky, tai useamman käskyn muodostama kokonaisuus halutaan toistaa. Tällöin ohjelmaan lisätään silmukka. EV3:a ohjelmoitaessa toistettava ohjelmaosuus joko raahataan hiirellä, tai laaditaan suoraan silmukkamodulin sisään.

Ohjelmoi seuraavaksi robotti kulkemaan neliön (35cmx35cm) muotoinen reitti vastapäivään, kahdesti.



(kuva)

Ohjelmoi nyt robotti kulkemaan neliön (35cmx35cm) muotoinen reitti vastapäivään, kolmasti, mutta niin, että se antaa lisäksi äänimerkin jokaisen kierroksen päätteeksi.



(kuva) Vinkki: Horn 1 -äänien löydät klikkaamalla äänilohkon tyhjää nimikenttää ja valitsemalla soitettavien äänien valikosta "LEGO Sound Filles" → "Mechanical" ja sieltä "Horn 1" -tiedoston.

Tehtävä3

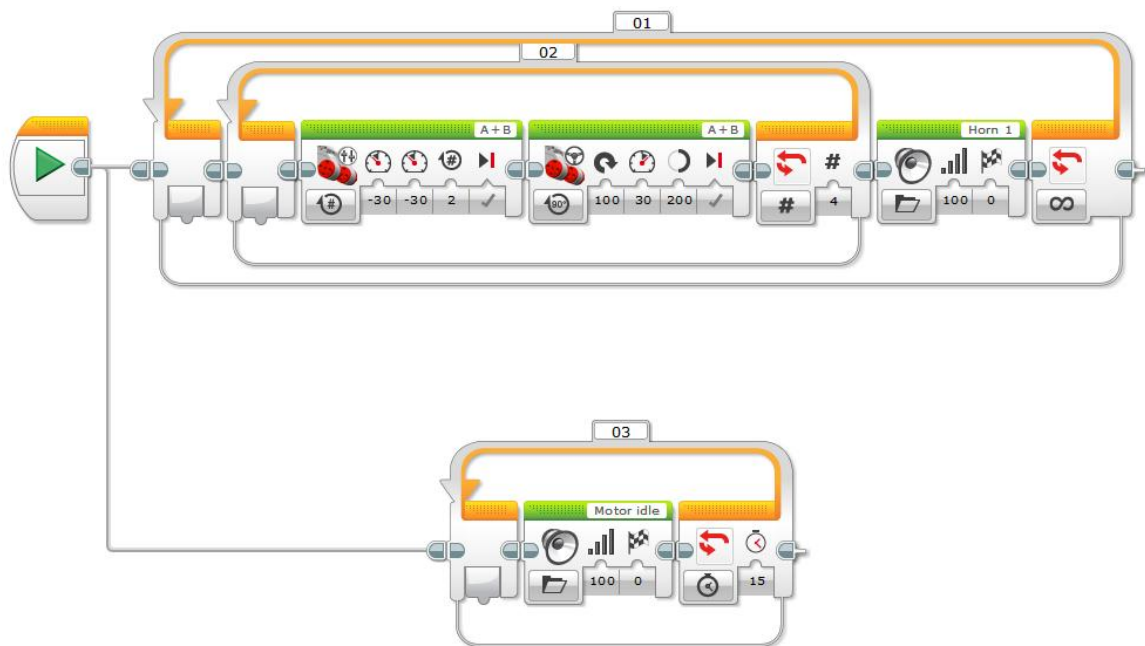
Rinnakkainen suoritus

Osa ohjelmoitavista laitteista kykenee suorittamaan useampia ohjelman säikeitä, eli haaroja, samanaikaisesti.

Nyt, vielä kun robottiimme on kytketty vasta kaksi moottoria, tälle toiminnolle on vähemmän käyttöä. Mutta viimeistään kilparobotteihin tutustuessamme tälle ominaisuudelle tulee käyttöä.

Rinnakkaiseen suoritukseen tutustutaan lisäämällä äsken tehtyyn ohjelmaan hiukan moottorin pärinää. Malli alla, tarkenna pärinä kestämään muutaman sekunnin verran ohjelman päähaaran suorittamista pidempään.

Kun teet ohjelmaan haaroitusta, huomaa että kaikki polut alkavat ohjelmamodulin portista ja päättyvät ohjelmamodulin porttiin.



(kuva) Vinkki: Motor idle -äänien löydät klikkaamalla äänilohkon tyhjää nimikenttää ja valitsemalla soitettavien äänien valikosta "LEGO Sound Filies" → "Mechanical" ja sieltä "Motor idle" -tiedoston.

Lisätehtävä 1

Keksikää robotillenne hauskoja tanssiliikkeitä. Miettikää samalla miten silmukoita voisi hyödyntää koreokrafian ohjelmoinnissa. Lopuksi voidaan järjestää vielä pienet robottien tanssiaiset.

Jatkot

Etsi esimerkiksi Wikipediasta lisää tietoa hyvistä ohjelmointikäytännöistä.

Seuraavalla harjoituskerralla rakennetaan robotti, joka osaa seurata viivaa. Viivan seuraamiseen voit tutustua jo ennakolta vaikka katsomalla netistä videoita hakulauseella "EV3 line follower".

Vastaukset, Tehtävä1

Menomatkalla robotti pysähtyy joka kerta suoritettuaan kahden moottorin toimilohkolla määritellyn tehtävän. Paluumatkalla robotti kulkee pysähtelemättä, jouhevasti. Näitä molempia ominaisuuksia voidaan hyödyntää robotin ohjelmoinnissa, tietysti tilanteesta riippuen.

3. Valokenno, paikoitus ja viivan seuraaminen, If-Then

Etkot

Viivan seuraamisen harjoitteluun tarvitaan harjoitusalue.

- o Jos käytettävissä on vaalea sileä lattiapinta, harjoitusradan voi tehdä käyttäen mustaa teippiä.
- o Jos käytettävissä on iso tulostin, harjoitusradan voi tehdä tulostamalla lopussa liitteenä olevan radan paperille.
- o Vastaavan voi tehdä ilman tulostintakin maalaamalla, piirtämällä, teippaamalla mille tahansa sileälle vaalealle alustalle (levy / kartonki / paperi / mainoskangas / jne.).

<https://www.youtube.com/watch?v=ODAGVeeDagk>

<https://www.youtube.com/watch?v=098Mzecwvkl>

<https://www.youtube.com/watch?v=YLCkjcWG8yw>

Intro

Viivan seuranta käytetään jo nyt monessa paikassa, kuten vihivaunuissa eli tavaroita kuljettavissa roboteissa. Tätä kirjaa kirjoittaessa ovat tälle rintamalle tulleet juuri mukaan mm. maantien reunamerkit ja tunnistavat autojen turvajärjestelmät ja erilaisia robottiautoja.

Valon ja värin mittaaminen, tekniikka

EV3:n valokennossa on yhdistetty kolme erilaista toimintoa.

- o Valon määrän mittaus
- o Heijastuvan valon mittaus
- o Värin tunnistus

Tämä valokenno on varustettu valoa vastaanottavan kennon lisäksi vihreällä, punisella ja sinisellä ledillä, joita käytetään värien tunnistuksessa ja apuvalona heijastuvan valon määrää mitattaessa.

Tunnistettavat värit ovat musta, sininen, vihreä, keltainen, punainen, valkoinen ja ruskea. Muut värit tulkitaan luokkaan "No Color", tai lähimpään edellä mainituista. Tarkista tarvittaessa miten kenno tulkitsee värin. Esimerkiksi oranssi saattaa tulla tulkituksi joko keltaiseksi tai punaiseksi, ruskeaksi, jopa mustaksi, sävystä ja etäisyydestä riippuen.

Tehtävä1

Liitetään valokenno robottiin (Portti 1) ja tehdään apuohjelma, joka näyttää ohjaimen näyttöruudulla mitä valokenno "näkee". Mitataan millä etäisyydellä EV3:n valokennon erottelukyky on parhaimmillaan ja kokeillaan eri värien tulkintaa.

Tehdään ohjelma, joka

- o Tyhjentää näytön ja kirjoittaa ensimmäiselle riville "VALOKENNO 1",
- o toiselle riville "Vari" + tätä vastaavan mittaustiedon,
- o kolmannelle riville "Valo" + tätä vastaavan mittaustiedon,
- o ja neljännelle riville "Heijast." + tätä vastaavan mittaustiedon.
- o Mittauksesta siirrytään aina seuraavaan 0.5s välein, päättymätön silmukka.



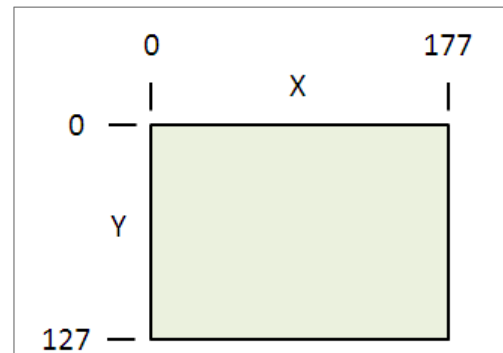
(kuva)

Näyttöruutu

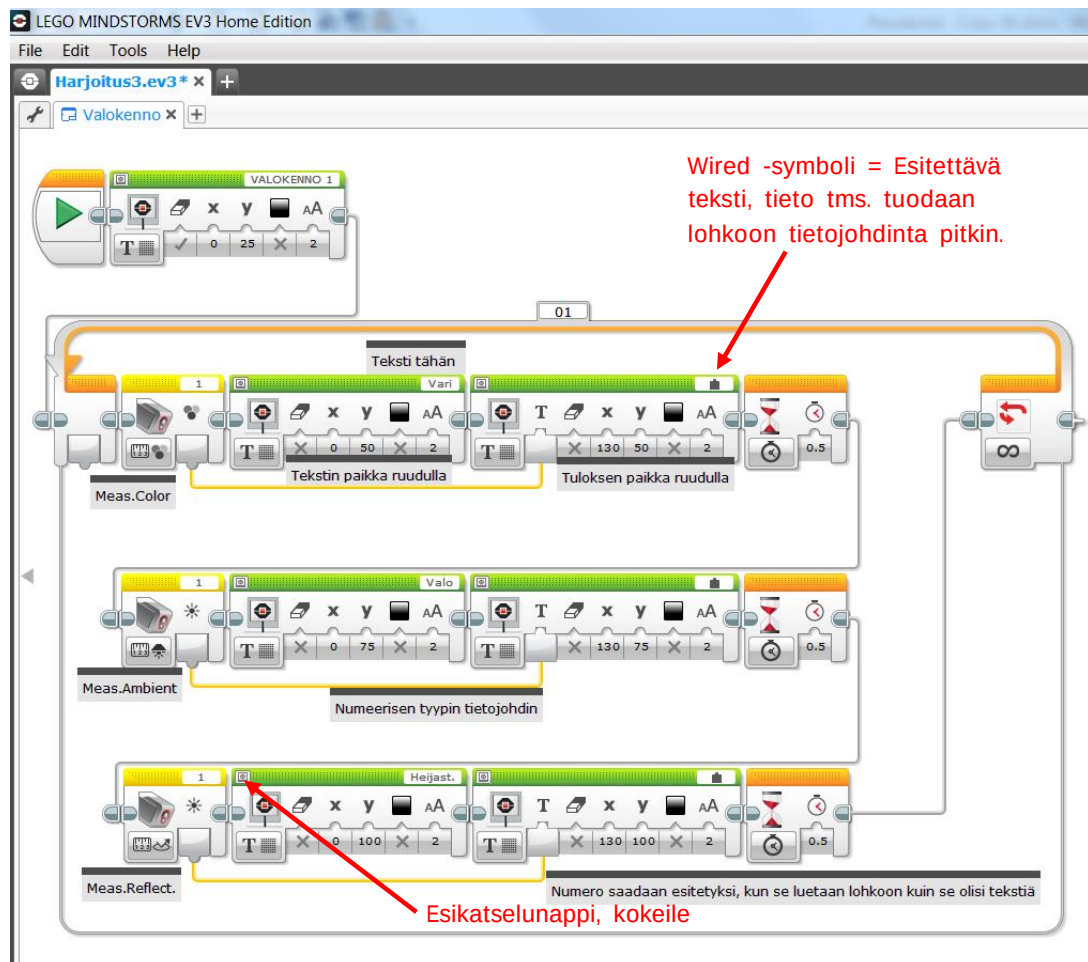
EV3:n näyttöruutu on 177x127 pisteen kokoinen. Tekstin aloituskohta, eli sen ensimmäisen merkin vasemman yläkulman paikka määritellään tämän vaaka- (X) ja pysty- (Y) suuntaisena etäisyytenä näyttöruudun vasemmasta ylänurkasta.

Edellisen sivun kuvan esimerkissä rivit (Y) oli määriteltä alkamaan korkeudelta 25, 50, 75 ja 100 pistettä.

Kaikki tekstit alkavat ruudun vasemmas- ta reunasta, eli kaikkien tekstien X parametrin arvo =0. Mittaustietojen sa- rake alkaa vastaavasti kohdasta X=130.



Kirjoitetaan ohjelma mallin mukaan.



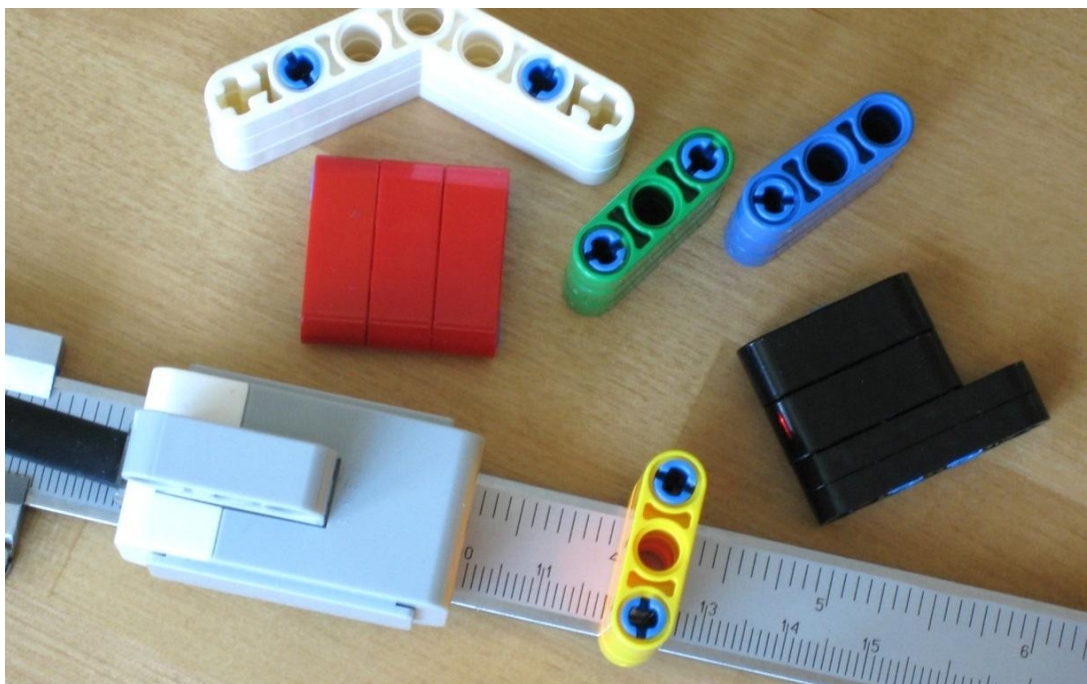
(kuva) Vinkki: Tee ensin tarkasti yksi rivi. Kopioi se kolmeksi ja ketjuta nämä. Siirrä aikaansaannos silmukkalohkon sisään ja viimeistele ohjelma.

Kun ohjelma on saatu toimimaan, voidaan ryhtyä selvittämään valokennon ominaisuuksia kokeellisesti.

Tehtävään tarvitaan viivotin mittaamiseen ja vihko tuloksien kirjaamista varten. Robottisarjan toistaiseksi vapaana olevista sinisistä, punaisista, vihreistä, keltaisista, valkoisista ja mustista osista saa rakennettua kätevät, kolme nuppia leveät mittapalikat (6/7 väriä, ruskea jää puuttumaan).

Harjoitus3, Tehtävä 1a)

- ❖ Kirjaa vihkoon sivun vasempaan reunaan alekkain numerot 0, 1, 2, 3, 4, 5, 6, ja 7.
- ❖ Selvitä mittapalikoiden avulla mitä väriä kukin numero vastaa, kirjoita värit numeroiden perään.
- ❖ Kokeile ja mittaa mikä on lyhin ja pisin etäisyys jolloin kenno vielä tunnistaa kunkin värin oikein. Kiinnitä huomiota myös kennon ja palikoiden asentoihin. Kirjaa tulokset vihkoon.
- ❖ Keskustelkaa tuloksista.
 - ? Oliko värien kesken eroja
 - ? Saivatko kaikki samansuuntaisia tuloksia
 - ? Mitä värejä mahtaisivat edustaa kaksi tyhjäksi jäänyttä riviä



(kuva) Vinkki: Kennon ja mittapalikan voi asettaa viivottimen päälle.

Harjoitus3, Tehtävä 1b)

Tutkitaan kohteesta heijastuvan valon määrän mittaamista.

- ❖ Apuvälineiksi tarvitaan viivotin, sekä mattapintainen paperiarkki, harjoitusalue tms. jolle on tulostettu tai tussilla maalattu, vähintään 3cm leveä, mahdollisimman musta viiva.
- ❖ Kokeile ja mittaa, millä etäisyydellä ja missä asennossa (tarkasteltavaan pintaan nähden) valokennon on oltava, jotta se näkisi jyrkimmän rajan, suurimman mahdollisen eron mustan ja valkoisen välillä. Kirjaa havainnot vihkoon.
- ❖ Keskustelkaa tuloksista.
 - ? Oliko valokennon asennolla väliä
 - ? Paras tunnistusetäisyys
 - ? Saivatko kaikki samansuuntaisia tuloksia

Harjoitus3, Tehtävä 1c)

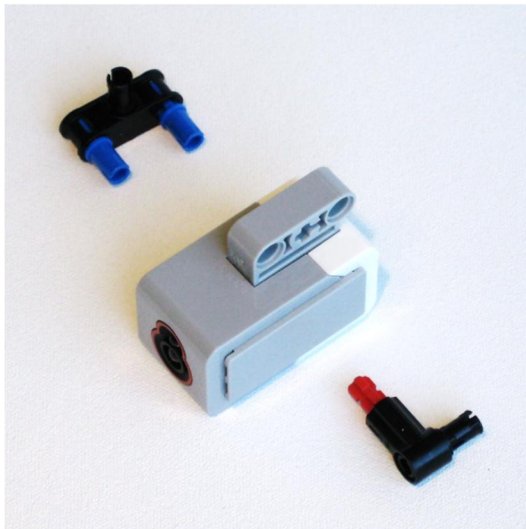
Mitataan valoisuutta, muista valon lähteistä peräisin olevan valon määrää.

- ❖ Kokeilkaa millaisia lukemia saatte ikkunasta tulevasta valosta, tietokoneen näytön valosta, jne.
- ❖ Opettajan / ohjaajan luvalla, voitte kilvoitella etsien työskentely tilanne valoisimman ja pimeimmän kohdan.
- ❖ Sääntöinä on, että mittausta suoritettaessa valokennon edessä pitää olla 30cm vapaata tilaa (viivottimen pituuden verran). Tavaroiden ja ihmisten liikuttelua tulee välttää tätä koetta tehtäessä. Etenkin, jos siitä voi seurata havaittavissa oleva muutos mittauksen tulokseen.

Nyt on tutkittu valokennon toimintaa. Mettikää yhdessä miten valokennoa voisi hyödyntää robotin toiminnassa.

Tehtävä2

Seuraavaksi valokenno kiinnitetään osaksi robottia. Lisätään aluksi sopivat kiinnikkeet kennoon.

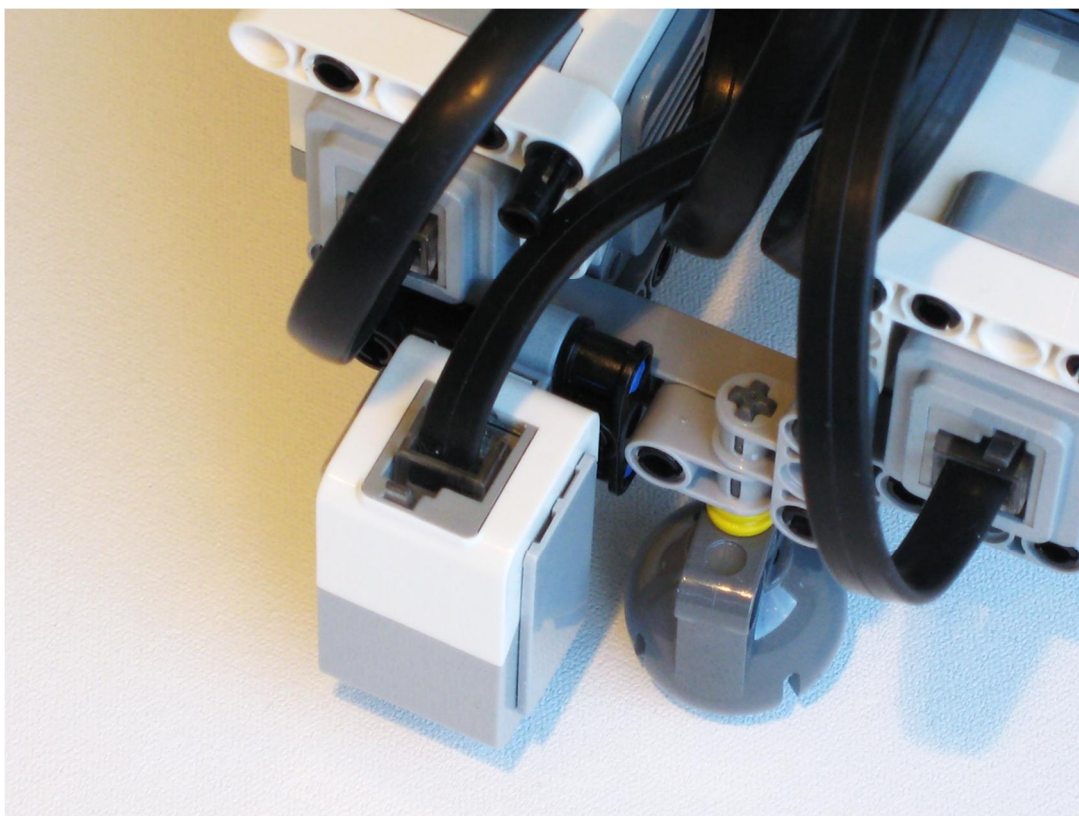


(kuva)



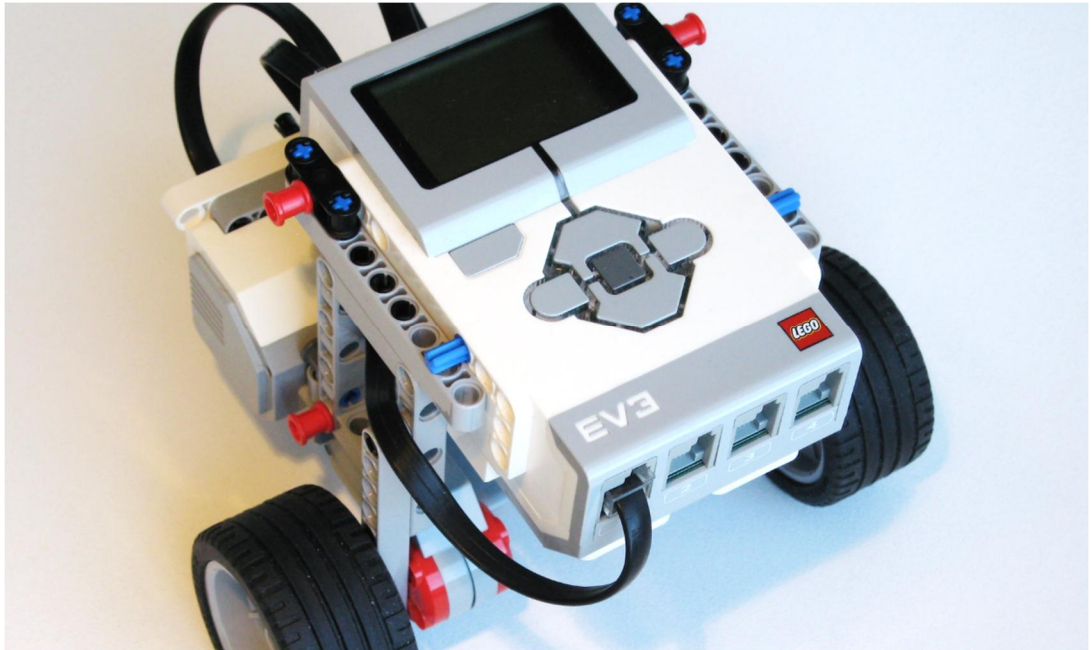
(kuva)

Valokenno tulee robotin pätyyn, Caster Ball:in viereen.



(kuva)

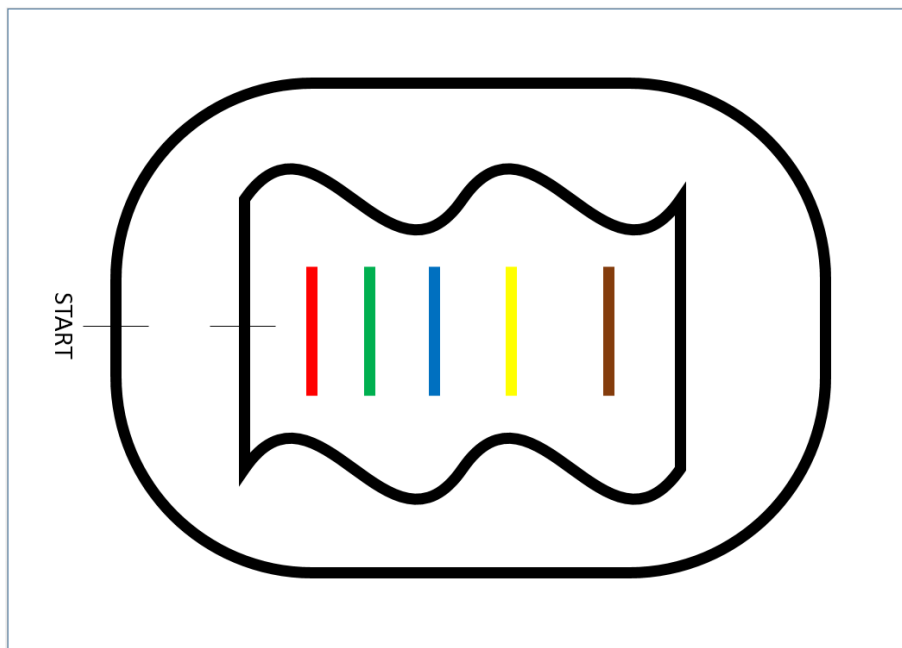
Kytetään valokenno vielä takaisin tulo-porttiin numero 1.



(kuva)

Tarkista, että valokennon alle jää nyt sopivasti tilaa (vertaa etäisyyttä edellisten mittauksien tuloksiin).

Seuraavaksi rakennetaan testirata käyttäen mustaa ja värillisiä PVC-teippejä (tunnetaan myös sähkömiehen teippinä, jota saa ostaa mm. kaikista rautakaupoista), tai käytetään soveltuvaa valmista pohjaa. Alla olevan kuvan mukainen testirata löytyy tämän kirjan liitteistä tulostettavana .pdf -tiedostona (100cm x 140cm, edellyttää suurkuvatulostusta, tai mosaiikkitulostusta).



(kuva)

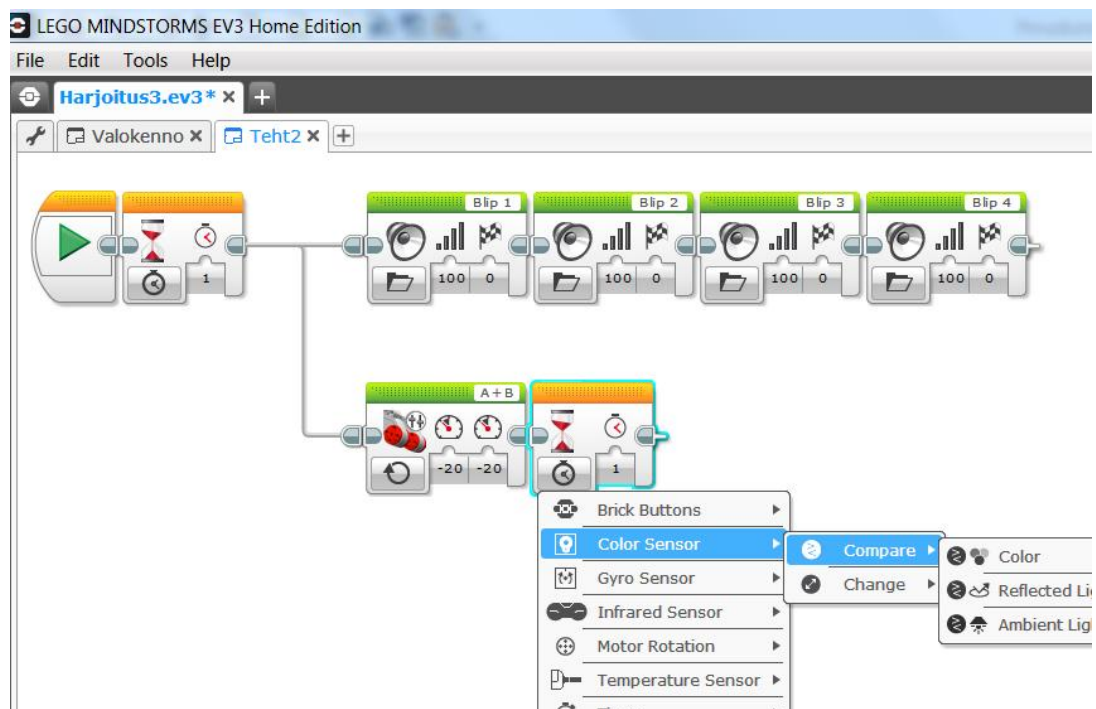
Paikoitus

Ohjelmoidaan robotti kulkemaan "Start" -tekstin luota suoraan siniselle viivalle asti ja peruuttamaan sieltä sitten takaisin lähtöpaikkaansa. Suoritusta voidaan havainnollistaa vapaavalintaisten äänimerkkien avulla.

Ohjelmointia voidaan aina tehdä lukuisilla ei tavoilla. Tässäkään tehtävässä ei ole olemassa yhtä oikeaa ratkaisua. Nyt on hyvä tilaisuus tehdä ihan oma ohjelma.

Viivelohko on hyvin monikäyttöinen. Se toimii kuin portilla seisova vartija. Matka voi jatkua vasta kun vartija on tarkastanut jonkin asian ja ehdot ovat kunnossa.

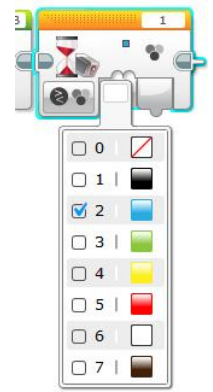
Viivelohko voi esimerkiksi pidätellä ohjelman suoritusta ohjelmoijan määrittelemän ajan verran, kuten alla olevassa kuvassa pienen aloitusviiveen muodossa. Ehdoksi voidaan määritellä ajan sijaan vaikka sellainen, että valokennon pitää kertoa tunnistaneensa jotakin sinistä.



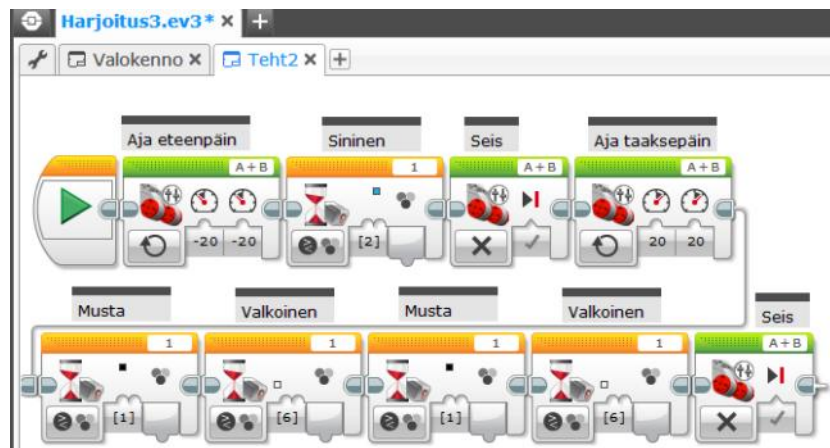
(kuva) Valitaan viivelohkolle oikea moodi, toiminta tapa. Se kertoo viivelohkolle, että haluamme sen lukevan valokennon mittaamaa tietoa (Color Sensor), ja vertaavan (Compare) väriä.

Lisäksi meidän pitää kertoa viivelohkolle, mitä väriä haluamme sen etsivän. Tässä tapauksessa etsimme sinistä väriä. Samalla kannattaa huomata, että samanaikaisesti tarkasteltaviksi väreiksi voidaan valita useampia kuin yksi (ruksi siniseen ja ruksi pois punaisesta).

Tärkeää: Tarkasta myös tuloportin tunnus, valokenno on kytketty porttiin numero 1.

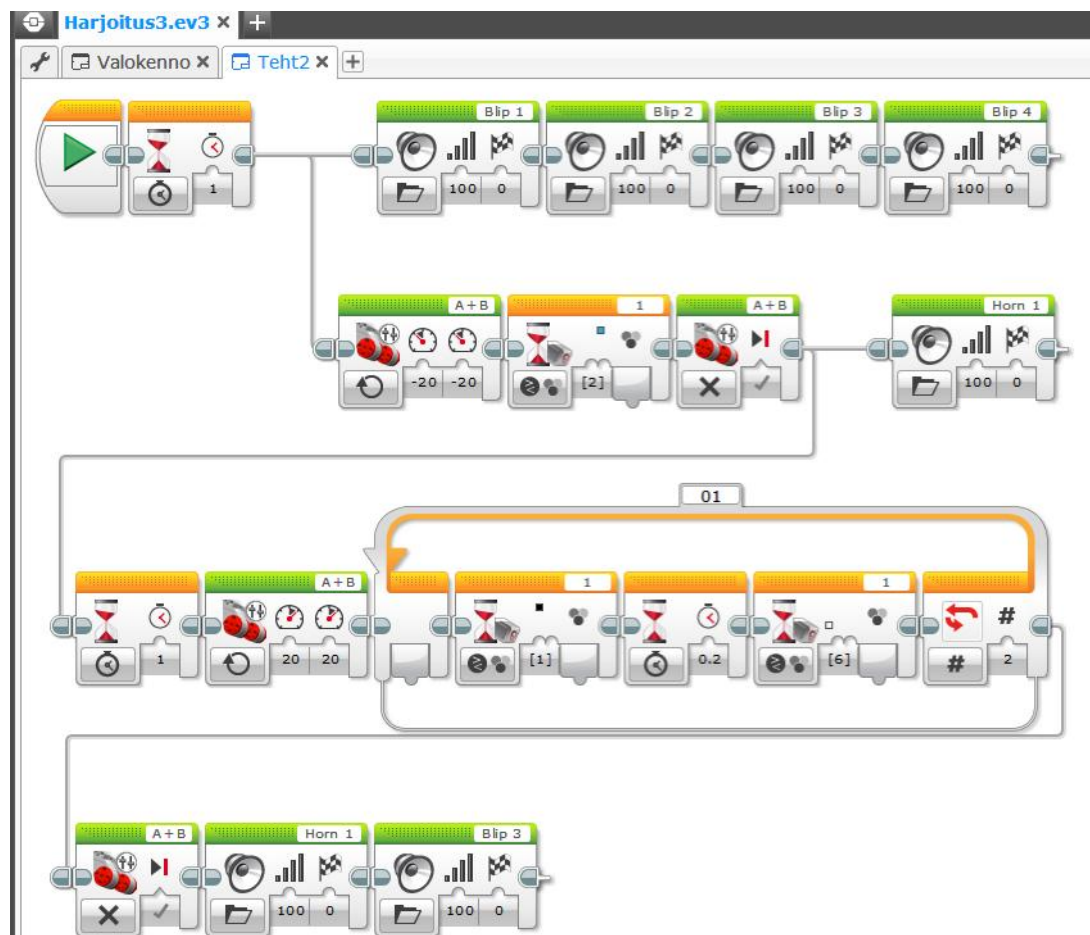


Yksinkertaistettu ratkaisu, kommentti kenttien käyttäminen helpottaa ohjelman lukemista merkittävästi, etenkin kun tehdään isompia ohjelmia.



(kuva)

Pidemmälle viety ratkaisu. Ajatus on molemmissa kuitenkin sama.



(kuva)

Tehtävä3, Viivanseuranta

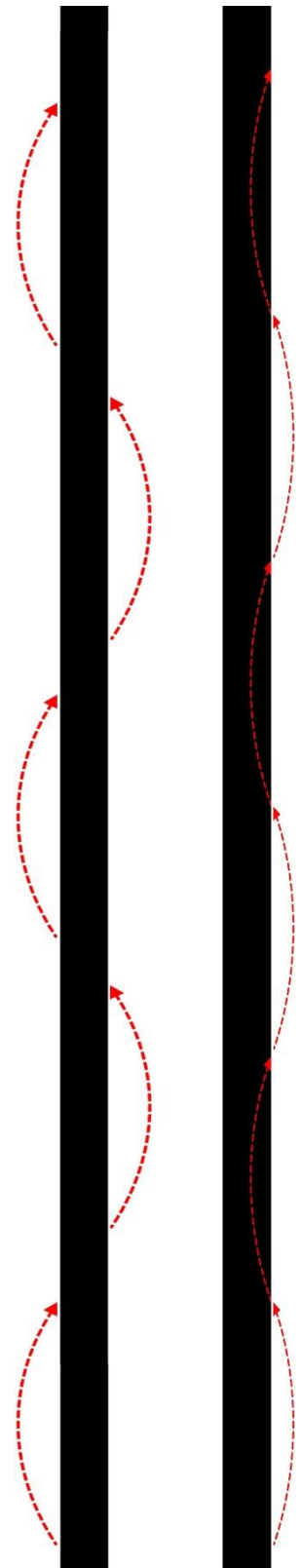
Viivanseurannassa robotti koittaa joko kulkea mustan viivan päällä, tai jompaa kumpaa sen reunaa pitkin. Käytännössä, molemmissa edellä mainituissa tapauksissa, robotti myös kaartaa koko ajan loivasti joko oikealle, tai vasemmalle.

Mustalla kuljettaessa periaatteena on, että kun valokenno näkee valkoista, robotti lähtee kaartamaan siihen suuntaan missä viimeksi oli mustaa. Eli valkoisesta seuraa aina kaartosuunnan vaihdos. Huom. ennen seuraavaa kurssin vaihdosta robotin pitää käydä selkeästi mustalla.

Selkeästi suositumpi tapa seurata viivaa on ollut viivan reunan seuraaminen. Tähän on parikin syytä. Tehtävään voi liittyä viivojen risteys, joista pitää kääntyä tiettyyn suuntaan. Robotti saadaan kääntymään haluttuun suuntaan helposti seuraamalla tämän puoleista viivan reunaa. Toinen liittyy tehtäviin joissa robotin pitää viivaa pitkin kuljettuaan osua hyvin tarkasti johonkin kohteeseen. Reunaa seuraamalla robotti kiemurtelee käytännössä vähemmän ja näin sen osumatarkuus paranee.

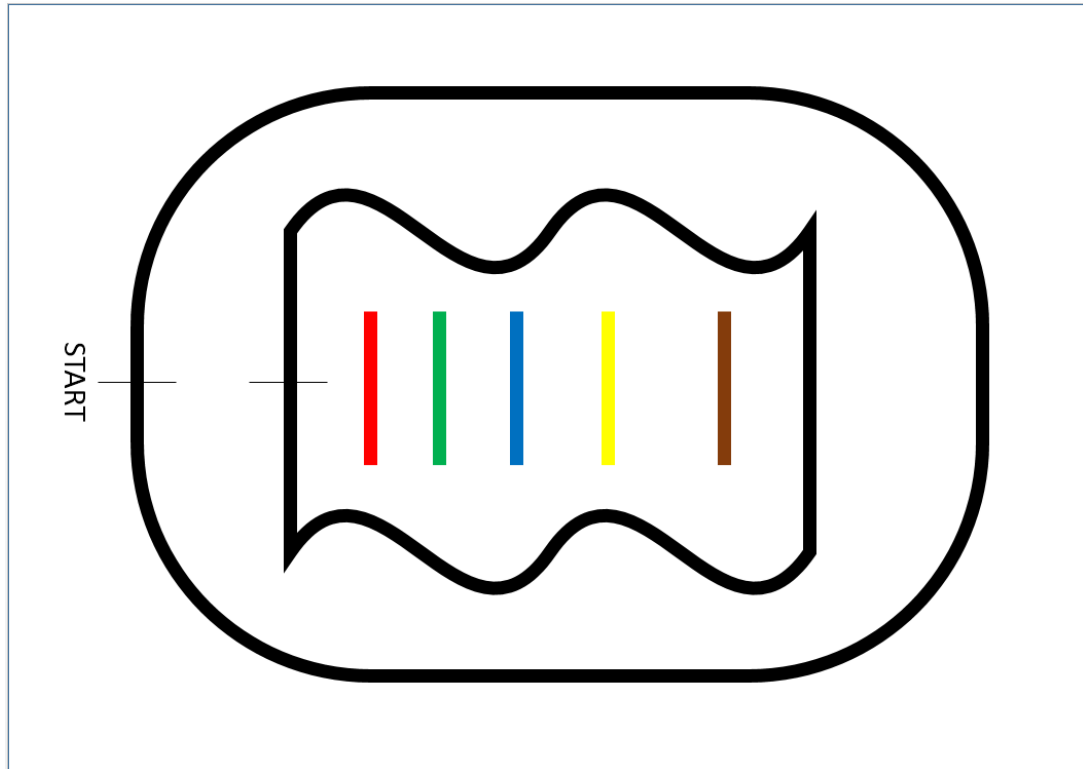
Robotissamme on kaksi vetävää pyörää ja oma moottori molemmille. Aina kun moottorit pyörivät samaan suuntaan yhtä lujaa, robotti kulkee suoraan. Vastaavasti se kääntyy aina kun ne pyörivät hiukankin eri nopeuksilla.

Lähdemme seuraavaksi tekemään viivan oikeaa reunaa seuraavaa ohjelmaa.



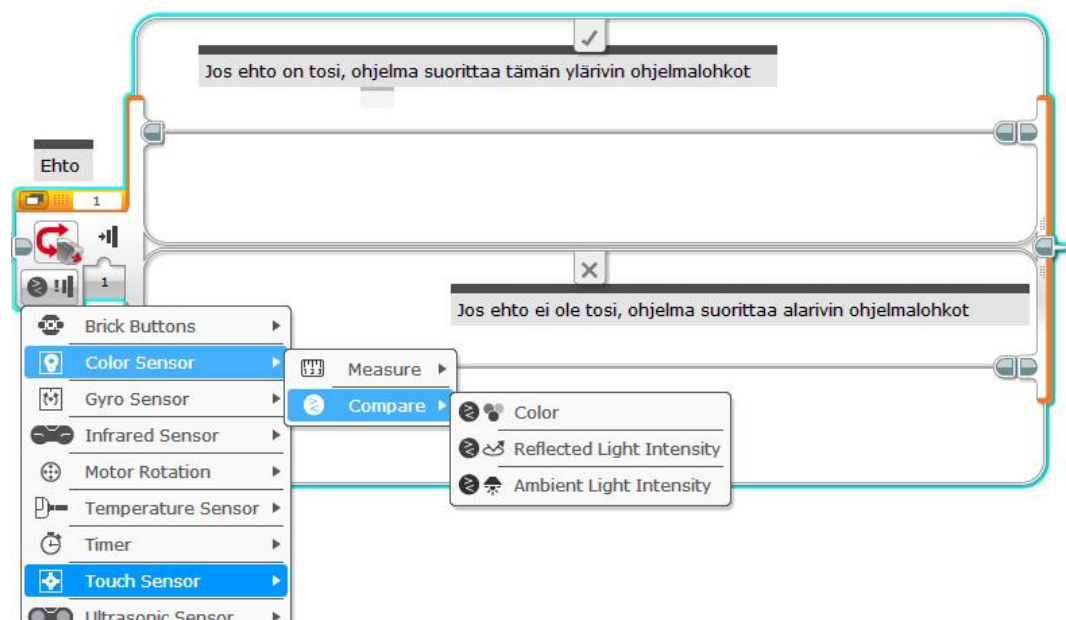
(kuva)

Tavoittena on ohjelmoida robotti kulkemaan alustan ulointa rataa jouhevasti, molempiin suuntiin. Robotti seuraa viivaa helpommin pienemmällä nopeudella. Vauhtia kannattaa lisätä vasta kun kierros sujuu ongelmitta.



(kuva)

Ehdollinen suoritus, If-Then -lohko



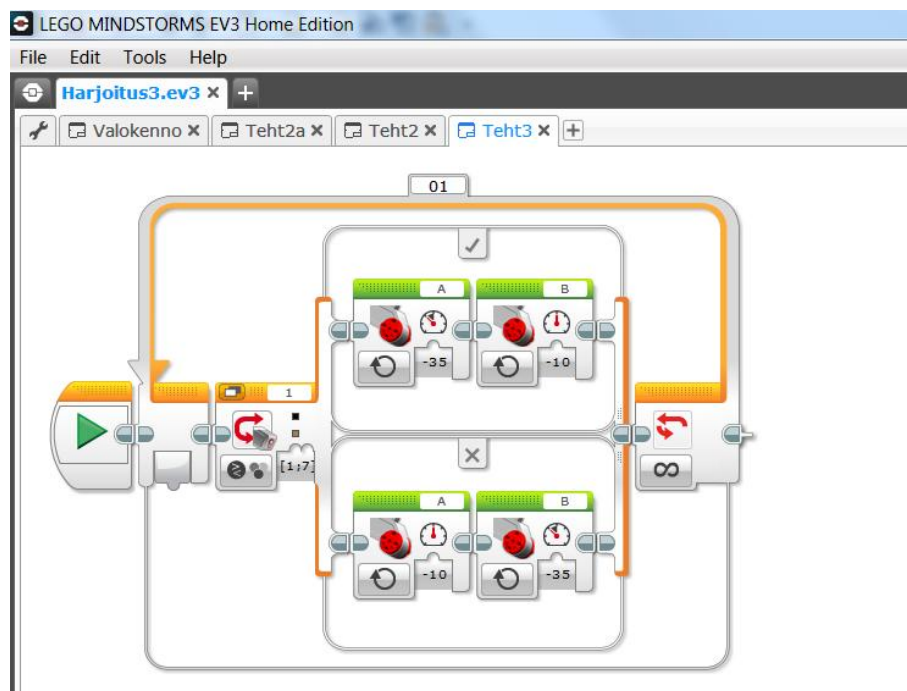
(kuva)

Perusmallin ohjelma viivan seuraamiseen on helppo tehdä käyttäen ehdollista ohjelman suoritusta.

Jos (If) valokenno näkee mustaa (tai ruskeaa),

- ➔ **Silloin (Then)** moottori A:n tulee pyöriä B:tä nopeammin (✓).
- ➔ **Muutoin (Else)** moottori B:n tulee pyöriä A:ta nopeammin (✗).

Siirretään ehdollinen lohko lopuksi päättymättömän silmukan sisään ja ohjelma on valmis.



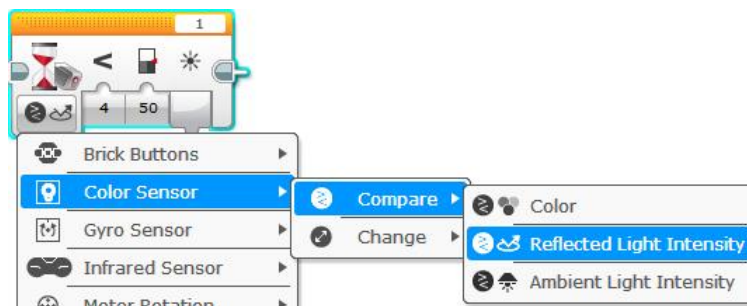
(kuva)

Ohjelman hienosäätö tehdään muuttamalla moottorilohkojen nopeusasetuksia. Mitä pienempi moottorien nopeusero, sitä loivemmin robotti kääntyy. Etsikää oikeat arvot kokeilemalla.

Lisätehtävä1

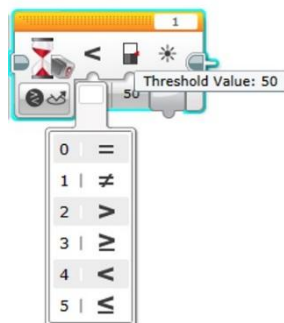
Tehtävässä 1 robotti kävi sinisellä viivalla. Kokeilkaa onnistuteko tunnistamaan robotilla kaikki värilliset viivat. Ongelmatilanteissa kannattaa käynnistää valokennon testausohjelma, viedä valokenno kyseisen viivan päälle ja katsoa mitä kenno oikeasti näkee.

Jos värin tunnistamisessa on vaikeuksia, voi pysähtymistä kokeilla myös heijastuvan valon määrän perusteella.



(kuva)

Tämä lohko odottaa kunnes heijastuneen valon määrä on laskenut alle 50% rajan



(kuva)

Lisätehtävä2

Paranneltu viivanseuranta

Seurataan mutkittelevaa rataa, kaarevuussäde min 50mm

Seurataan mutkittelevaa rataa, myös aivan teräviä, yli 90 asteen käännöksiä

Jatkot

Vastaukset

Harjoitus3, Tehtävä1a)

nro	väri	min	max
0	Ei väriä (ei kohdetta)		
1	Musta	4	17
2	Sininen	5	15
3	Vihreä	3	18
4	Keltainen	1	18
5	Punainen	2	15
6	Valkoinen	2	21
7	Ruskea	-	-

Harjoitus3, Tehtävä1b)

Valokenno tunnistaa parhaiten mustan viivan reunan, kun se on asennettu kohtisuoraan tarkasteltavaa pintaa vastaan ja n.8mm etäisyydelle siitä.

Näihin mittauksiin, sekä valokennon toimintaan yleisesti liittyviä virhelähteitä:

- 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 kykenee lukemaan valokennon mittaustiedon 1000 kertaa sekunnissa. Mutta riippuen siitä, miten kennoa luetaan ohjelmassa, isokin kohde saattaa jäädä huomaamatta.

(Täytetekstejä ja kuvia)

Otsikoita mediassa:

Robotit ja robottien moraalinen vastuu Yle ti 21.4.2015

Stephen Hawking näkee tulevaisuuteen: robotit ottavat vallan Tivi, 16.5.2015

Robotit eivät vie työtä ihmisiltä, mutta työ muuttuu Elinkeinoelämän keskusliitto 29.4.2015

Googlen robotit pian entistä lähempänä sydäntä 28.3.2015

Robotti-Audi pääsi maaliin – terässilta pelästytti Digitoday 3.4.2015

Google: Myös robotti ruttaa peltiä liikenteessä It-viikko 12.5.2015

Tutkijat siirsivät madon aivot Lego-robotille - tulos hämmästyttää 04.02.2015 MTV3

Robotti pesee, ruokkii ja lypsää Etelä-Suomen Sanomat 16.4.2015

Robottikoira osaa jo hyppiä esteiden ylitse – tutkijat kuvasivat testijuoksun videolle HS 30.5.2015

[Ihmistä muistuttava robotti lähetetään syyskuussa avaruuteen](#) Tähdet ja avaruus 21.04.2010

Voiko robotti olla vastuussa? STT 16.3.2015

Robotti, ihmisen paras ystävä? Tekniikan Maailma 06.03.2015

Robotit ruokkivat luovuutta Opetin 28.4.2015

Robotti tutki tehdasalueen räjähdekonttia Keskisuomalainen 10.7.2013

2015-04-15 - ABB on esitellyt maailman ensimmäisen ihmisen kanssa yhteistyöhön kykenevän kaksikäätisen robotin nimeltään Yumi®.

Robotti tottelee yhdeksänvuotiasta HSTV 17.6.2014

Kuka korvaa, jos robottiauto kolaroi? – Yle: Vakuutusyhtiöillä pohdinnan paikka Helsingin Uutiset 10.3.2015

Outouden laakso näkyy robottien päätösten hyväksymisessä Turun Sanomat 16.3.2015

Kolmas teollinen vallankumous on robottien Liikenne- ja viestintäministeriö 25.05.2015

Robotti korvaa toimittajaa Mediaviikko 18.01.2013

Tätä on tulevaisuus? Spot-robotti laukkaa kuin oikea koira Stara.fi 17.5.2015

Trafi: Robottiauto voi lähteä Suomessa liikenteeseen Tuulilasi 24.02.2015

Robotti ryhtyy remontoimaan viemäreitä Töölön alla Metro 19.1.2015

Robotti tanssii tai tulkkaa viittomia Hämeen sanomat 18.04.2015

KUINKA ROBOTTI RANGASTAAN? Kuka on vastuussa, jos hoivarobotti tekee kohtalokkaan virheen? Mitä enemmän kone muistuttaa ihmistä, sitä vastuullisempaan sitä tunnutaan pidettävän. Ja kun kone inhimillistetään, sen syyksi saatetaan laskea erilaisia virheitä – koneita ohjelmoivien ihmisten sijaan. Helsingin Yliopisto 23.3.2015

Robotti lypsää lehmät Moksissa Vihdin sanomat 14.1.2015

Avant Robot 185 –purkurobotti koneporssi.com 21.01.2014

Satakunta, eli robottirannikko Satakunnan kansa 8.3.2015

Robotti vartijan kollegaksi? G4S Suomi 27 helmikuu 2015

"Miettikää! – Robotti joka oppii välttämään kipua, siis toi on periaatteessa älykkäämpi kuin yksikään Duudsoneista!" Noin viikon uutiset 5.3.2015: Robotit tulevat

Fukushima Robot Dies Three Hours After Entering Radioactive Reactor Vessel [sputniknews.com /20150411/](http://sputniknews.com/20150411/)

Uusi robotti kykenee tekemään aivoleikkauksen – Kärpäselle Tekniikka&Talous 28.5.2015

KYS: Robotti aloittaa antibioottiannosten massavalmistuksen 29.5.2015

Robotti ryhtyy palomieheksi Tekniikka&Talous, 5.2.2015

Voiko robotti rikkoa lakia pelastaakseen ihmisiä? Tekniikka&Talous, 23.3.2015

Tieteiskirjailija Isaac Asimov kehittämät robotiikan kolme pääsääntöä:

1. Robotti ei saa vahingoittaa ihmistä eikä laiminlyönnin johdosta saattaa tätä vahingoittumaan.

2. Robotin on toteltava ihmisen sille antamia määräyksiä paitsi milloin ne ovat ristiriidassa ensimmäisen pääsäännön kanssa.

3. Robotin on varjeltava omaa olemassaoloaan niin kauan kuin tällainen varjeleminen ei ole ristiriidassa ensimmäisen eikä toisen pääsäännön kanssa.

Myöhemmin Asimov lisäsi nollannen pääsäännön: "0. Robotin tulee suojella ihmiskuntaa." Tällöin muihin pääsääntöihin lisätään määreet "mikäli se ei riko edellisiä.

Robotti työkaverina Atrialla Telma 15.05.2015

Robotti hoitajan kaveriksi? Kaleva 3.3.2015

Koira otti osaa kevätsiivoukseen: Kulkupelinä robotti-imuri The Voice Linda Arponen | 05.04.2015

Palkataan rehellinen ja reipas CRM-robotti Blogi 6.4.2015 Jari Parantainen

Robottien tulo pakottaa meidät pois lepsusta relativismista HS 5.5.2015

Keskustan salainen vaaliase: Robotti-Sipilä! Ilta-lehti 7.2.2015

Pientä suloista ja vaaleanpunaista, meillä imuroi Potti Blogi 27. helmikuuta 2015

Robotti opettelee laittamaa ruokaa katsomalla YouTube-videoita Aamulehti 15.01.2015

Robotti tarkastaa generaattorin kunnon ProMaint 09.3.2015

Tulevaisuudessa sähkölinjoja voi tarkastaa robotti Karjalainen 24.09.2014

Jäätikön alle laskettu robotti löysi eläviä kaloja hilavitkutin.com 27.1.2015

Husiin tulee maailman suurin täysautomaattinen lääkevarastojärjestelmä - 35 metriä pitkä, 7 robottikättä Mediuutiset 15.4.2015

Robottien käytön arvoperusteet selvitettävä Valomerkki 19.01.2015

Kiuruveteläinen lypsyrobotti yllätti tilalliset yli tonnin puhelinlaskulla. Iisalmen sanomat 19.5.2015

Eteläkorealainen ihmismäinen robotti vei miljoonien eurojen potin – paras katastrofityöhön HS 7.6.2015