

This is a self-archived version of an original article. This version may differ from the original in pagination and typographic details.

Author(s): Roos, Susanne; Hilpinen, Markku; Fagerlund, Janne

Title: Ohjelmoinnillinen ajattelu laaja-alaisessa oppimisessa

Year: 2022

Version: Published version

Copyright: © 2022 FCLab.fi ja eNorssi

Rights: In Copyright

Rights url: <http://rightsstatements.org/page/InC/1.0/?language=en>

Please cite the original version:

Roos, S., Hilpinen, M., & Fagerlund, J. (2022). Ohjelmoinnillinen ajattelu laaja-alaisessa oppimisessa. In T. Tammi, & M. Horila (Eds.), *Oppimis- ja toimintaympäristöjen kehittäminen harjoittelukouluissa. III, Tilat ja tekniikka pedagogiikan kehittämisen tukena* (pp. 29-40). eNorssi - Opettajankouluttajien yhteistyöverkosto. Suomen harjoittelukoulujen julkaisuja. <https://enorssi.fi/oppimis-ja-toimintaymparistojen-kehittaminen-harjoittelukouluissa-iii/>

OHJELMOINNILLINEN AJATTELU LAAJA-ALAISESSA OPPIMISESSA

Susanne Roos, Markku Hilpinen, FCLab/Jyväskylän normaalikoulu
Janne Fagerlund, Koulutuksen tutkimuslaitos, Jyväskylän yliopisto

Laaja-alainen ohjelmointi

Ohjelmointitaidot ja niiden kautta opittava laajempi osaaminen, ohjelmoinnillinen ajattelu, sisällytettiin nykyiseen perusopetuksen opetussuunnitelmaan (Opetushallitus 2014) kaikkien vuosiluokkien osalta osaksi matemaattisten taitojen tavoitteita ja sisältöjä. Luokilla 3–6 ja 7–9 ohjelmointi on huomioitu myös käsityön tavoitteissa. Matematiikan oppikirjat sisältävät yhtenä osa-alueena ohjelmoinnin, jota on mahdollisuus harjoitella erilaisten kirjallisten tehtävien avulla. Alkuopetuksen matemaattisten taitojen tavoitteissa ja sisältö-alueissa kohdennetaan ohjelmointi erityisesti ajattelutaitojen kehittymiseen. Vaikka ohjelmointi sijoitetaan perusopetuksen opetussuunnitelman laaja-alaisen osaamisen tavoitteissa ainoastaan tieto- ja viestintäteknologisen osaamisen (L5) sektorille, voidaan ohjelmointi nähdä paljon laaja-alaisempana mahdollisuutena oppimisen eri osa-alueilla, esimerkiksi erilaisten tietojen ja taitojen harjoittelussa opetussuunnitelman eri alueilla (Fagerlund & Roos 2020). Ytimeltään ohjelmoinnin kautta opittava ohjelmoinnillinen ajattelu voidaan ymmärtää monipuolisena tieto- ja taitokokonaisuutena, joka ruokkii sekä eri tilanteissa sovellettavia ohjelmoinnillisia ongelmanratkaisutaitoja että ymmärrystä ohjelmoidusta maailmasta (Fagerlund 2021). Kouluoppimisessa ohjelmoinnin alkeisiin tutustutaan lasten ja nuorten ikätason mukaisilla tavoilla: leikitellen, oivaltaen ja opetellen yhdessä erilais-

ten ajattelutaitojen lisäksi esimerkiksi vuorovaikutustaitoja, oppimaan oppimista sekä monilukutaitoa. Tässä artikkelissa esittelemme käytännön esimerkkien avulla, miten erilaiset projektit ja monialaiset oppimiskokonaisuudet voivat tuoda ohjelmoinnillisen ajattelun osaksi isompaa kokonaisuutta oppilaita motivoivalla tavalla.



Kuvio 1.
Laaja-alainen osaaminen ohjelmoinnillisen ajattelun kautta.

Laaja-alainen osaaminen koostuu tietojen, taitojen, arvojen, asenteiden ja tahdon muodostamasta seitsemän alueen kokonaisuudesta. Artikkelin projekteissa toteutuvat vahvasti laaja-alaisen osaamisen tavoitteista ajattelu- ja oppimaan oppiminen, vuorovaikutus- ja ilmaisutaidot, monilukutaito sekä luonnollisesti tieto- ja viestintäteknologinen osaaminen. Kun tätä peruskoulun opetussuunnitelman perusteista (Opetushallitus 2014) löytyvää kiekkoa tarkastelee ohjelmoinnillisen ajattelun kautta, seitsemälle osaamiskokonaisuudelle voi löytää perusteen ohjelmointitavoitteet huomioiden. Kuviossa 1. laaja-alaiset tavoitteet on mukautettu ohjelmoinnillisen ajattelun tavoitteiksi.

Ohjelmoinnin tavoitteellista opettamista ohjaa myös Opetushallituksen Uudet lukutaidot-kehittämishjelma (www.uudetlukutaidot.fi). Ohjelman kolme osaamisaluetta ovat medialukutaito, ohjelmointiosaaminen sekä tieto- ja viestintäteknologinen osaaminen. Ohjelmointiosaamisen määritelmässä korostetaan erityisesti monipuolisia ajattelun taitoja ja ymmärrystä digitaalisesta, ohjelmoidusta maailmasta ja siinä toimimisesta. Se on jaettu kolmeen pääalueeseen, joita ovat ohjelmoinnillinen ajattelu, tutkiva työskentely ja tuottaminen sekä ohjelmoidut ympäristöt ja niissä toimiminen. Pääalueet ja niihin liittyvät tarkemmat kuvaukset pohjautuvat alakoulun osalta Perusopetuksen opetussuunnitelman perusteissa (Opetushallitus 2014) tehtyyn tarkasteluun.

Harjoittelukoulujen yhteistyöverkosto eNorssi on omalta osaltaan pyrkinyt helpottamaan ohjelmoinnin roolin jäsentämistä osana opetussuunnitelmaa sekä eri vuosiluokkien opetusta Ohjelmointipolun avulla (eNorssi 2018). Ohjelmointipolku tarjoaa tietoa ja esimerkinomaisia konkreettisia keinoja ohjelmoinnin opetuksen suunnitteluun ja toteutukseen eri luokka-asteilla.

Harjoittelukoulujen yhteistä tietostrategiaa päivitetään kolmen vuoden välein. Myös ohjelmointipolkua päivitetään osana strategiatyötä. Tulevalle kaudelle 2022–2025 laadittuun strategiaan sisällytetty vuosiluokittainen osaamistasotaulukko mukailee Uudet lukutaidot-kehittämishjelman osaamisen kuvauksia. Tätä kautta osaamistasoissa on omana osa-alueenaan myös ohjelmoinnillinen ajattelu. Ohjelmointipolusta pyritään rakentamaan väline, joka osaamiskuvausten rinnalla helpottaa ohjelmoinnillisen ajattelun ja ohjelmoinnin opetteluun liittämistä osaksi muuta oppimista.

Matatalab ja ScratchJr osana monialaista projektia

Suomen harjoittelukoulujen FCLab-hankkeen yritysyritysteistyön myötä Joensuun ja Jyväskylän normaalikoulut aloittivat yhteistyön kiinalaisen Matatalabin kanssa. Keväällä 2021 pilotoitiin Matatalabin 5–9-vuotiaille suunnattuja ohjelmointivälineitä testaamalla niitä ensin luokissa. Tähän yhteistyöhön osallistuivat myös normaalikoulujen opetusharjoittelijat. Kokeilujen jälkeen yhdysopettajat pitivät koordinoijan kanssa yhdessä kansallisia webinaareja, joissa esiteltiin ohjelmointivälineiden toimintoja ja niiden avulla tehtyjä projekteja.

Matatalab-ohjelmointilaitteistoon sisältyy alusta, jolle haluttu ohjelma koodataan erilaisilla konkreettisilla komentopalikoilla. Alustaan liitetään "majakka", joka lukee käskysarjan alustalle asetetuista palikoista ja välittää sen bluetooth-yhteyden välityksellä robotille. Mu-

kana on myös maisema-alusta robotille, erilaisia osia robotin reittien suunnitteluun sekä eritasoisia koodaushaasteita, joita voi ratkoa edeten taso kerrallaan. Lisäpakettien avulla Matatalab-välineistö monipuolistuu muun muassa taiteen, liikunnan ja musiikin puolelle. Legopalikat ovat yhteensopivia Matatalabin kanssa, joten niiden avulla voi rakennella ohjelmointiympäristöä monipuolisemmaksi.

Matatalab-laitteisto mahdollistaa ohjelmoinnin yhdistämisen myös luovempaan tuottamiseen ja monialaisiin projekteihin. Sen ohjelmointiympäristöä on mahdollista muokata eri teeman mukaiseksi esimerkiksi askartelemalla. Myös robottien päälle voi rakennella teemaan sopivia hahmoja, joita on mahdollista liikutella ohjelmoimalla. Näin eri oppiaineiden tavoitteita on mahdollista yhdistää ohjelmoinnin avulla laajemmiksi projekteiksi.

ScratchJr on mobiililaitteilla (esim. iPadeilla) toimiva varsinaisen Scratch-ohjelmointiympäristön esiasteeksi suunniteltu erityisesti alkuopetusikäisille lapsille tarkoitettu junioriversio. Scratchin tavoin se on kuvakepohjaiseen ohjelmointiin perustuva monipuolinen alkeisohjelmointiympäristö, jolla oppilaat voivat suunnitella ja toteuttaa omia luovia interaktiivisia tarinoitaan, animaatioitaan ja pienoisperlejä. Ympäristö on niin sanotusti avoin, eli se ei tarjoa suoraan valmiita haasteita tai tehtäväpohjia, vaan se rohkaisee ennen kaikkea omien ohjelmointitöiden luomiseen ”tyhjästä” soveltamalla erilaisia vapaasti valittavia visuaalisia hahmoja ja taustoja (tapahtumapaikkoja) sekä ääniä. Ohjelmoinnin näkökulmasta ScratchJr on suunniteltu erityisesti oppilaiden tutustuttamiseksi oikean ohjelmointikielen käyttöön sen jälkeen, kun ohjelmoinnin logiikan perusteisiin on tutustuttu leikinomaisilla ohjelmointiharjoitteilla (esim. alkeisroboteilla, leikkien). ScratchJr-ympäristössä käytettävä kuvakepohjainen ohjelmointikieli perustuu lähes yksinomaan värikkäiden ohjelmointilohkojen yhdistämiseen ”skripteiksi” eli käskysarjoiksi, missä perusajatuksena on erityisesti tuntuma hauska ja motivoivasta ”itse näpertelystä” sekä siitä, että koodilohkojen käyttö ei edellytä runsaiden tekstimäärien lukemista. (Bers ym. 2019.)

Seuraavaksi esittelemme Jyväskylän normaalikoulussa tehtyjä monialaisia projekteja, joissa ohjelmoinnillinen ajattelu on tavoitteellisesti mukana, ja Matatalab ja/tai ScratchJr ovat osana projektin toteuttamista.

Neljä esimerkkiä toteutetuista projekteista

1. Juonellisen tarinan kirjoittaminen/ohjelmoiminen yhteistoiminnallisesti

Kolmannella luokalla toteutetun kirjoitusprojektin lähtökohtana oli harjoitella vuoropuhelua sisältävän juonellisen tarinan kirjoittamista. Tavoitteena oli sisällyttää tarinaan selkeät tapahtumapaikat ja päähenkilöt. Niiden avulla rakennettiin johdonmukaisesti etenevä tarina, jolla oli selkeä loppuratkaisu. Kirjoittaminen toteutettiin yhteistoiminnallisesti samantasoisien kirjoittajaparin kanssa, jolloin oppilaat pystyivät tukemaan parhaiten toisiaan tekstin tuottamisessa (Ferguson-Patrick 2007). Ohjelmoinnin tavoitteena oli saada päähenkilöt liikkumaan mahdollisimman tarkkaan juonen mukaisesti. Ohjelmointi toteutettiin sekä Matatalab-roboteilla että ScratchJr-sovelluksella, jolloin kirjoitettu tarina tehtiin eläväksi ja esitettäväksi kahdella eri tavalla.



Kuva 1. Oppilaita kirjoittamassa yhteistä tarinaa.

Projektin vaiheet:

1. Kirjoitetaan parin kanssa tarina, joka sisältää vuoropuhelun.
 - a. Valitaan hahmot (päähenkilöt), jotka lähtevät johonkin (tapahtumapaikat).
 - b. Matkalla hahmot keskustelevat yhdessä (vuoropuhelu).
 - c. Keskustelun päätteeksi tapahtuu jokin ratkaisu (juonellinen tarina).
2. Ohjelmoidaan tarina Matatalab:lla.
 - a. Tapahtumapaikat valitaan alustasta ja päähenkilöt askarrellaan robottien päälle.
 - b. Juonen kulku ohjelmoidaan kahdella Matatalabilla.
3. Videoidaan ohjelmoitu tarina lukien samalla.
4. Ohjelmoidaan tarina sen jälkeen ScratchJr:lla.
 - a. Tapahtumapaikoiksi otetaan kuvat Matatalab-alustasta ja päähenkilöt muokataan sopiviksi ScratchJr-ohjelmalla.
 - b. Vuoropuhelu toteutetaan joko puhekuplilla tai äänittämällä.
 - c. Juonen kulku ohjelmoidaan.
5. Animoinnit ja videot esitetään koko luokalle.
6. Lopuksi tehdään tarinoista ja animaatioista itse- ja vertaisarvioinnit.



Kuva 2. Tarinan ohjelmointia Matatalabilla.

2. Ohjelmoi satu

Lukutaidon heikentyminen ja lukuinnostuksen väheneminen ovat olleet huolenaiheena jo jonkin aikaa. Sen myötä on alettu miettiä motivointikeinoja esimerkiksi erilaisten lukuhankkeiden myötä. Yhteistyössä eri yliopistojen, harjoittelukoulujen sekä peruskoulujen kanssa on meneillään kansainvälinen Lukupesä-hanke (<https://finrainfo.fi/lukupesa/>), jossa muun muassa erilaisten oppimisympäristöjen sekä pelillistämisen avulla pyritään kasvattamaan lukumotivaatiota. Tämän hankkeen innostamana kehiteltiin projekti, jonka vaiheita esitellään seuraavaksi. Projektin nimeksi muodotui Ohjelmoi satu.

Ohjelmoi satu -projektin tavoitteena oli yhdistää lukemista ja ohjelmointia niin, että lopputuloksena olisi animaatioksi kuvitettu satu. Projektin lähtökohtana oli oletus, että ohjelmointi toimisi mahdollisena motivointikeinona ja tukena lukemiselle. Ohjelmointitaidoissa oli tavoitteena hyödyntää vertaisoppimista. Tämä projekti toteutettiin koulumme kummityhteistyönä ensimmäisen luokan ja neljännen luokan oppilaiden kanssa. Yhteistyötä tehtiin kummipareittain.

Projektin vaiheet:

1. Neljännen luokan oppilaat valitsevat mukavan sadun, jonka he haluaisivat lukea yhdessä ensimmäisen luokan kummioppilaansa kanssa.
2. Kummiparit lukevat yhdessä sovittuina aikoina vuorotellen valitsemaansa satua eteenpäin. Lukemista eriytetään ensimmäisen luokan oppilaan lukutaidosta riippuen niin, että ekaluokkalaiselle tulee sopivia omia lukupätkiä. Kun toinen lukee, niin toinen kuuntelee.
3. Kun satu on luettu, siitä piirretään yhdessä kummiparin kanssa juonikaavio, 3–4 tapahtumapaikkaa ja päähenkilöt, kaikki erikseen. Juonikaavio toimii ohjelmoidessa animaation ”punaisena lankana”, tapahtumapaikat ohjelmointitaustoina ja päähenkilöiden kuvat ohjelmitavina hahmoina.
4. Ohjelmointi toteutetaan ScratchJr-ohjelmointialustalla. Nelosluokan kummit, jotka hallitsevat ScratchJr:n käytön, opettavat samalla ohjelmointitaitoja nuoremmille.
5. Lopuksi ohjelmoiduista saduista pidetään koulussa kirjavinkkauksia lukemalla katkelmia ja näyttämällä yhdessä ohjelmoituja animaatioita.

3. Opetusanimaatio hämähäkkiteemalla

Matatalab-ohjelmointilaitteisto sisältää erilaisia oheistuotteita, joista yksi on taidepaketti. Sen avulla voi harjoitella piirto-ohjelmointia. Mahdollisuutena on piirtää ohjelmointimalleista esimerkiksi erilaisia tasokuvioita robottia ohjelmoiden. Ohjelmitavaan robottiin asetetaan kynä, joka piirtää ohjelmoidun kuvion. Kuvion muodostuminen perustuu perusohjelmointitoimintojen lisäksi kulman erilaisiin astemääritelmiin.



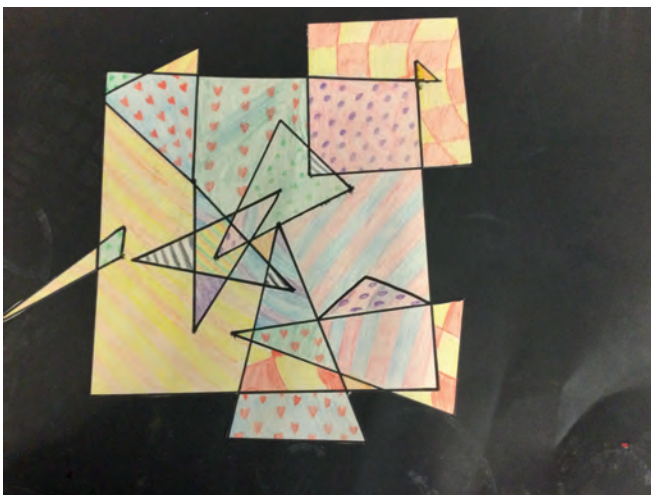
Kuva 3.
Piirto-ohjelmointia.

Kuvataidetta ja luonnontiedettä yhdistävä kokonaisuus hämähäkkiteemalla toteutettiin pareittain Matatalabin ja ScratchJr:n avulla ohjelmoiden. Projektissa hyödynnettiin piirtomahdollisuutta, jonka avulla suunniteltiin ja toteutettiin yhdessä kuvioita, joista koostui lopulta verkkomainen taideteos. Taideteosta taustakuvana käyttäen oli lopullisena tavoitteena saada ohjelmoiden aikaan opetusanimaatio hämähäkeistä. Kahden eri ohjelmointitavan lisäksi projektissa yhdistettiin kuvataidetta, luonnontiedettä ja lisäksi luki- ja monilukutaitoja.

Tämänkaltaisen projektin voi toteuttaa monista muistakin luonnontieteen aiheista. Tavoitteiden asettelulla voi laajentaa myös kuvataiteen osuutta. Kuvioidun taustan voi värittää esimerkiksi vastavärein tai erilaisin sinisin sävyin. Projektiin voi yhdistää lisää myös matemaattisia tavoitteita esimerkiksi tasokuvioita ohjelmoiden ja piirtäen.

Projektin vaiheet:

1. Parit suunnittelevat ja piirtävät Matatalabin taidepaketin avulla yhdessä ohjelmoiden kuvioita isolle kartongille niin, että niistä muodostuu vähitellen yhtenäinen verkkomainen kuvio.
2. Kun kuvio on tarpeeksi seittimäinen, se väritetään kirkkailla väreillä. Tavoitteena on saada hämähäkille taiteellinen verkko.
3. Taideteoksesta otetaan taustakuva ScratchJr-ohjelmaan. Ohjelmoitavaksi hahmoksi piirretään ScratchJr:ssa hämähäkki.
4. Eri tietolähteiden (tietokirjat, internet) avulla etsitään tietoa hämähäkistä, ja ne kirjaan muistiin.
5. ScratchJr:lla ohjelmoidaan opetusanimaatio, jossa hämähäkkiä koodataan käskyjen ja komentorivien avulla verkon eri paikkoihin. Samalla puhekupien ja äänityksen myötä kerrotaan tietoa hämähäkistä.
6. Lopuksi animaatiot esitetään luokalle.



*Kuva 4.
Valmis robottivälineillä
piirretty taideteos.*

4. Pelastetaan saimaannorppa ja ilves

Tämä monialainen oppimiskokonaisuus oli osana kiinalaisten organisoimaa kansainvälistä ohjelmointikilpailua, jossa tavoitteena oli uhanalaisten eläinten suojeleminen. Ohjelmointiosuus toteutettiin Matatalabin välineillä, mutta ympäristö ja alusta, jossa ohjelmoidut "robotit" liikkuvat, askarrettiin teeman ja eläinten elinympäristön mukaisesti. Aihetta lähestyttiin hyvin oppilaslähtöisesti ja opetussuunnitelmassa (Opetushallitus 2014) olevan monialaisen oppimiskokonaisuuden tavoitteiden mukaisesti. Ryhmätyö toteutettiin ensimmäisessä luokassa kahdessa eri ryhmässä.



*Kuva 5.
Ohjelmointi- ja
askartelutyö
käynnissä.*

Projektin osat ja toteutus:

1. Eläinten valinta

Ensin mietittiin yhdessä, mitä uhanalaisia eläimiä asuu Suomessa. Eläimistä katsottiin myös erilaisia opetusvideoita. Kiinnostavimmista eläimistä äänestettiin kaksi mielenkiintoisinta. Näin ryhmien eläimiksi valikoituivat saimaannorppa ja ilves.

2. Tiedonhankinta

Valitusta eläimistä etsittiin tietoa eri tavoin. Kirjastossa vierailtiin lainaamassa tietokirjoja, joita luettiin yhdessä. Tiedonhankintataitoja harjoiteltiin myös internetin avulla. Teksteistä ja kuvista opetettiin löytämään oleelliset tiedot ja niistä kirjoitettiin muistiinpanoja.

3. Askartelu

Eläimille valmistettiin elinympäristöt askartelemalla, piirtämällä, maalaamalla ja erilaisia

käsityötekniikoita käyttäen. Elinympäristöt mietittiin ja muotoiltiin Suomen vuodenaikojen mukaisesti, ja niihin piirrettiin myös ohjelmointikentät ruutuineen. Lisäksi ohjelmoitavien ”robottien” päälle askarreltiin kyseiset eläimet ja ihmiset, jotka suojelivat eläimiä esimerkiksi kolaamalla lunta saimaannorppien pesätarpeiksi.

4. Ohjelmointi

Ryhmässä yhteistyönä valmistuneet asuinympäristöt toimivat ohjelmointialustoina. Ohjelmoitavia robotteja oli liikkeellä samaan aikaan jopa neljä, jolloin tarvittiin useampi oppilas ohjelmoimaan niitä. Oppilaat miettivät yhdessä hahmojen liikkeitä ja tarvittavat koodaukset niille. Uhanalaisten eläinten auttamisesta tuli kuin kertomus, joka sanoitettiin toimimaan ohjelmointilausekkein. Oli mietittävä, miten esimerkiksi lumena toiminut pum-puli saatiin liikkeelle robotin päällä lumikola kädessä seisovan hahmon toimesta.

5. Videointi

Lopuksi molemmat projektit videoitiin. Ensiksi opeteltiin esittelemään projektit kertomalla ja ohjelmoimalla. Videot kuvattiin GreenScreen -taustalla, jolloin päästiin lähemmäksi suomalaista luontoa ja vaihtuvia vuodenaikoja.

Projekti kokonaisuudessaan oli upean toimiva monialainen oppimiskokonaisuus, johon sisältyi muun muassa kuvataiteen, käsityön, matematiikan ja äidinkielen oppimistavoitteita. Projekti toteutettiin ryhmätyönä, jolloin yhteistoiminnallinen oppiminen oli keskiössä. TVT-taitoja harjoiteltiin kokonaisuuden aikana laajalti. Tiedonhakua internetistä, monilukutaitoa, videointia ja erityisesti ohjelmointitaitoja opittiin yhdessä miettimällä ja kokeilemalla. Ohjelmoinnillinen ajattelu korostui ryhmätyönä tehdyssä useamman Matatalab-robotin liikuttamisessa samaan aikaan johdonmukaisesti toistensa toimintoihin liittyen.

Kansainväliseen kilpailuun osallistui yhteensä yli 200 joukkuetta 11 eri maasta. Jyväskylän normaalikoulusta osallistuneista ryhmistä Saimaannorppa-tiimi palkittiin erityisesti monipuolisuuden, yhteistoiminnallisuuden ja ryhmätyöskentelytaitojen johdosta kymmenen parhaan joukkoon.



QR-koodi norppavideon



Kuva 6. Norppatyön robotit.

Ohjelmoinnillisen ajattelun oppimista laaja-alaisesti

Kuten Uusissa lukutaidoissa ja eNorssin ohjelmointipolussa kuvataan, ohjelmointiosaaminen ja ohjelmoinnillinen ajattelu ovat monipuolisia tieto- ja taitokokonaisuuksia. Esimerkkeinä ohjelmointiosaamisen erilaisista opittavista taitoalueista ovat esimerkiksi oman ohjelmointityön eli ohjelmoitavaksi aiotun ratkaisun suunnittelu ja mallintaminen, joita voidaan tukea luontaisesti esimerkiksi hyödyntämällä animaation taustalle kirjoitettua vuoropuhelua (ks. projekti 1) tai juonikaaviota (ks. projekti 2), antaen algoritmisen rakenteen myöhemmin ohjelmoitavalle varsinaiselle työlle. Itse ohjelmoinnissa ohjelmointikielellä hankitaan puolestaan aktiivisen itse tekemisen kautta perusymmärrystä sääntöpohjaisesta ohjelmoinnista eli vaiheittaisten käskysarjojen eli algoritmien ohjelmoimisesta erilaisilla koodirakenteilla sekä ohjelmointiin kuuluvista perustoimenpiteistä, kuten “debuggaamisesta” eli ohjelmointivirheiden korjaamisesta. Oppimisen kontekstina oleva monialainen autenttinen projekti (esim. peli, keksintö tai animaatio) motivoi oppimaan ja ylläpitää sitoutumista oppimiseen. Ohjelmoimissa voidaan lisäksi oppia erilaisia yksityiskohtaisempia ohjelmoinnillisen ongelmanratkaisun peruseräitä, kuten esimerkiksi algoritmisten prosessien koordinoimista esimerkiksi synkronoimalla rinnakkaisia käskysarjoja kahden ScratchJr-hahmon vuoropuhelua animoimissa. (Fagerlund 2021.)

On kuitenkin tärkeä huomata, kuten tässä artikkelissa kuvatuista esimerkeistä voi havaita, että ohjelmointiosaamisen ja ohjelmoinnillisen ajattelun oppimisen ei tarvitse käytännössä olla erityisen järjestelmällistä tai etukäteen harkittua ohjelmoinnin oppisisältöjen omaksumista. Ohjelmoinnin oppimisen ydinajatuksena on ennen kaikkea tutustuminen sekä ohjelmoituun ympäristöön että ohjelmoinnin luovaan ja yhteistoiminnalliseen tekemiseen erilaisten leikinomaisten harjoitusten ja motivoivien ohjelmointitöiden, kuten animaatioiden, tarinoiden, pelien ja omien luovien keksintöjen kautta (Fagerlund 2021). Ohjelmoinnin oppimisen taustalle usein sijoitetun konstruktionistisen oppimiskäsityksen (ks. esim. Brennan & Resnick 2012) mukaan ohjelmointiin ja ohjelmoinnilliseen ajatteluun kuuluvia tietoja ja taitoja voidaankin harjoitella erilaisin tavoin ja välinein ohjelmoimissa ennen kaikkea vähitellen yhdessä oivaltamalla ja uusia asioita “tarpeen tullen” löytäen.

Vaikka ohjelmointiosaaminen voidaan nähdä tärkeänä oppimisen kohteena jo itsessään, ohjelmointi voidaan myös ymmärtää ennen kaikkea myös tehokkaana työtapana muussa oppimisessa, eli käytännössä mahdollisuutena käsitellä ja harjoitella eri oppiaineiden sisältöjä ja laaja-alaista osaamista. Klassisena esimerkkinä ohjelmoinnista suoraan osana muiden oppiaineiden sisältöihin tutustumista onkin tässä artikkelissa erityisesti projektissa 3 kuvattu käskysarjojen laatiminen geometrinen kuvioiden luomisessa, jota käytettiin esimerkkinä konstruktionistisesta oppimisesta jo 1980-luvulla (Papert 1980). Ohjelmoinnin oppimisen työvälineet, kuten ohjelmoitavat laitteet ja ohjelmointikielet, ovat toki sittemmin kehittyneet samoin kuin ymmärrys ohjelmoinnin pedagogisista mahdollisuuksista (esim. oppiaineintegroinnissa). Muita monenlaisia esimerkkejä on esimerkiksi arkisten toimien, kuten tanssiliikesarjojen esittäminen algoritmisina kokonaisuuksina, kuin myös aiemmassa julkaisussamme (ks. Fagerlund & Roos 2020) kuvatut esimerkit alkuopetuksessa.

Ohjelmoinnin integroimiseen opetussuunnitelman eri alueille voidaan löytää tulevaisuudessa varmasti runsaasti lisää erilaisia vaihtoehtoja, mikäli ohjelmoinnin tarjoamia mahdollisuuksia tarkastellaan ennen kaikkea laaja-alaisen osaamisen valossa, jota tämän artikkelin Kuvio 1 pohjustaa ja jota artikkelissa esitellyt projektit eri tavoin havainnollistavat. Ohjelmointi voi toimia esimerkiksi niin itse oppimisessa kuin eri oppiaineita ja niiden tavoitteita kokoavana osana sekä tuotosten dokumentoinnin eli tiedon esittämisen välineenä, kun oppilaat tuottavat muille esitettävää opiskelumateriaalia ja tekevät työtään näkyväksi. Varsinainen ohjelmointityöskentely (esim. oman robotin ohjelmoiminen, pelin tekeminen) voi ruokkia lisäksi muun muassa ryhmätyöskentelytaitoja ja vuorovaikutustaitoja, luovan suunnittelun ja tuottamisen taitoja ja esimerkiksi pitkäkestoiseen työhön sitoutumista.

Käytännön opetustyössä on kaikkineen tärkeä huomioida, että niin varsinaisten ohjelmointitaitojen harjoittelussa, ohjelmoinnin soveltamisessa työtapana erilaisten tietojen ja taitojen harjoitteluun kuin ohjelmoidun ympäristön tutkimisessäkin ilman ohjelmointia on monipuolisimmillaan kyse sekä tietojen, taitojen, arvojen, asenteiden että tahdon harjoittamisesta. Tällaista kokonaisuutta ei ole mielekäästä "kuitata tehdyksi" kertaluontoisesti, vaan sitä on hyvä ottaa osaksi opetusta pitkin kouluvuosia. Tässä yhteydessä onkin lisäksi tärkeä oivaltaa, kuten Uusissa lukutaidoissakin korostetaan, ettei ohjelmoinnin opiskelun koulussa tarvitse aina sisältää ohjelmoimista itse eli karkeasti ottaen "koodaamisen" opettelua, vaan ohjelmointikasvatus voi sisältää myös ohjelmoidun maailman tutkimista esimerkiksi tunnistamalla ja nimeämällä ohjelmoituja laitteita ja sovelluksia omasta arkiympäristöstä sekä pohtimalla niiden moninaisia käytännöllisiä, sosiaalisia ja eettisiäkin syy-seuraussuhteita (Mertala ym. 2020). Tästä näkökulmasta ohjelmoinnin opiskelu voikin tarjota monenlaisia kytköksiä erityisesti monilukutaitoon ja arjen taitoihin.

Lähteet

Bers, M. U., González-González, C., & Belén Armas Torres, M. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education*, 138, 130–145.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Haettu osoitteesta: <https://www.media.mit.edu/publications/new-frameworks-for-studying-and-assessing-the-development-of-computational-thinking/>

Fagerlund, J. (2021). Teaching, Learning and Assessing Computational Thinking through Programming with Scratch in Primary Schools [Väitöskirja, Jyväskylän yliopisto]. Jyväskylän yliopisto. <https://jyx.jyu.fi/handle/123456789/78190>

Fagerlund, J., & Roos, S. (2020). Motivoivaa ja tavoitteellista oppimista ohjelmointiympäristöjä hyödyntäen. Teoksessa A. Veijola, O.-P. Salo, & S. Roos (toim.) Merkityksellistä oppimista etsimässä - oppimisympäristöjen moninaisuus nyt ja tulevaisuudessa. Jyväskylä: Jyväskylän normaalikoulu, Jyväskylän yliopisto, 135–162.

Ferguson-Patrick, K. (2007). Writers develop skills through collaboration: an action research approach. *Educational Action Research*, 15 (2), 159–180.

Opetushallitus (2014). Perusopetuksen opetussuunnitelman perusteet 2014. Helsinki: Opetushallitus.

eNorssi (2018). Suomen harjoittelukoulujen ohjelmointipolku. Haettu osoitteesta: https://enorssi.fi/wordpress/wp-content/uploads/ohjelmointipolku_08102018.pdf

Mertala, P., Palsa, L., & Dufva, T. S. (2020). Monilukutaito koodin purkajana: Ehdotus laaja-alaiseksi ohjelmoinnin pedagogiikaksi. *Media & viestintä*, 43(1), 21–46.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.