



SUOMEN HARJOITTELUKOULUJEN OHJELMOINTIPOLKU

17.12.2015

Työryhmä

Sampo Forsström
Mikko Horila
Anja Kareinen
Anna-Maija Laukkarinen
Tero Luodeslampi
Ilpo Ollikainen
Päiviö Peltokorpi
Tapani Saarinen

eNORSSIin ohjelmointipolku

Perusopetuksen opetussuunnitelmien perusteiden 2016 mukaisesti ohjelmointi tulee uutena sisältönä ja opiskeltavana taitona alakouluun, ja sen opiskelu tulee kaikille pakolliseksi yläkoulussa. Koska ohjelmointi tulee erityisesti alakouluihin uutena sisältönä, pyrkii eNorssi osaltaan helpottamaan koulujen tehtävää ohjelmoinnin opetussuunnitelman laatimisessa. eNorssin ohjelmointipolun tavoitteena on jäsentää ohjelmoinnin roolia opetussuunnitelmassa sekä eri vuosiluokkien opetuksessa.

Dokumentin alkuun on koottu taulukko, joka antaa kuvan mahdollisesta etenemismallista ohjelmoinnin opetuksessa. Myöhemmillä sivuilla on pyritty avaamaan toteutusmahdollisuuksia tarkemmin alakoulun osalta luokkatasopareittain ja yläkoulun osalta kokonaisuutena. Dokumentin loppupuolella on puolestaan koottu valtakunnallisten opetussuunnitelmien perusteiden viittaukset ohjelmointiin. Tavoitteena on, että tämän dokumentin luettuaan opettaja saa kuvan siitä, mitä perusopetuksen opetussuunnitelman perusteissa todetaan ohjelmoinnista ja miten sen pohjalta voidaan suunnitella ja toteuttaa opetusta eri luokkatasoilla aina alkuopetuksesta yläkouluun saakka.

Ohjelmoinnin opetuksen ajatuksena on erityisesti ongelmanratkaisutaitojen sekä loogisen ajattelun kehittäminen, mutta ohjelmoinnin opetus tukee varmasti myös hahmottamisen kehittymistä, taitoa pilkkoa asiat pienempiin osiin sekä sosiaalisten taitojen kehittymistä. Luonteeltaan ohjelmointi sopii mainiosti myös pari- ja ryhmätehtäviin ja toimii myös eriyttävänä osa-alueena mm. matematiikassa.

Ohjelmointipolun tavoitteena on tarjota vinkkejä erityisesti ilmaisiin ja matalan kynnyksen palveluihin, mutta joukossa on myös (mm. robotiikka) toimintamalleja, jotka vaativat erityisiä laitteistoja, sarjoja tai pelejä. Lähes kaikkeen toimintaan löytyy myös ilmainen vaihtoehto, mutta ymmärrettävästi pienellä taloudellisella panostuksella toimintaan saadaan lisää vaihtoehtoja. Dokumentin *Kursivoidulla kirjoitetut* palvelut ja työtavat ovat ko. luokka-asteille mahdollisia, mutta eivät suinkaan automaattisia lähtökohtia. Dokumenttia tulee muutenkin lukea ja toteuttaa valikoiden, ajatuksena ei ole edetä automaattisesti kaikkia sisältöjä ylhäältä alas vaan valita koululle toimivimmat osa-alueet. Erityisesti yläkoulun osalta koulujen toteutustapaerot lienevät suuria valittuun ohjelmointikieleenkin perustuen.

eNorssin ohjelmointipolkua on ollut suunnittelemassa ja toteuttamassa lähes kaikkien norssien opettajia. Taustamateriaalina on hyödynnetty eri toimijoiden verkossa julkaisemaa materiaalia, joten käsillä olevasta dokumentista löytyy varmasti samankaltaisuuksia muiden ohjelmointiin liittyvien materiaalien kanssa. Kiitokset kaikille tahoille, jotka osaltaan ovat tuottaneet yhteiseen tarpeeseen vastaavaa materiaalia. Näitä tahoja on lukuisia, mainittakoon tässä vaikkapa. Koodiaapinen, Innokas-verkosto ja Anna-Maija Partanen (ULapland). Ohjelmointipolun työtapoja sekä tausta- ja virikemateriaaleja saa vapaasti hyödyntää ja parantaa yksiköiden omassa OPS-prosessissa ja ohjelmoinnin opetuksessa.

Ohjelmointipolkua laatineiden opettajien puolesta,
eNorssin projektivastaava, Mikko Horila (@uta.fi)

1. - 2. 3. - 4. 5. - 6. 7. - 9.

Tavoitteet Harjoitellaan ohjeiden antamista ja niiden noudattamista ja oppii niiden

yhteyden ohjelmointiin

Tutustutaan ohjelmoinnin

perusasioihin kuten ohjeiden

antaminen, syy-seuraussuhteet

Saadaan aikaan positiivisia kokemuksia ohjelmointiin liittyen

Tutustutaan ohjelmoinnin läsnäoloon arkielämässä

Kehitetään ohjelmointiin liittyviä ajattelutaitoja: vertailu, luokittelu

Syvennetään taitojaan

yksityiskohtaisten ohjeiden antamisessa

Tutustutaan ohjelmoinnin perusteisiin graafisessa

ohjelmointiympäristössä

Saadaan aikaan positiivisia kokemuksia ohjelmointiin liittyen

Tarkastellaan ohjelmoinnin läsnäoloa ympäristössä,

esimerkiksi robotiikassa ja teollisuudessa

Kehitetään ohjelmointiin liittyviä ajattelutaitoja: ongelmanratkaisu, luovuus

Tutustutaan ohjelmoinnin perusrakenteisiin (silmukka, if – then – else) sekä loogisiin operaatioihin (ei, ja, tai).

Osataan tehdä ohjeen mukaan ohjelmia graafisessa

ohjelmointiympäristössä

Saadaan aikaan positiivisia kokemuksia ohjelmointiin liittyen.

Harjoitellaan ohjelmoinnin

perusrakenteiden käyttöä: muuttuja, ehtolause ja toistolause.

Nivelletään ohjelmointia osaksi oppiaineiden opetusta.

Osataan tehdä ohjelmia graafisessa tai tekstipohjaisessa

ohjelmaympäristössä.

1. - 2. 3. - 4. 5. - 6. 7. - 9.

Materiaalit ja välineet

(esimerkkejä)

Pelejä ja leikkejä

Robogem-peli

Bee-Bot

Hello Ruby -kirja (Linda Liukas) Ope on robotti -leikki

Loogiset palat

Piirretään parin sanallisen ohjeen mukaan

Graafisia

ohjelmointiympäristöjä *Scratch Jr.*

Daisy the Dinosaur

Pelejä ja leikkejä

Robogem-peli

Loogiset palat

Binäärilukukortit

Kuvia binäärikoodin periaattein Salakirjoitus

Piirrä ohjeen mukaan

Graafisia

ohjelmointiympäristöjä *Scratch Jr.*

Koodaustunti.fi

Oppimateriaaleja

Piirrä ohjeen mukaan

Salakirjoitus

Helpot kaavat

taulukkolaskennassa

Graafisia ohjelmointiympäristöjä *Scratch*

Kodu

Lego Mindstorms

Oppimateriaaleja

Koodikerho.fi

koodaustunti.fi

code.org

www.codecademy.com/

www.w3schools.com/

(html, css, Javascript, php, Bootstrap)

Ohjelmointiympäristöjä

Scratch

Kodu

Lego Mindstorms

JavaScript-ohjelmointikieli

Selaineditorit

Notepad++

Processing-ohjelmointi

Lua-kieli

Racket

1.- 2. luokat

Tavoitteet:

- Tutustuu ohjelmointiin perusasioihin kuten ohjeiden antaminen, syy-seuraussuhteet -
- Harjoittelee ohjeiden antamista ja niiden noudattamista ja oppii niiden yhteyden ohjelmointiin

Yleistä:

- Alkuopetuksessa ohjelmoinnin opetus voi tapahtua suureksi osaksi ilman tietokonetta ja moni ohjelmointiin väljästi luokiteltava tehtävä on tuttua jo aiempien vuosien opetussuunnitelmista.
- Vaikka ohjelmointi on mainittu erityisesti matematiikan sisällöissä, voi ja kannattaa sitä toteuttaa myös muissa oppiaineissa (esim. ohjeiden antaminen ja noudattaminen)

Suositteluvia työtapoja ja materiaaleja 1.-2. luokkien opetukseen:

● [Robogem-ohjelmointipeli](#)

Robogem on ohjelmointiin liittyvään ajatteluun johdattava peli, joka soveltuu 2 - 4 oppilaalle. Peli sopii mainiosti vaikkapa matematiikan tai äidinkielen tunneille. Robogem on maksullinen tuote, joka sisältää pelilaudan ja muut peliin tarvittavat tarvikkeet.

● [Hello Ruby -kirja](#) (Linda Liukas)

Hello Ruby on tarina pienestä tytöstä, jolla on suuri mielikuvitus. Linda Liukas: "Uteliaisuudella pääsee pitkälle! Suurin osa meistä oppii ja kehittyy olemalla uteliaita, purkamalla ja kokeilemalla asioita, yrittämällä parantaa niitä." Siten koodauskaan ei ole vain tietotaitoa, vaan luomisen väline.

Joku toinen rakentaa legoja tai piirtää, kolmas taas koodaa mielikuvituksensa näkyväksi. Linda Liukkaan lapsia ja aikuisiakin inspiroiva kirja on lastenkirja teknologiasta, tietokoneista ja ohjelmoinnista.

● **Ope on robotti -leikki**

Leikissä oppilaat ohjaavat opettajaa (tai luokkatoveria) antamalla yksinkertaisia ja tarkkoja ohjeita kuten "kävele eteenpäin kolme askelta. Käänny vasempaan. Kulje eteenpäin vielä kolme askelta". Tavoitteena on harjoitella ohjelmoinnin perusasioita, eli ohjeiden antoa ja miten toimia tilanteessa, jolloin vastaan tulee vaikkapa este (seinä tms.)

● **Loogiset palat**

Loogisten palojen avulla voidaan harjoitella eri tavoin luokittelua, ominaisuuksien ja ehtojen toteutumista ja muuta loogista päättelyä. Pienissä loogisissa paloissa on neljä eri ominaisuutta ja neljä eri väriä. Katso vinkkejä esim.

http://www.opperi.fi/02_opetusvinkkeja/Loogiset_palat_ohje.pdf

<http://peda.net/veraja/nokia/emkoski/matikkanaata/toiminnallisia/toiminnallisialoog>

http://www.tevella.fi/Ohjeet/muut_matikkaty%C3%B6pajavinkit.pdf

● **Bee-Bot**

Bee-botit ovat ohjelmitavia mehiläisrobotteja, joilla harjoitellaan yksinkertaisten ohjeiden antamista (kulje eteenpäin, käänny...).

● **Scratch Jr.**

Scratch Jr. on yksinkertainen, pelinomainen ja ilmainen graafinen ohjelmointiympäristö, joka toimii niin iOS-laitteilla kuin Android-laitteillakin. Scratch Jr. toimii hyvin ensimmäisenä ohjelmointiympäristönä pienille lapsille ja se on opittavissa muutaman tunnin kokeilulla. Perustietoja ja ohjeita: <http://www.scratchjr.org/>

● **Daisy the Dinosaur**

Daisy the Dinosaur-sovellus on ilmainen ja hauska sovellus, jossa on helppo "vedä ja pudota käyttöliittymä". Sovelluksella opetellaan ohjelmoinnin perusasioita ratkaisemalla haasteita. Katso vinkkejä esim.: <https://www.youtube.com/watch?v=JNStkjGhxGQ>

Muita vinkkejä toteutukseen:

Ohjelmointi on ennen kaikkea loogista ajattelua, ohjeiden noudattamista ja antamista, syy-seuraussuhteiden hahmottamista ja sääntöjen merkityksen ymmärtämistä. Näin ajatellen ohjelmointia on laajasti ymmärtäen opetettu ja opiskeltu aina ennenkin - ohjelmoinnin tuloa alakouluun ei kannata pelätä. Ohjelmointi voi näkyä 1.-2. -luokilla mm. seuraavasti:

- ohjeiden laatiminen ja seuraaminen puhutulla kielellä ja kirjoittamalla/piirtämällä (kaikki oppiaineet)
- askartelu, käsillä tekeminen (ai, ks, ku)
- laskusäännöt, syy-seuraussuhteet (ma)
- suunnistaminen ja pihakartan lukeminen (li)
- Lisää hyviä vinkkejä:

<https://www.ulapland.fi/loader.aspx?id=b4741b4b-3fc9-4ebf-86b2-8c07c1b9d927>

3. - 4. luokat

Tavoitteet:

- Tutustuu ohjelmoinnin perusteisiin graafisessa ohjelmointiympäristössä
- Kehittää ohjelmointiin liittyviä ajattelutaitoja
- Syventää taitojaan yksityiskohtaisten ohjeiden antamisessa

Suosittelavia työtapoja ja materiaaleja 3. - 4. luokkien opetukseen:**● Robogem-ohjelmointipeli**

Robogem on ohjelmointiin liittyvään ajatteluun johdettava peli, joka soveltuu 2 - 4 oppilaalle. Peli sopii mainiosti vaikkapa matematiikan tai äidinkielen tunneille. Robogem on maksullinen tuote, joka sisältää pelilaudan ja muut peliin tarvittavat tarvikkeet.

● Piirrä ohjeen mukaan

Ajatuksena on harjoitella ohjeiden antamista ja noudattamista. Tehtävä tehdään parityönä. Toisella oppilaista on piirrosmalli ja toisella kynä sekä ruutupaperi. Oppilas, jolla on kynä ja paperi, ei saa nähdä mallipiirrosta. Oppilas, jolla on piirrosmalli, pyrkii ohjeistamaan paria piirtämään piirrosmallin mukaisen kuvan. Lopuksi katsotaan, millaiseen lopputulokseen päästiin ja arvioidaan prosessia.

● Loogiset palat

Loogisten palojen avulla voidaan harjoitella eri tavoin luokittelua, ominaisuuksien ja ehtojen toteutumista ja muuta loogista päättelyä. Pienissä loogisissa paloissa on neljä eri ominaisuutta ja neljä eri väriä. Katso vinkkejä esim.

http://www.opperi.fi/02_opetusvinkkejä/Loogiset_palat_ohje.pdf,

<http://peda.net/veraia/nokia/emkoski/matikkanaata/toiminnallisia/toiminnallisialoog>

, http://www.tevella.fi/Ohjeet/muut_matikkaty%C3%B6pajavinkit.pdf

● Scratch Jr.

Scratch Jr. on yksinkertainen, pelinomainen ja ilmainen graafinen ohjelmointiympäristö, joka toimii niin iOS-laitteilla kuin Android-laitteillakin. Scratch Jr. toimii hyvin ensimmäisenä ohjelmointiympäristönä pienille lapsille ja se on opittavissa muutaman tunnin kokeilulla. Perustietoja ja ohjeita: <http://www.scratchjr.org/>

● Sphero

Sphero on ohjelmoitava kauko-ohjattava pallo, joka on yhteensopiva 25 appsin kanssa. Spheron avulla voi hyvin konkretisoida komentoketjuja.

● Binääriluvut

Tietokoneiden toiminta perustuu binäärilukuihin, joihin voidaan johdattaa muutamilla harjoituksilla. Oppilaille osoitetaan, että järjestelmä eroaa tutummasta kymmenjärjestelmästä.

Binäärilukukorteilla voi vaikkapa muodostaa ryhmässä lukuja oppilasvoimin.

Video: <https://drive.google.com/drive/folders/0B0GI6VOYOYvtNHZuMUI5eE42WjA>

Lataa kortit: <https://drive.google.com/file/d/0B0GI6VOYOYvtMIQ3ZG1sSFUzYTg/view>

Kuvia binäärikoodin periaattein

Binäärilukuja voidaan mainiosti havainnollistaa mustavalkokuvilla, joissa jokainen yksittäinen piste on värittämätön/valkoinen (0) tai väritetty/musta (1).

Sopivia valmiita harjoituksia:

<https://drive.google.com/file/d/0B0GI6VOYOYvtX1NrOFNkQXFPX2M/view> (sivut 15 - 20)

● Salakirjoitus

Salakirjoitus sopii erilaisten ohjelmointiin liittyvien asioiden opettamiseen: kielen kääntäminen, salaus, ongelmanratkaisu jne.

Sopivia valmiita harjoituksia:

<http://blogs.helsinki.fi/summamutikka/files/2015/02/Salakirjoituksia.pdf>

● Koodaustunti.fi

Koodaustunti.fi-verkkosivuston materiaaleja voidaan käyttää jo 3. - 4. luokilla - joko yhteisesti tai eriyttävästi. Se on valmis ja helposti käyttöönotettava valmis materiaali, joka sisältää ohjeet itsessään. Suurin osa oppilaista pystyy etenemään hyvin itsenäisesti, ja pienin askelin vaikeutuvia tehtäviä on runsaasti. www.koodaustunti.fi

Muita vinkkejä toteutukseen:

Ohjelmointi on ennen kaikkea loogista ajattelua, ohjeiden noudattamista ja antamista, syy-seuraussuhteiden hahmottamista ja sääntöjen merkityksen ymmärtämistä. Näin ajatellen ohjelmointia on laajasti ymmärtäen opetettu ja opiskeltu aina ennenkin - ohjelmoinnin tuloa alakouluun ei kannata pelätä. Ohjelmointi voi näkyä 3.-4. -luokilla mm. seuraavasti:

- ohjeiden seuraaminen (kaikki oppiaineet)
- ohjeiden antaminen ja noudattaminen (kielet, ma, ks)
- laskusäännöt, syy-seuraussuhteet (ma)
- suunnistaminen ja kartan lukeminen (li, bg)

Lisää hyviä vinkkejä: <https://www.ulapland.fi/loader.aspx?id=b4741b4b-3fc9-4ebf-86b2-8c07c1b9d927>

5. - 6. luokat

Tavoitteet:

- 5. - 6. luokkien aikana ohjelmoimisen opetuksen painotus on visuaalisessa ohjelmoimisessa
- Tehdään ohjeen mukaan ohjelmia graafisessa ohjelmointiympäristössä
- Ratkaistaan ongelmia ja tuotetaan vaihtoehtoisia ratkaisuja
- Tutustutaan ohjelmoinnin perusrakenteisiin (silmukka, funktio, if – then – else) sekä loogisiin operaatioihin

(ei, ja, tai)

- Saadaan aikaan positiivisia kokemuksia ohjelmointiin liittyen
- Ymmärtää ohjelmoinnin läsnäolo arkielämässä

Suositteluvia työtapoja ja materiaaleja 5. - 6. luokkien opetukseen:

● Piirrä ohjeen mukaan

Ajatuksena on harjoitella ohjeiden antamista ja noudattamista. Tehtävä tehdään parityönä. Toisella oppilaista on piirrosmalli ja toisella kynä sekä ruutupaperi. Oppilas, jolla on kynä ja paperi, ei saa nähdä mallipiirrosta. Oppilas, jolla on piirrosmalli, pyrkii ohjeistamaan paria piirtämään piirrosmallin mukaisen kuvan. Lopuksi katsotaan, millaiseen lopputulokseen päästiin ja arvioidaan prosessia.

● Salakirjoitus

Salakirjoitus sopii erilaisten ohjelmointiin liittyvien asioiden opettamiseen: kielen kääntäminen, salaus, ongelmanratkaisu jne. Sopivia valmiita harjoituksia:

<http://blogs.helsinki.fi/summamutikka/files/2015/02/Salakirjoituksia.pdf>

● Helpot kaavat taulukkolaskennassa

Taulukkolaskentaohjelmissa (esim. Excel) on mielekäästä tehdä omia yksinkertaisia laskentakaavoja.

Ohjeita esimerkiksi: <https://www.youtube.com/watch?v=UXE01zGj8Jk>

● Graafisten ohjelmointiympäristöjen käyttö tietokoneilla ja mobiililaitteilla

Scratch

Scratch on MIT:n kehittämä lapsille ja nuorille soveltuva ohjelmointikieli ja maksuton ohjelmointiympäristö. Scratchin graafisen käyttöliittymän avulla on helppo oppia ohjelmoinnin periaatteita, ongelmanratkaisua ja loogista päättelyä. Scratch sisältää grafiikka- ja äänityökaluja, valmiita kuvia ja ääniä voi poimia kirjastosta.

video: <https://www.youtube.com/watch?v=ywG6lv9mFLI>

vinkejä: www.koodaustunti.fi, <https://sites.google.com/site/scratchklubi/>,

<http://www.slideshare.net/mobile/raisukka/scratch-ohjelmointia>,

http://linkki.cs.helsinki.fi/sites/linkki.cs.helsinki.fi/files/scratchohjelmointikerhomateriaali_0.pdf

Kodu

Kodu on visuaalinen ohjelmointikieli, joka on suunniteltu erityisesti pelien tekoon. Se on suunniteltu erityisesti lasten käytettäväksi. Ohjelmointikieli on visuaalinen ja helppo.

Syöttölaitteena voi käyttää näppäimistöä ja hiirtä tai XBOXin ohjainta

video: <https://www.youtube.com/watch?v=8-FjskAHgN0>

Lego Mindstorms

Lego Mindstorms yhdistää legorakentelun, robotiikan ja ohjelmoinnin. Valmiiden sarjojen ja

tehtävien avulla voidaan harjoitella ohjelmointia visuaalisia komponentteja yhdistellen.

video: https://www.youtube.com/watch?v=hkN_DObmh1c

vinkkejä: <https://education.lego.com/en-us/lesi/>

Muita vinkkejä toteutukseen:

- Robottiikka teollisuudessa, ihmisen ja koneen yhteistoiminta (yo, ks)
- Ihmisen korvaaminen robotilla? (etiikka)
- Päätely, ongelmanratkaisu, kielentäminen (ma)
- Ohjeiden antaminen ja noudattaminen (kaikki aineet)
- Ohjeiden antaminen ja noudattaminen (kaikki aineet)

Lisää hyviä vinkkejä: <https://www.ulapland.fi/loader.aspx?id=b4741b4b-3fc9-4ebf-86b2-8c07c1b9d927>

7. - 9. luokat

Tavoitteet:

- Saadaan aikaan positiivisia kokemuksia ohjelmointiin liittyen.
- Harjoitellaan ohjelmoinnin perusrakenteiden käyttöä: muuttuja, ehtolause ja toistolause.
- Nivelletään ohjelmointia osaksi oppiaineiden opetusta.
- Osataan tehdä ohjelmia graafisessa tai tekstipohjaisessa ohjelmaympäristössä. Voidaan tutustua ohjelmalliseen ajatteluun myös taulukkolaskentaohjelman avulla.

Huomioitavaa:

Kaikille yläkouluille sopivaa ja "oikeaa" ohjelmointikieltä tai kokonaisratkaisua ei ole helppoa löytää. Siksi seuraavilla sivuilla on ehdotettu useita erilaisia ohjelmointikieliä ja kukin koulu etsii niistä itselleen sopivimman ratkaisun tai käyttää jotakin tämän materiaalin ulkopuolista ohjelmointikieltä / lähestymistapaa. Ajatuksena ei ole sitoa kouluja yhteen malliin, vaan tarjota enemmänkin virikkeitä ja vaihtoehtoja sopivan ratkaisun tueksi.

Ohjelmointiympäristöjä ja materiaaleja

koodaustunti.fi ja code.org

Scratch-pohjaisia vaiheittain eteneviä harjoituksia. Koodaustunnin tarkoituksena on tutustuttaa "koodaamiseen" ja tehdä sitä arkipäiväiseksi, varsinaiset harjoitukset code.org-sivustolla. Soveltuu useaan eri oppiaineeseen, mahdollisuus useaan eri kieleen (esimerkiksi englanti, saksa, ranska).

www.codecademy.com/

Mittava kokoelma ohjelmoinnin periaatteiden ja lausekielisen ohjelmoinnin harjoituksia. Työskentely on selainpohjaista, oppilaat voivat rekisteröityä palveluun ja harjoitella myös omatoimisesti. Ohjelmointikielinä esimerkiksi Java, Javascript, Python, Ruby ja php.

www.w3schools.com/

Kansainvälisen W3C-konsortion opetussivusto, jossa voi harjoitella esimerkiksi html-, css- ja JavaScript –kieliä. W3C-konsortion tarkoituksena on osaltaan standardisoida web-julkaisemisen välineitä niin, että esimerkiksi eri laitteet ja selainohjelmat näyttäisivät tuotetut sisällöt yhtäläisellä tavalla. Sivustolla myös hyvä osio html-, css- ja JavaScript –kielet yhdistävälle Bootstrapille (<http://getbootstrap.com/>).

Lego Mindstorms

Lego Mindstorms yhdistää legorakentelun, robotiikan ja ohjelmoinnin. Ohjelmointi suoritetaan yhdistelemällä visuaalisia komponentteja. Ideoita projekteihin löytyy runsaasti Youtubesta.

<https://education.lego.com/en-gb/lesi/middle-school/mindstorms-education-ev3>

Scratch

Scratch on lapsille ja nuorille soveltuva graafinen ohjelmointiympäristö. Scratchin graafisen käyttöliittymän avulla on helppo oppia ohjelmoinnin periaatteita, ongelmanratkaisua ja loogista päättelyä. Scratchista on monia muunnelmia: (esimerkiksi code.org-sivuston harjoitukset). Scratch:n avulla voi suunnitella ja toteuttaa myös laajoja ja vaativiakin ohjelmointiprojekteja.

<https://scratch.mit.edu/>

http://linkki.cs.helsinki.fi/sites/linkki.cs.helsinki.fi/files/scratch-ohjelmointikerhomateriaali_0.pdf

<http://avoinoppikirja.fi/tite-scratch>

<http://www.teromakotero.fi/scratch-klubi/>

Kodu

Kodu on Microsoftin kehittämä visuaalinen ohjelmointikieli, joka on suunniteltu erityisesti pelien tekoon. Toimii Windows-ympäristössä. Ohjelmointiympäristö on suunniteltu käytettäväksi tietokoneeseen kytketyn xBox-ohjaimen avulla.

video: <https://www.youtube.com/watch?v=8-FjskAHgN0>

Processing

Processing-kieli on taiteilijoiden kieli taiteilijoille. Muistuttaa Javaa, mutta yksinkertaisempaa.

<https://processing.org/>, <http://www2.uef.fi/fi/anja.kareinen/processing>

Lua

Lua-kieltä käytetään mm. peliohjelmoinnissa. Opetusmateriaalia on tarjolla suhteellisen vähän. Lua-kieleen on tehty materiaalia matematiikan näkökulmasta:

http://www.codebymath.com/index.php/welcome/lesson_menu.

Racket

Ohjelmoinnin opettamiseen kehitetty kieli, johon voi perehtyä esimerkiksi koodiaapinen.fi-sivustolla.

http://racket.koodiaapinen.fi/manuaali/racket_bsl.html

Koodikerho

Ohjelmoinnin opettamisessa kerhomuotoinen toiminta mahdollistaa syvemmän opiskelun aihepiiristä kiinnostuneille. Esimerkkinä toimii mm. koodikerho. Koodikerho on konsepti, jonka sivustolle kerätään materiaaleja kerhojen toteuttamisen näkökulmasta.

<http://koodikerho.fi>

JavaScript-ohjelmointikieli

JavaScript on Web-ympäristöön suunniteltu ja laajasti käytetty ohjelmointikieli, joka soveltuu erinomaisesti ensimmäiseksi tekstipohjaiseksi ohjelmointikieleksi. JavaScript-ohjelmia voidaan kirjoittaa millä tahansa tekstieditorilla (Notepad, Notepad++) eikä se tarvitse erillistä kääntäjää tai ohjelmointiympäristöä – kaikki nykyaikaiset Internet-selaimet suorittavat JavaScript-lähdekoodia sellaisenaan tai upotettuna osaksi html tiedostoa. Syntaksiltaan JavaScript muistuttaa yleisesti käytettyjä C-sukuisia ohjelmointikieliä (C, C++, C#, Java).

Ensimmäiseksi tekstipohjaiseksi ohjelmointikieleksi JavaScript soveltuu erinomaisesti laajan vapaasti saatavan esimerkki- ja opetusmateriaalinsa sekä helppolukuisen dokumentaationsa ansiosta (w3schools.com). JavaScript-ohjelmointikieltä on helppo oppia ja jo muutamalla koodirivillä saa aikaiseksi visuaalisia ja interaktiivisia ohjelmia. Lisäksi JavaScript-ohjelmointikielessä on sisäänrakennettuna lähes kaikki funktiolaskimessa olevat laskutoiminnot.

JavaScript-ohjelmointikieli soveltuu niin ammattikäyttöön kuin ensimmäiseksi kieleksi ohjelmoinnin opettelussa. Helposti rakennettavat ja osaksi Internet-sivuja upotettavat ohjelmat innostavat oppilasta oppimaan ohjelmointia – oli kyseessä sitten hauskat peliohjelmat tai käytännön ongelmia ratkaisevat ohjelmat.

Valmiita JavaScript harjoituksia alkeista eteenpäin löytyy esimerkiksi codecademy.com -sivustolta
Materiaaleja 7. - 9. luokkien opetukseen (esimerkkinä Scratch ja Processing-kieli. Koulu voi halutessaan päättää muunkin lähestymistavan/ohjelmointikielen)

Ohjelmoinnin perusteiden ymmärtäminen
Scratch yms. graafinen
ohjelmointiympäristö

Peräkkäiset komennot: esim. liikkuminen, kääntyminen, odotus, värin vaihtaminen, siirtyminen kuvaruudulla tiettyyn paikkaan, liukuminen, äänet, arpominen

Tekstipohjainen

ohjelmointiympäristö

[Graafisten kuvioiden piirtäminen](#) [Tehtäviä grafiikasta](#)

[Tehtäviä värittämisestä](#)

Taulukkolaskentaohjelma Kaavojen tekeminen

Muuttujan käyttö ohjelmoinnissa Ohjelmat, jotka voivat esimerkiksi kysellä ja tarkastaa onko vastaus

oikein

Kierroslaskuri

Pistelaskuri (if-lauseen yhteydessä)

Arpominen

Ehtolausekkeet Yksinkertainen if-lause:

Törmäyksen testaaminen:

a. Törmätäänkö seinään

b. Törmätäänkö toiseen olioon

Pistelaskuri

Sisäkkäiset if-lauseet / operaattoreiden käyttö (eriyttävää)

Törmäyksen testaaminen:

a. Törmääkö kuva seinään tai

onko kuva/ hiiri / kohdistin

esim. punaisella alueella

[Kuvion liikuttaminen esim.](#)

[näppäimistön avulla](#)

[Kuvion koon muuttaminen](#)

[Kuvion paikan arpominen](#)

Yksinkertainen if-lause

[Mitä näppäintä painettu](#)

[Onko kuvio liikkunut kuvaruudun](#) [ulkopuolelle](#)

Sisäkkäiset if-lauseet /

operaattoreiden käyttö (eriyttävää) [Onko kuvio osunut johonkin toiseen](#) [kuvaan tai hiirellä klikattu](#)

[tietyllä alueella](#)

jos-lause taulukkolaskentaohjelmassa: esim. jos hinta alle 100 euroa niin ei alennusta, muuten alennus 10%

b. Törmääkö kuva seinään esim.
kymmenennen kerran

Toistolausekkeet Liike

Kuvioiden ja säännöllisten
monikulmioiden piirtäminen

Hyvät ohjelmointikäytännöt Ohjelman pitäminen selkeänä ja helposti luettavana
For- ja/tai while-lause (eriyttävää) Kuvioiden piirtäminen

Ohjelmakoodin kommentointi [Metodit ohjelmoinnissa](#)

Selkeän ja helposti päivitettävän taulukon luominen (esim. absoluuttinen osoite)

Mahdollisia käsittelytapoja oppiaineittain jaoteltuna

oppiaine Scratch yms. graafinen ohjelmointiympäristö Tekstipohjainen ohjelmointiympäristö
Taulukkolaskentaohjelma

Matemaattiset aineet Kulmakäsite

Geometristen kuvioiden piirtäminen

[Geometrian tutkimustehtäviä](#)

[Pienten opetusohjelmien tekeminen](#)

(esim. peruslaskutoimitukset negatiivisilla luvuilla, potenssi, neliöjuuri)

[Suurennus ja pienennys](#)

[Peilaus ja siirto](#) (voit käyttää taustakuvana myös koordinaatistoa)

[Eesitys](#)

Prosenttilasku: esim. tee ohjelma, joka laskee lopullisen hinnan,
kun alkuperäinen hinta ja alennusprosentti tiedetään

Kuvataide [Avatarin suunnittelu](#)

Animaatiot: [asusteiden avulla](#), [asusteiden ja liikkeen avulla](#)

[Tarina/esitys](#)

Kielet Opetusohjelmien tekeminen

[Tarina/esitys](#)

Reaaliaineet Opetusohjelmien tekeminen

[Tarina/esitys](#)

Geometristen kuvioiden piirtäminen

Graafisten kuvioiden tekeminen
Prosenttilaskenta

Korkoa korolle tehtävät Geometrian kaavoja

Tilastotiede

Yksinkertaisten taulukoiden tekeminen ja
tutkimustehtävät

Ohjelmointi opetussuunnitelman perusteissa

Perusopetuksen opetussuunnitelman perusteet päivittyvät syksyllä 2016. Koodaustaidot mainitaan osana laaja-alaisia osaamistavoitteita sekä ala- että yläkoulun puolella. Opetushallitus on hyväksynyt esiopetuksen-, perusopetuksen-, ja lisäopetuksen opetussuunnitelman perusteet 22.12.2014. Alla koottuina kohdat, joissa ohjelmointi mainitaan määräyksessä perusopetuksen opetussuunnitelman perusteista:

Opetus vuosiluokilla 1–2

Tieto- ja viestintäteknologinen osaaminen:

- *Oppilaat saavat ja jakavat keskenään kokemuksia digitaalisen median parissa työskentelystä sekä ikäkaudelle sopivasta ohjelmoinnista (OPS 2016, 101). [1]*

Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 1–2

- *Tutustuminen ohjelmoinnin alkeisiin alkaa laatimalla vaiheittaisia toimintaohjeita, joita myös testataan (OPS 2016, 129). [1]*

Opetus vuosiluokilla 3–6

Tieto- ja viestintäteknologinen osaaminen (L5)

- *Ohjelmointia kokeillessaan oppilaat saavat kokemuksia siitä, miten teknologian toiminta riippuu ihmisen tekemistä ratkaisuista (OPS 2016, 157). [1]*

Matematiikan opetuksen tavoitteet vuosiluokilla 3–6

- *T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoina graafisessa ohjelmointiympäristössä (OPS 2016, 235). [1]*

Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 3–6

- *Suunnitellaan ja toteutetaan ohjelmia graafisessa ohjelmointiympäristössä (OPS 2016, 235). [1]*

Matematiikan arviointikriteerit 6. vuosiluokan päätteeksi arviota ”hyvä” / arvosanaa kahdeksan varten

- *T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoina graafisessa ohjelmointiympäristössä (OPS 2016, 239). [1]*
- *Ohjelmointi graafisessa ohjelmointiympäristössä (OPS 2016, 239). [1]*
- *Oppilas osaa ohjelmoida toimivan ohjelman graafisessa ohjelmointiympäristössä (OPS 2016, 239). [1]*

Käsityön tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 3–6

- *Harjoitellaan ohjelmoimalla aikaan saatuja toimintoja, joista esimerkkinä robotiikka ja automaatio (OPS 2016, 271). [1]*

Opetus vuosiluokilla 7–9

Tieto- ja viestintäteknologinen osaaminen (L5)

- *Ohjelmointia harjoitellaan osana eri oppiaineiden opintoja (OPS 2016, 284). [1]*

Matematiikan opetuksen tavoitteet vuosiluokilla 7–9

- *T20 ohjata oppilasta kehittämään algoritmista ajatteluaan sekä taitojaan soveltaa matematiikkaa ja ohjelmointia ongelmien ratkaisemiseen (OPS 2016, 375). [1]*

Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 7–9

- *Ohjelmoidaan ja samalla harjoitellaan hyviä ohjelmointikäytäntöjä (OPS 2016, 375). [1]*

Matematiikan päättöarvioinnin kriteerit hyvälle osaamiselle (arvosanalle 8) oppimäärän päättyessä

- *T20 ohjata oppilasta kehittämään algoritmista ajatteluaan sekä taitojaan soveltaa matematiikkaa*

ja ohjelmointia ongelmien ratkaisemiseen (OPS 2016, 379). [1]

- *Algoritminen ajattelu ja ohjelmointitaidot (OPS 2016, 379). [1]*
- *Oppilas osaa soveltaa algoritmisen ajattelun periaatteita ja osaa ohjelmoida yksinkertaisia ohjelmia (OPS 2016, 379). [1]*

Käsityön tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 7–9

- *Käytetään sulautettuja järjestelmiä käsityöhön eli sovelletaan ohjelmointia suunnitelmiin ja valmistettaviin tuotteisiin (OPS 2016, 431). [1]*

Lähteet:

http://www.oph.fi/download/163777_perusopetuksen_opetussuunnitelman_perusteet_2014.pdf