

**Ohjelmoinnin opetus 1-4 lk. matematiikan  
oppimateriaaleissa ja luokanopettajien kokemana**  
Petri Järvelä ja Jasmin Vilmusenaho

Kasvatustieteen pro gradu -tutkielma  
Syyslukukausi 2022  
Kokkolan yliopistokeskus Chydenius  
Jyväskylän yliopisto

## TIIVISTELMÄ

Järvelä, Petri & Vilmusenaho, Jasmin. 2022. Ohjelmoinnin opetus 1-4 lk. matematiikan oppimateriaaleissa ja luokanopettajien kokemana. Kasvatustieteen pro gradu -tutkielma. Jyväskylän yliopisto. Kokkolan yliopistokeskus Chydenius. 120 sivua.

Tämä pro gradu -tutkielma tarkastelee ohjelmoinnin opettamista matematiikan oppimateriaaleissa vuosiluokilla 1-4 ja luokanopettajien kokemana. Lisäksi tarkastellaan, miten toiminnallisuus näyttäytyy ohjelmoinnin opetuksessa. Toiminnallisuutta korostetaan opetussuunnitelmassa monia kertoja eri yhteyksissä ja sen oletetaan näkyvän ja toteutuvan jokaisessa suomalaisessa peruskoulussa. Tutkimuksen teoriaosassa käsitellään ohjelmoinnin ja algoritmisen ajattelun käsitettä ja sitä, miten se näkyy peruskoulun opetussuunnitelman matematiikan tavoitteissa ja sisällöissä. Teoriassa kuvataan toiminnallisuutta ja sen yhteyttä ohjelmointiin.

Tutkimus on monimenetelmäinen. Tutkimuksessa haettiin vastauksia seuraaviin kysymyksiin: Miten ja kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1-4? Miten toiminnallisuus on otettu huomioon ohjelmoinnin opetuksessa matematiikan oppimateriaaleissa vuosiluokilla 1-4? Minkälaisia kokemuksia luokanopettajilla on ohjelmoinnin opettamisesta? Osa tutkimuksen aineistosta kerättiin puolistrukturoidulla teemahaastattelulla haastatteleamalla kahta alakoulun luokanopettajaa, joille toiminnallisuus ohjelmoinnin opetuksessa on tuttua. Haastatteluiden analysointi tapahtui teorialähtöistä sisällönanalyysia käyttäen. Tulokset lajiteltiin teoriasta nousseen teemoittelun mukaan. Toinen osa tutkimuksen aineistosta kerättiin tarkastelemalla matematiikan oppimateriaaleja vuosiluokilla 1-4, joista laskettiin kaikki tehtävät, algoritmista ajattelua kehittävät ja tukevat tehtävät sekä varsinaiset ohjelmointitehtävät.

Tuloksista nousi esille se, että luokanopettajat kokivat matematiikan oppimateriaalit ohjelmoinnin osalta puutteellisina. Oppikirja-analyysi puoltaa opettajien näkemystä, koska varsinaisia ohjelmointitehtäviä on oppikirjoissa suhteellisen vähän. Algoritmista ajattelua kehittäviä ja tukevia tehtäviä löytyy puolestaan runsaasti. Luokanopettajat kuvailivat oppimateriaaleja puutteellisiksi toiminnallisen ohjelmoinnin näkökulmista. Oppilaan oppikirjoissa toiminnallisuutta ei juurikaan ole, mutta opettajan oppaiden tehtävät keskittyvät enimmäkseen toiminnallisuuteen. Luokanopettajien kuvauksissa toiminnallinen ohjelmointi opetuksessa ilmeni muun muassa ongelmanratkaisuna ja ryhmätöiden tekemisenä, missä oppilas on aktiivisessa roolissa. Opettajat myös korostivat algoritmisen ajattelun taidon merkitystä ohjelmoinnin opetuksessa. Luokanopettajat kertoivat hyödyntävänsä opetuksessaan muita toiminnallista ohjelmointia tukevia materiaaleja ja oppimisympäristöjä, kuten robotiikkaa. Tukea toiminnallisen ohjelmoinnin toteuttamiseen luokanopettajat saivat koulutuksista, mutta omatoiminen perehtyminen aiheeseen on ollut avainasemassa.

Asiasanat: ohjelmointi, algoritmisen ajattelu, toiminnallinen oppiminen, oppimateriaalitutkimus

Tämän julkaisun alkuperäisyys on tarkistettu Turnitin Originality Check -ohjelmalla.

# SISÄLTÖ

<b>TIIVISTELMÄ</b> .....	<b>2</b>
<b>SISÄLTÖ</b> .....	<b>3</b>
<b>1 JOHDANTO</b> .....	<b>5</b>
<b>2 TOIMINNALLISUUS OPPIMISEN TUKENA</b> .....	<b>8</b>
2.1 Oppiminen ja sosiokonstrukttiivinen oppimiskäsitys.....	8
2.2 Toiminnallisuus ja toiminnallinen oppiminen .....	10
<b>3 OHJELMOINTI JA ALGORITMINEN AJATTELU</b> .....	<b>13</b>
3.1 Ohjelmoinnista yleisesti .....	13
3.2 Algoritmi .....	14
3.3 <i>Computational thinking</i> -käsite.....	16
3.4 Algoritminen ajattelu .....	18
3.5 Ohjelmointi ja toiminnallisuus perusopetuksen opetussuunnitelman perusteissa .....	20
3.6 Toiminnallisuus ohjelmoinnissa .....	28
<b>4 OPPIMATERIAALEISTA JA NIIDEN TUTKIMUKSISTA</b> .....	<b>30</b>
<b>5 TUTKIMUSTEHTÄVÄ JA -KYSYMYKSET</b> .....	<b>34</b>
<b>6 TUTKIMUKSEN TOTEUTTAMINEN</b> .....	<b>35</b>
6.1 Tutkimuksen metodologiset valinnat .....	35
6.1.1 Mixed methods .....	35
6.1.2 Kvalitatiivinen osuus .....	35
6.1.3 Kvantitatiivinen osuus.....	37
6.2 Aineistonkeruu ja tutkimusaineistot.....	38
6.3 Aineiston analyysi .....	53
6.4 Eettiset ratkaisut.....	57

<b>7</b>	<b>TULOKSET</b> .....	<b>59</b>
	7.1 Tuloksien taustaa .....	59
	7.2 Miten ja kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4? .....	59
	7.3 Miten toiminnallisuus on otettu huomioon ohjelmoinnin opettamisessa? .....	76
	7.4 Minkälaisia kokemuksia luokanopettajilla on ohjelmoinnin opettamisesta? .....	79
<b>8</b>	<b>POHDINTA</b> .....	<b>87</b>
	8.1 Tulosten tarkastelu ja johtopäätökset.....	87
	8.2 Tutkimuksen arviointi.....	90
	8.3 Jatkotutkimusaiheita.....	91
	<b>LÄHTEET</b> .....	<b>93</b>
	<b>LIITTEET</b> .....	<b>101</b>

# 1 JOHDANTO

Ohjelmoinnin merkitys nyky-yhteiskunnan alati kehittyvässä tietoteknisessä ympäristössä on hyvin suuri. Meitä ympäröivässä maailmassa kohtaamme yhä useammin ohjelmointia tiedostamattakin. Esineiden internet (*Internet of Things*) kasvattaa suosiotaan ja tämän myötä ohjelmointikoodia löytyy mitä erikoisimmista paikoista. Kotiemme laitteet, kuten jääkaapit, ilmastointilaitteet ja lämmitysjärjestelmät, ovat nykyään usein kytkettyinä tietoverkkoon ja sisältävät jonkinlaista tekoälyä, jota ohjaa ihmisen kirjoittamat ohjelmat. Esimerkiksi älykellot, autot ja kodin valaistus voivat olla kytkettyinä internetiin, jolloin laitteiden etäohjaus ja -seuranta on mahdollista. Laitteet antavat tietoa erilaisten antureiden avulla ja automaattista tiedonsiirtoa tapahtuu hyvin paljon huomaamattamme. (vrt. Rose, ym. 2015.) Kaiken tämän on mahdollistanut kehittynyt ohjelmointiosaaminen.

Meidän tutkijoiden kiinnostus tutkittavaa aihetta kohtaan kumpuaa aikaisemmista kandidaatin töistä ja edellisten koulutusten myötä. Petri on tietoliikennetekniikan diplomi-insinööri, jota kautta hän on saanut myös koulutusta ohjelmointiin. Petrin kandidaatin tutkielma käsitteli ohjelmoinnin ja matematiikan osaamisen välistä yhteyttä. Petrillä on myös lähes kymmenen vuoden työkokemus luokanopettajana, jolloin hän on myös opettanut ohjelmointia alakoulussa usean vuoden ajan. Jasmin käsitteli kandidaatin työssään joukkuevoimistelijoiden liikunnan vaikutusta heidän opiskelumotivaatioonsa, joten kiinnostus tehdä tutkimusta toiminnallisesta oppimisesta oli vahva. Jasminilla on myös kokemusta luokanopettajan työstä sekä lyhyemmissä että pidemmissä sijaisuuksissa, joissa hän on myös päässyt hieman opettamaan ohjelmointia. Koimme, että meitä kiinnostavat aiheet saisi yhdistettyä samaan tutkielmaan ja tutkielman tekeminen yhdessä innosti meitä molempia.

Tutkimuksemme tarkoituksena on selvittää, miten ohjelmointia opetetaan kahden suomalaisen kustannusyhtiön Otava Oy:n ja Sanoma Pro Oy:n matematiikan oppimateriaaleissa vuosiluokilla 1–4, ja kuinka paljon ne sisältävät

ohjelmointiin liittyviä ja siihen läheisesti liittyviä algoritmista ajattelua tukevia ja kehittäviä tehtäviä ja harjoitteita sekä toiminnallisia tehtäviä. Lisäksi tarkoituksena on selvittää, miten oppimateriaalit tukevat luokanopettajien opetustyötä ja minkälaisia käsityksiä ja kokemuksia heillä on ohjelmoinnin opettamisesta alakoulussa. Olemme kiinnostuneita myös siitä, kuinka toiminnallisuus on huomioitu ohjelmoinnin opetuksessa ja miten sitä hyödynnetään.

Koska tietotekniikasta ja teknologiasta on tullut pysyvä osa arkeamme, on myös koulutuksemme vastattava siihen. Ohjelmointia ja siihen liittyvää algoritmista ajattelukykyä pidetään yhtenä oleellisena taitona lasten ja nuorten opetuksessa (mm. Bocconi ym., 2018). Ohjelmointiin on kiinnitetty enemmän huomiota peruskoulutasolla myös Suomessa nykyisen opetussuunnitelman myötä. Opetussuunnitelman (Opetushallitus, 2014) mukaan ohjelmointia opetetaan jo alakoulun alemmilta luokilta saakka. Ohjelmointia ei opeta omana oppiaineena, vaan se on lähes kokonaan integroitu luontevasti matematiikan oppiaineen sisältöihin ja tavoitteisiin. Matematiikan oppimateriaaleista löytyy ohjelmointiin liittyviä tehtäviä ja opetusta ohjaavia tapoja ja vinkkejä vaihtelevin määrin. Varsinaisten ohjelmointitehtävien lisäksi oppimateriaaleissa on runsaasti myös ohjelmointia tukevia ja sen vaatimaa ajattelutapaa eli algoritmista ajattelua kehittäviä tehtäviä, kuten ongelmanratkaisutehtävät ja loogista päättelyä vaativat tehtävät sekä toimintaohjeen noudattamista vaativat tehtävät. Opetussuunnitelmankaan tavoitteena ei ole luoda kaikista lapsista ohjelmoijia, vaan pikemminkin kehittää ajattelun taitoja, joiden avulla ohjelmoinnin ymmärtäminen, toteuttaminen ja kehittäminen olisi mahdollista.

Toiminnallisuus opettamisen tukena edesauttaa ja vahvistaa oppimista (mm. Dewey, 1957). Nykyisen perusopetuksen opetussuunnitelman mukaisen sosiokonstruktivistisen oppimiskäsityksen mukaan opiskelijan rooli on olla vuorovaikutuksessa muiden kanssa ja osallistua yhteiseen toimintaan (vrt. Kauppila, 2007, 35). Toiminnallisuus ohjelmoinnin opettamisessa voi esiintyä esimerkiksi kehollisen toiminnan kautta, yhteistoiminnallisesti tai robotiikan välityksellä. Robotiikan hyödyntäminen ohjelmoinnin opettamisessa

havainnollistaa ja konkretisoi oppijan algoritmisen ajattelun kulkua.

Tutkimuksemme osoitti, että ohjelmoinnin opetuksessa on kehitettävää esimerkiksi oppimateriaalien ja opettajien koulutuksen näkökulmista. Oppimateriaaleissa varsinaisia ohjelmointitehtäviä on vähäisesti ja oppilaan oppikirjoista toiminnallisuutta ei juurikaan löydy. Myös luokanopettajien koulutukseen olisi syytä lisätä enemmän ohjelmointikoulutusta, jotta opettajilla olisi valmiuksia opettaa sitä monipuolisemmin. Nyt ohjelmointikouluttautuminen jää paljolti luokanopettajan oman kiinnostuksen ja aktiivisuuden varaan.

## 2 TOIMINNALLISUUS OPPIMISEN TUKENA

### 2.1 Oppiminen ja sosiokonstrukttiivinen oppimiskäsitys

Konstrukttiivisen oppimiskäsityksen perustana ovat tieteenteoria ja positiivinen psykologia, joiden perusajatuksia edustavat Kant, Piaget ja Baldwin. Konstruktivismissa korostetaan oppilaan aikaisempien tietorakenteiden merkitystä ja omia kokemuksia oppimisessa. Siinä oppilas konstruoi käsityksiään tapahtumista ja ilmiöistä omiin kokemuksiinsa perustuen ja aikaisempien konstruoimiensa mallien pohjalta. Oppiminen nähdään aktiivisena rakentamisena, tietojen tarkentamisena ja uudelleenmuotoiluna. Opettajan rooli konstruktivisessa oppimisessa on luoda oppimiselle ympäristö, jossa oppilaalla on mahdollisuus tarkastella aikaisempia kokemuksiaan ja prosessoida tietoa aktiivisesti sekä käsitellä uutta tietoa mielekkäässä asiayhteydessä. (Sahlberg & Leppilampi, 1994, 24–25.)

Filosofi ja kasvatustieteilijä John Deweyä pidetään kokemuksellisen oppimisen perustajana. Hän pyrki 1900-luvun alussa yhdistämään tiedollisen ja käytännöllisen opetuksen niin, että opetuksessa yhdistyisivät tekeminen, ajattelu ja tarkoitus. Edistyksellisessä opetuksessa tulisi saada yhteys oppijan henkilökohtaiseen kokemukseen, joka on edellytys tehokkaaseen oppimiseen. Tärkeää kokemuksen hankkimisessa on sen laatu ja sen positiivinen vaikutus tuleviin kokemuksiin. Jokainen kokemus tulisi hyväksyä sellaisenaan ja kokemuksella pitäisi olla vaikutus myöhempisiin kokemuksiin. Dewey korostaa kokemuksen jatkuvuuden merkitystä, jossa aiemmilla kokemuksilla on vaikutusta tulevien kokemusten muotoutumiseen. (Dewey 1938, 27–28, 35; Öystilä 2003, 27–29.) Dewey (1957, 7–8) kuvailee oppimista ongelmien ratkaisuna ja uusien kokemusten hankkimisena. Paras mahdollinen koulu lapselle on sellainen, jossa hän oppii omien toimintojensa ja kokemustensa avulla. Deweyn kasvatustieteiden filosofia korostaa kasvatustyössä sosiaalisuuden ja toiminnallisuuden periaatetta. Deweyn mukaan ajattelemalla ja toimintaa refleктоimalla tapahtuu toiminnallista oppimista (Dewey, 1938, 38). Myös Sahlberg ja Leppilampi (1994,



25) korostavat yhteistoiminnallisen oppimisen tärkeyttä ja toimivuutta, jonka avulla oppilaan tiedonmuodostus vahvistuu ja tiedollisia ristiriitoja syntyy. John Deweyn ajatukset pohjautuvat siihen, että oppilas on oppimisprosessissa aktiivisessa roolissa ja rekonstruoi aiemmin opittuja tietojaan ja käsityksiään, joista muodostuu oppimisen ydin. Ongelmatilanteiden kautta syntyy uutta tietoa sekä tietämystä ja se on yksi parhaista oppimiskeinoista. (Kauppila, 2007, 34; Sahlberg & Leppilampi, 1994, 25.) Dewey kuvailee ei-reflektiivisiä, vakiintuneisiin toimintatapoihin ankkuroituvia kokemuksia hallitsevina. Hän pitää tärkeänä sitä, että reflektiivisessä oppimisessä lähtökohtana ei ole ainoastaan kokemus, vaan totuttujen toimintatapojen hajoamisesta johtuva ongelmatilanteiden ratkaiseminen kokeellisesti. Kasvatuksen ihanteellinen tavoite on toimia itseohjautuvasti, jota toiminnallisuus ja reflektio edistävät yhdessä. (Öystilä, 2003, 31.)

Sosiokonstruktivistista oppimiskäsitystä on tutkittu laajasti ja se on kehittynyt konstruktivistisesta oppimiskäsityksestä ja kognitiivisesta oppimisesta tehdyn tutkimustyön myötä. Konstruktivistisen ihmiskäsityksen mukaan ihminen nähdään sosiaalisena olentona, jossa yhteiskunnan ja kulttuurin vaikutuksen merkitystä korostetaan ihmisen käyttäytymiseen. Kognitiivinen ihmiskäsitys taas korostaa ihmistä tiedon käsittelijänä. Sosiokonstruktiiivinen oppimiskäsitys on muodostunut näiden ihmiskäsitysten myötä ja se perustuu humanistiseen ihmiskäsitykseen. (Kauppila, 2007, 12.) Kauppilan (2007, 35) mukaan sosiokonstruktivismissa olennaista on se, että tietoa rakennetaan sosiaalisesti ja kognitiivisesti. Opiskelijan rooli on olla vuorovaikutuksessa muiden kanssa ja osallistua yhteiseen toimintaan.

Sahlbergin ja Leppilammen (1994, 24) mukaan hyvä ja tehokas oppiminen on konstruktiiivista eli oppiminen tapahtuu aktiivisesti, oppilaan omakohtaisen ymmärtämisen ja merkitysten rakentamisen kautta. Hyvän oppimisen lähtökohtina ovat myös itseohjautuvuus ja kumulatiivisuus. Itseohjautuvuudella tarkoitetaan oppilaan kykyä ymmärtää ja hallita omaa toimintaansa, esimerkiksi oman oppimisen suunnittelua ja etenemistä tavoitteitaan kohti. Kumulatiivisesta oppimisesta puhutaan silloin, kun uuden oppiminen perustuu aiemmin

oppituihin tietoihin ja taitoihin. Opettajalla täytyy olla tieto siitä, mitä oppilas jo tietää, jotta tietorakenteita voi kasvattaa. Hyvään ja tehokkaaseen oppimiseen liittyy myös tavoitesuuntautuneisuus, tilannesidonaisuus ja yhteistoiminnallinen oppiminen.

Perusopetuksen opetussuunnitelman perusteissa (Opetushallitus, 2014, 17) oppilas nähdään aktiivisena toimijana opetuksessa. Oppilaan tehtävä on asettaa tavoitteita itselleen ja ratkaista ongelmia yhdessä muiden kanssa ja itsenäisesti. Kehollisuutta ja monipuolista aistien käyttöä korostetaan hyödyntämään opetuksessa. Oppimisessa pyritään saavuttamaan oppimisen iloa ja positiivisia tunnekokemuksia. Oppimista kuvaillaan muun muassa ajatteluna, yhdessä ja yksin tekemisenä, suunnitteluna sekä tutkimisena. Yhteistoiminnallinen työskentely nähdään mahdollistavan luovan ja kriittisen ajattelun, mikä kehittää ongelmanratkaisutaitoja. Erilaisten tietojen ja taitojen oppiminen on kumuloituvaa ja vaatii ahkeraa ja säännöllistä harjoittelua.

## **2.2 Toiminnallisuus ja toiminnallinen oppiminen**

Toiminnan kautta oppiminen on yksi pitkään tiedossa ollut oppimismenetelmä, ja opetuksen monenlaiset toiminnalliset työtavat ovat olleet vahvasti esillä jo pitkän aikaa. Käsitteenä toiminnallisuus on haastava rajata, sillä moniin työtapoihin liittyy erilaisia mahdollisuuksia toiminnalliseen aktiivisuuteen. Toiminnallisuus nähdään laajana käsitteenä monenlaisille toiminnallisille lähestymistavoille. (Vuorinen, 1993, 179–180.) Heikkisen (2005, 38) mukaan toiminnallisilla työtavoilla on sama idea, jossa tekemisellä on jokin tavoite ja yhteys suurempaan kokonaisuuteen. Kun oppimiseen liittyy toimintaa, se on helpommin sisäistettävissä.

Toiminnallisella opetuksella voidaan tarkoittaa sosiaalista, fyysistä tai kognitiivista aktiivisuutta. Toiminnallisessa opetuksessa huomioidaan erilaiset oppimistyylit, joiden avulla pyritään saavuttamaan itseohjautuvuutta, rakentavaa vuorovaikutusta, osallisuuden sekä yhteisöllisyyden tunnetta. (Saaranen-Kauppinen ym., 2018.) Koskenkarin (2012) mukaan

toiminnallisuudella tarkoitetaan aktiivista toimintaa, johon yhdistyy osallisuus, ajatuksellisuus, kokemuksellisuus ja vuorovaikutus.

Norrena (2016, 14) kuvailee toiminnallista oppimista oppilaan aktiivisena toimimisena ja ajattelemisena oppimisprosessin aikana. Toiminta on vuorovaikutteista ja oppimisprosessissa oppilas saa kokemuksia ja elämyksiä sekä oivalluksia yhdessä toisten oppilaiden ja ohjaajien kanssa. Toiminnan ja leikin kautta oppiminen on luonnollista erityisesti alkuopetusikäiselle lapselle. Norrenan (2016) mukaan myös Fisher ja Madsen (1984) toteavat, että oppilas on sitoutunut tekemiseen, kun hän pääsee käsittelemään konkreettista materiaalia tai pääsee seuraamaan aktiivisesti opettajan toimintaa konkreettisilla välineillä. Lapsi on myös helpompi saada sitoutumaan toimintaan, kun opettaja ja lapsi toimii yhdessä ja tekeminen sisältää tutkimista, etsimistä ja eläytymistä esimerkiksi satuihin ja tarinoihin.

Norrena (2016, 14; Koskenkari, 2012) kuvailee toiminnallisiksi työtavoiksi tutkimukset, projektityöt, ryhmätyöt, leikit ja yhteistoiminnallisen oppimisen. Myös draama, kuvittaminen sekä digitaaliset ja taiteelliset esitykset ovat toiminnallisen oppimisen muotoja. Toiminnallista opetusta hyödynnetään monipuolisesti fyysisen toiminnan lisäksi oppimistavoitteiden saavuttamisessa. Rainio (2002, 195) mainitsee, että toiminnallisia työtapoja hyödyntäessä on tärkeää varata sille reilusti aikaa ja tilaa, jotta opetuksen kokonaisuus säilyy, eikä osat jää irrallisiksi. Perinteisiin työtapoihin verrattuna toiminnallisissa työtavoissa abstraktit ilmiöt konkretisoidaan ulkoisiksi tapahtumiksi. Toiminta on tarkoitus muuttaa sellaiseksi, mikä on mielekästä ja rakentavaa sekä tilanteeseen sopivaa, mutta painottuu kinesteettiseen ja motoriseen toimintaan. (Sura, 1999, 222–227.) Konstruktivistisessä oppimiskäsityksessä opettaja ei siirrä tietoa, vaan toimii oppimisprosessissa ohjaajana. Opettajan tehtävä on saada oppilaat ymmärtämään opittava asia. Opettajan täytyy hallita keinot esittää opetettava asia niin, että sen ymmärtäminen oppilaalle on helpompaa. Tärkeää on ottaa oppilaan ikäkausi huomioon opittavaa asiaa opettaessa ja ohjatessa. (Järvinen, 2011, 36–37, 222.) Toiminnallisessa oppimisessä opettajan tehtävä on ohjata oppimista eikä vain jakaa tietoa oppilaille. Opettajan roolia luokassa

voitaisiin kuvailla ohjaavaksi ja mentoroivaksi, mikä tukee oppilaslähtöistä työskentelyä.

Luukkaisen (2004, 95) mukaan toiminnallisessa työskentelyssä opettajan tehtävä on suunnitella oppimisympäristö ja toimia oppimisen suunnittelijana huomioiden sen, että oppilaat ovat opetuksessa tekijöitä. Työskentelyssä olisi tärkeää, että opettaja innostaisi ja motivoisi oppilaita aktiiviseen työskentelyyn sekä asenteellaan että omalla esimerkillään.

## 3 OHJELMOINTI JA ALGORITMINEN AJATTELU

### 3.1 Ohjelmoinnista yleisesti

Ohjelmointi on läsnä meidän jokapäiväisessä elämässämme, vaikkemme sitä aina näe tai tiedosta. Ohjelmointia käytetään monissa arkielämän tilanteissa, joissa tehokkuus ja automaatio ovat tärkeässä roolissa. Liikennevalot, pelit, hissit ja lukuisat muut toiminnot kätkevät sisäänsä lukuisia koodirivejä suoritettavaa ohjelmaa. Maailmassa olevaa koodimäärää ja verkkoliikennettä on lisännyt huomattavasti myös esineiden internetin yleistyminen, kun erilaiset laitteet sensoreineen välittävät tietoa automaattisesti taustalla (vrt. Rose, ym. 2015).

Ohjelmoitava laite, esimerkiksi tietokone, ei itse osaa ajatella, vaan se suorittaa nopeasti ja tehokkaasti ihmisen sille antamia käskyjä eli komentoja. Ohjelmoitaessa toimintaohje eli algoritmi on määriteltävä tarkasti ja täsmällisesti, sillä ohjelmoitava laite osaa suorittaa vain sen tuntemia komentoja. Ohjelmoitavalle laitteelle voidaan syöttää kolmenlaisia komentoja. Komennot voivat olla algoritmin etenemistä ohjaavia, laskutoimituksia suorittavia tai ohjelmoitavan laitteen ominaisuuksia hyödyntäviä. Algoritmia suorittava laite suorittaa komentoja lause kerrallaan niiden esitysjärjestyksessä. Suoritusjärjestystä voidaan muuttaa erityisillä ehto- ja toistolauseilla. Ehtolauseet mahdollistavat vaihtoehtoisia suoritustapoja, mikäli lauseelle asetettu ehto täyttyy. Ehtolauseen jälkeen suorittaminen voi joko päättyä tai jatkua ehtolauseen jälkeisestä kohdasta. Toistolauseetta käytetään, mikäli halutaan toistaa useamman kerran peräkkäin samaa lausetta – niin kauan kuin annettu ehto on voimassa. Ohjelmoitavan laitteen antama lopputulema riippuu usein ihmisen sille antamista syötteistä (*input*), joista ohjelma suorittaa laskutoimituksia ja algoritmia sekä hyödyntää laitteen ominaisuuksia. Syötteenä voi toimia esimerkiksi näppäimistöltä tai kosketusnäytöltä annettu komento. Ohjelman antama lopputulos on algoritmin suorituksesta saatu tuloste (*output*), joka voi olla esimerkiksi näytölle tulostuva teksti. (Larsson ym., 2021, 5–6.)

## 3.2 Algoritmi

Nykyään algoritmin käsite ymmärretään usein liittyvän tietojenkäsittelytieteisiin. Algoritmit ovat olleet matematiikan alalla jo kuitenkin pitkään käytössä. Matemaatikot ovat tutkineet laskemista ja algoritmeja jo babylonialaisten aikaan, tiettävästi jo 300 vuotta ennen ajanlaskun alkua (Soaren, 1999, 4.) Algoritmi-nimitys prosessille, joka kuvaa tarkasti määriteltyä mahdollisesti peräkkäiseen toistoon perustuvaa toimintaohjetta, on peräisin arabialaiselta matemaatikolta Al-Khwarizmilta. Al-Khwarizmi on merkittävin matemaatikko, joka on vaikuttanut intialais-arabialaisen numerojärjestelmän leviämiseen sekä algebran kehittymiseen ja leviämiseen. (Luoma-aho, 2010.)

Matemaattisten algoritmien kehittyessä alkoi myös laskennallisten koneiden kehittäminen. Ensimmäisiä keksintöjä olivat 1600-luvulla ranskalaisen Blaise Pascalin yhteenlaskukone ja saksalaisen Gottfried Wilhelm Leibnizin kehittyneempi laskin, joka osasi myös kertoa ja jakaa. Leibnizin oivalluksena oli binaarijärjestelmän paremmuus koneessa verrattuna desimaalijärjestelmään. 1800-luvun alkupuolella Babbage kehitti idean analyyttisestä koneesta, jolla pystyi suorittamaan pitkiäkin mekaanisia laskutoimituksia. Brittiläinen matemaatikko ja loogikko ideoi vuonna 1936 mallin Turingin koneesta, jolla pystyisi suorittamaan jonkin tietyn matemaattisen operaation tai algoritmin (Soaren, 1999, 4.)

Kiinnostus matemaattisiin ongelmakysymyksiin ja niiden tehokkaaseen ratkaisemiseen kasvoi ja samalla teoreettisten algoritmien suorittaminen tehokkaalla laskennalla sai aikaan myös algoritmikäsitteen yleisemmän käytön. Fantin (1993, 3) mukaan tämä tapahtui venäläisen Andrei Markovin vuonna 1954 julkaiseman kirjan *Teoriya Algorifmov* myötä. Englanniksi käännettynä Markov toteaa ensimmäisessä virkkeessään:

In mathematics, "algorithm" is commonly understood to be an exact prescription, defining a computational process, leading from various initial data to the desired result. (Markov, 1954)

Saman suuntaisesti toteaa esimerkiksi Jeannette Wing (2010) julkaisussaan *Computational Thinking: What and Why?*: algoritmi on abstraktio prosessista, joka vastaanottaa syötteitä, suorittaa sekvenssin toimintoja ja tuottaa tuloksia halutun tavoitteen saavuttamiseksi. Knuthin (1997, 4–6) mukaan algoritmilla on viisi ominaisuutta:

1. *Finiittisyys*: Algoritmin tulee aina päättyä äärellisen askelmäärän jälkeen.
2. *Definiittisyys*: Jokaisen algoritmin askeleen tulee olla täsmällisesti ja yksiselitteisesti määritelty.
3. *Syöte*: Määritellystä kohdejoukosta otetut suureet ennen algoritmin aloittamista.
4. *Tuloste*: Suureita, joilla on määritelty suhde syötteisiin.
5. *Tehokkuus*: Kaikkien suoritettavien toimintojen on oltava riittävän perusluonteisia, jotta ne voidaan periaatteessa suorittaa tarkasti ja rajallisessa ajassa ihmisen toimesta paperilla ja kynällä.

Fantin (1993, 4) listaus tyypillisestä tietojen käsittelyn algoritmista on hyvin samankaltainen:

1. Algoritmin täytyy olla askel askeleelta etenevä sekvenssi toimintoja.
2. Algoritmin jokainen askel täytyy olla täsmällisesti määritelty.
3. Algoritmin täytyy päättyä äärellisen askelmäärän jälkeen.
4. Algoritmin täytyy tuottaa tehokkaasti oikeita ratkaisuja.
5. Algoritmin täytyy tuottaa aina sama ratkaisu samalla syötteellä.

Peruskoulun matematiikan oppitunneilla harjoitellaan yksinkertaisia algoritmeja, kuten yhteen- ja vähennyslaskualgoritmeja, joissa laskujen ratkaisut saadaan noudattamalla tiettyjä toimintavaiheita oikeassa järjestyksessä. Algoritmeissa esiintyy myös toistoa ja yksinkertaisia ehtoja, joita myös ohjelmoinnissa esiintyy ja tarvitaan.

### 3.3 *Computational thinking* -käsite

Usein ajatellaan, että ohjelmointi ja siihen liittyvä ajattelumaailma olisi vain ohjelmistoteollisuuden ja koodareiden tarvitsemaa käsitteistöä. Jeannette Wingin (2006) mukaan ohjelmoinnillisten taitojen oppiminen on hyödyllistä kaikille. Kekäläinen (2015, 27) toteaa, että ohjelmoinnin opetuksen suosituksia ja ohjelmoijien määrän tarvetta tulevaisuudessa usein kyseenalaistetaan. Onkin tärkeämpää puhua ohjelmoinnin yhteydessä abstraktimmasta taidosta, jota opitaan käytännössä tietokoneohjelmoinnin kautta. Matemaatikko ja tietojenkäsittelytieteilijä Seymour Papert (1980) mainitsi jo vuonna 1980 ensimmäisenä termin *computational thinking* Mindstorms-kirjassaan, joka käsitteli ohjelmoinnillista ajattelua tietokoneohjelmointia laajempänä käsitteenä. Toinen maininta *computational thinking*-termistä Papertilta (1996) löytyy hänen julkaisustaan ”*An Exploration in the Space of Mathematics Educations*” vuodelta 1996. Termin käyttö yleistyi vasta kuitenkin Wingin (2006) julkaistua artikkelinsa *Computational Thinking (CT)*, jossa hän korostaa *computational thinking* -taitojen merkittävyyttä ja hänen mukaansa ne tulisi rinnastaa osana lukemisen, kirjoittamisen ja aritmeettisten taitojen kanssa jokaisen lapsen perustaidoiksi (Wing, 2006, 33). Wing (2006) ennakoi, että tietokoneita on tulevaisuudessa niin helposti saatavilla, että jokaisen on mahdollista käyttää niitä ongelmien ratkaisemiseen. Tällöin on tärkeää ymmärtää, mikä on tietokoneistettavissa ja automatisoitavissa.

CT-termiin viitataan usein erilaisin ilmauksin eri kielissä. Englanniksi käännettyinä termeinä esiintyy muun muassa *algorithmic thinking*, *coding*, *programming*, *informatics thinking*, *technological literacy* ja *language of technology*. Erilaisten termien käyttöä perustellaan esimerkiksi näkökulman valinnalla ja termin ymmärrettävyydellä. *Computational thinking* mielletään myös usein liian viralliseksi opetuslalle. (Bocconi ym., 2018, 22–23.)

Englanninkielisestä termistä *computational thinking* käytetään useita erilaisia suomennoksia. Suoraan käännettynä termi olisi ”laskenta ajattelu”. Tämä kuitenkin on jo kauppatieteissä laskentatoimessa käytössä eikä ole tarpeeksi kuvaava ohjelmoinnin puolelle. (Kekäläinen, 2015, 28.) Kekäläisen (2015)



mukaan suomennos ei vastaa kovin hyvin Jeannette Wingin (2006) ajatusta termin sisällöstä ja laajuudesta koskien ajattelutapaa ja ajattelutaidon merkityksestä. Kekäläinen (2015, 29) ehdottaa termin suomennokseksi automatisointiajattelu, jolla hän viittaa siihen, että ohjelmoinnissa luodaan uusia toistettavia prosesseja ja automaatiota. Lisäksi Kekäläinen (2015) perustelee suomennosta sillä, että se ei ole käytössä muussa tarkoituksessa ja että kaikki, mitä voidaan ohjelmakoodilla tehdä, on myös automatisoitua.

Liukas ja Mykkänen (2014) käyttävät termistä suomennosta ohjelmoinnillinen ajattelu Koodi2016-oppaassa, jossa he kertovat opettajille ja opetusalan päättäjille, miksi ohjelmointi on tärkeää ja miten sitä voisi opettaa. He mainitsevat myös termin algoritminen ajattelu. Kekäläisen (2015) mukaan suomennetut termit ohjelmoinnillinen ajattelu tai ohjelmointiajattelu tuntuvat kömpelöiltä ja ne viittaavat liiaksi pelkkään ohjelmointiin. Ohjelmoinnillinen ajattelu on kuitenkin paljon käytössä. Esimerkiksi Luostarinen ja Peltomaa (2016) käyttävät termiä ohjelmoinnillinen ajattelu opettajille suunnatussa Reseptit OPSin käyttöön -oppaassa. Myös useat Suomen yliopistot, kuten Jyväskylän yliopisto ja Turun yliopisto, käyttävät termiä ohjelmoinnillinen ajattelu omissa opintojaksojen nimissä opetussuunnitelmissaan (Jyväskylän yliopisto, 2021; Turun yliopisto, 2021). Opetus- ja kulttuuriministeriön Oikeus oppia -kehittämishjelman osa Uudet lukutaidot käyttää materiaaleissaan niin ikään termiä ohjelmoinnillinen ajattelu yhdessä loogisen ajattelun kanssa (Uudet lukutaidot, 2020).

Perusopetuksen opetussuunnitelman perusteissa (Opetushallitus, 2014) käytetään termiä algoritminen ajattelu ja se on sisällytetty osaksi matematiikan oppiainetta. Vuosiluokkien 7–9 matematiikan tavoitteisiin liittyvässä keskeisessä sisältöalueessa on maininta ”Syvennetään algoritmista ajattelua.” Matematiikan tavoitteessa T20 sanotaan: ”ohjata oppilasta kehittämään algoritmista ajatteluaan sekä taitojaan soveltaa matematiikkaa ja ohjelmointia ongelmien ratkaisemiseen.” Päätötodistuksessa on yhtenä arvioinnin kohteena matematiikassa algoritminen ajattelu ja ohjelmointitaidot. Opetushallituksen rahoittama ja LUMA-keskus Suomi -verkoston toteuttama LUMATIikka-

ohjelma käyttää myös suomennosta algoritminen ajattelu. Algoritminen ajattelu on terminä käytössä myös kansainvälisesti (Bocconi ym., 2016, 22–23).

Tässä tutkielmassa *computational thinking* -termin suomenkieliseksi vastineeksi on valittu termi algoritminen ajattelu, koska mielestämme se kuvaa hyvin ohjelmoinnissa tarvittavaa ajattelutapaa. Valintaa perustelee myös tutkimuksemme aihe tutkia toiminnallisuutta ohjelmoinnin opetuksessa, jolloin termi ei liikaa korosta ohjelmointia, johon yleensä liitetään tietokoneet. Lisäksi on luontevaa käyttää termiä algoritminen ajattelu, koska se on käytössä myös perusopetusta ohjaavissa asiakirjoissa.

### 3.4 Algoritminen ajattelu

Algoritminen ajattelu on ongelman ratkaisemiseen tai tehtävän suorittamiseen tarvittavien toimenpiteiden ja niiden suoritusjärjestyksen määrittämistä (Larsson ym., 2021, 17). Algoritmisen ajattelun taito nykyisessä teknologisessa maailmassa on tärkeä ja Yadav, Hong ja Stephenson (2016, 565) esittävätkin sen olevan yksi oppilaiden keskeisimmistä 2000-luvun taidoista. Algoritminen ajattelu on yksi analyttisen ajattelun muodoista ja sillä on paljon yhteistä muun muassa matemaattisen ajattelun kanssa, kuten esimerkiksi ongelmanratkaisutaitojen vaatimaan matemaattiseen ajatteluun (Bers 2010, 3).

Futschek (2006, 160) määrittelee algoritmisen ajattelun koostuvan erilaisista kyvyistä, jotka muodostavat ymmärryksen algoritmeista. Näitä ovat kyky analysoida ongelmaa ja määrittellä se täsmällisesti, kyky keksiä perusmekanismi tiettyyn ongelmaan, kyky rakentaa algoritmi perusmekanismia käyttäen ongelman ratkaisemiseksi, kyky ymmärtää ongelmaa erilaisissa tilanteissa ja kyky kehittää algoritmin tehokkuutta.

Liukas ja Mykkänen (2014, 76–78) ovat jakaneet opettajille suunnatussa oppaassaan algoritmisen ajattelun neljään osa-alueeseen, joita ajattelutaito vaatii pystyäkseen ratkaisemaan ongelmia tietokoneen avulla. Ihmisellä tulee olla taito pilkkoa ongelma osiin, taito kaavojen ja toistuvien sääntöjen tunnistamiseen,

kyky luoda algoritmi ongelman ratkaisuksi ja kyky yleistää ja automatisoida toimiva ratkaisu. Liukas ja Mykkänen (2014) korostavat, että ohjelmoinnin opetus peruskoulutasolla on algoritmisen ajattelutaidon kehittämistä.

Wingin (2006, 33) mukaan algoritmisen ajattelu sisältää ongelmanratkaisua, järjestelmän suunnittelua ja ihmisen käyttäytymisen ymmärtämistä tietojenkäsittelyn peruskäsitteitä hyödyntäen. Aho (2011, 2) määrittelee algoritmisen ajattelun ajattelunprosessiksi, jossa ongelman ratkaisut voidaan esittää laskennallisina vaiheina ja algoritmeina. *The International Society for Technology in Education (ISTE)* ja *The Computer Science Teachers Association (CSTA)* mukaan ongelmaratkaisuprosessin ominaisuuksiin kuuluvat myös tietojen järjesteleminen ja looginen analysointi, tiedon kuvaaminen abstraktioilla, ratkaisujen automatisointi, mahdollisten ratkaisujen tunnistaminen, analysoiminen ja toteuttaminen tehokkaasti sekä ongelmanratkaisuprosessin yleistäminen (ISTE & CSTA, 2011, 1). Leppäaho (2007, 30) kuvaa algoritmista ajattelua taitotiedoksi, joka antaa välineet tehtävien ratkaisemiseen ja tulosten kehittämiseen. Tämä esiintyy kognitiivisena taitona esimerkiksi silloin, kun verbaalista lausetta täsmennetään symbolikieleksi tai päinvastoin.

Matemaattisen ajattelun kehittymistä voidaan tutkia kielentämisen avulla, jolloin voidaan tarkastella oppilaiden matemaattisen ajatteluprosessin vahvuuksia ja heikkouksia sekä käsitteiden ymmärtämistä. Kielentäessään oppilas tuo esiin matemaattista ajatteluaan puhekielen, kirjoitetun kielen, matematiikan symbolikielen keinoin sekä graafisin esityksin. Kielentäminen auttaa oppilasta jäsentämään ajatteluaan. (Joutsenlahti & Tossavainen, 2018.) Joutsenlahti ja Tossavainen (2018, 414, 422) tuovat artikkelissaan esille, että kielentämistä tulisi harjoitella systemaattisesti, jotta sitä voidaan onnistuneesti käyttää oppimisessa. Ohjelmoinnin algoritmisen ajattelu ja matemaattinen ajattelu ovat hyvin rinnastettavissa toisiinsa. Sekä ohjelmoinnissa että matematiikassa tarkastellaan asioita täsmällisten määrittelyjen ja tarkkojen ohjeiden avulla (Larsson ym., 2021, 4.) Algoritmisen ajattelun kehittymistä voidaan siten myös tutkia kielentämisen avulla. Kehittymistä ja taitojen näkyväksi tuomista voidaan myös konkretisoida ja havainnollistaa

ohjelmoitavan robotin, kuten BeeBotin, LEGO Mindstorm EV3:n tai Sphero BOLT-robotin avulla (liite 8).

Matematiikan oppikirjoissa algoritmista ajattelua tukevia ja kehittäviä tehtäviä ovat ongelmanratkaisua ja loogista päättelyä vaativat tehtävät, kielentämistehtävät, suunta- ja sijaintikäsitteiden hahmottaminen sekä toimintaohjeen noudattamista vaativat tehtävät. Ongelmanratkaisua ja loogista päättelyä vaativa tehtävä on mekaanista tehtävää avoimempi ja tehtävään voi olla useampia ratkaisuvaihtoehtoja. Ongelmanratkaisutehtävissä oppilaalta vaaditaan aiemmin opittujen tietojen soveltamista. Leppäahon (2007, 32) mukaan monet tutkijat ovat todenneet, että ongelmanratkaisutaito on koko matemaattisen ajattelun ydin. Kielentämistehtävät harjoittavat oppilasta puhumaan ja selittämään tehtävän kulkua ääneen ja esimerkiksi muodostamaan tarinasta lauseke tai lausekkeesta tarina. Kielentäminen auttaa oppilasta jäsentämään ja syventämään omaa ajatteluaan ja tuomaan sen muiden tietoon (Joutsenlahti & Tossavainen, 2018). Suunta- ja sijaintikäsitteiden hahmottamista oppilas voi harjoitella tehtävissä, joissa hänen on hahmotettava esimerkiksi vasemmalle ja oikealle siirtymistä tilanteen mukaan. Toimintaohjeen noudattamista vaativat tehtävät voivat olla joko vaiheittaisten ohjeiden seuraamista, kuten peleissä tai leikeissä, tai sitten toimintaohjeiden suunnittelua ja toteuttamista. Tiedostamme, että myös lukujonotehtävät tukevat algoritmisen ajattelun kehittämistä, mutta ne on kuitenkin rajattu tästä aineistosta pois, koska varsinkin alkuopetuksen matematiikan oppikirjojen tehtävätyypit keskittyvät valtaosin lukujonotehtäviin.

### **3.5 Ohjelmointi ja toiminnallisuus perusopetuksen opetussuunnitelman perusteissa**

Ohjelmat ja ohjelmoitavat laitteet ovat digitalisoituvan yhteiskuntamme toimintaperiaate ja osa jokapäiväistä arkeamme. Opetushallitus hyväksyi perusopetuksen opetussuunnitelman perusteet 22.12.2014 ja näiden perusteiden mukainen paikallinen opetussuunnitelma otettiin käyttöön 1–6 vuosiluokkien

osalta 1.8.2016. Perusopetuksen opetussuunnitelman mukaan ohjelmoinnin tavoitteena on saada kokemuksia siitä, miten ihmisen tekemät ratkaisut vaikuttavat teknologian toimintaan (Opetushallitus, 2014, 157).

Perusopetuksen opetussuunnitelman perusteissa ohjelmointi ja algoritmisen ajattelu on sisällytetty osaksi matematiikan oppiainetta. Matematiikan oppiaineen tehtävässä myös ohjelmoinnin ja algoritmisen ajattelun taidot korostuvat monessa kohtaan. Oppiaineen tehtävässä vuosiluokille 1–2 ja 3–6 (Opetushallitus, 2014, 128, 234) sanotaan seuraavaa:

Matematiikan opetuksen tehtävänä on kehittää oppilaiden loogista, täsmällistä ja luovaa matemaattista ajattelua. Opetus luo pohjan matemaattisten käsitteiden ja rakenteiden ymmärtämiselle sekä kehittää oppilaiden kykyä käsitellä tietoa ja ratkaista ongelmia. Matematiikan kumulatiivisesta luonteesta johtuen opetus etenee systemaattisesti. Konkretia ja toiminnallisuus ovat keskeinen osa matematiikan opetusta ja opiskelua. Oppimista tuetaan hyödyntämällä tieto- ja viestintäteknologiaa.

Ohjelmointi ja algoritmisen ajattelu sisältävät juuri loogista, täsmällistä ja luovaa matemaattista ajattelua sekä tiedon käsittelyä ja ongelmien ratkaisemista. Ohjelmoinnin avulla oppii myös yleisesti hyödyllisiä kognitiivisia taitoja, kuten loogista ja luovaa ajattelua, tarkkaa työskentelyä ja ongelman hahmottamis- ja ratkaisukykyä (Liukas ja Mykkänen, 2014, 57). Matematiikan oppiaineen tehtävässä tuodaan esiin myös selkeästi konkretia ja toiminnallisuus sekä tieto- ja viestintäteknologian hyödyntäminen. Ohjelmoinnissa apuna voidaan käyttää erilaisia ohjelmitavia robotteja konkretisoimaan ja havainnollistamaan algoritmista ajattelua. Muñoz-Repiso ja Caballero-Gonzálezin (2019, 65, 69) tutkimuksen mukaan ohjelmitavat robotit edistivät algoritmisen ajattelun kehittymistä positiivisesti. Lapset, jotka olivat opiskelleet ohjelmointia robotiikan avulla, olivat kehittyneet havaitsemaan virheitä ohjelmoinnissa ja tarkistamaan ohjelman toimivuuden. Robotiikalla on suuri vaikutus opiskelijoiden taitoihin ja kykyihin, kuten ongelmanratkaisutaitoihin, kriittiseen ajatteluun, päätöksentekokykyyn ja tiimityöskentelyyn (Khanlari, 2013).

Matematiikan oppiaineen oppimisympäristöissä ja työtavoissa korostuvat

toiminnallisuus ja välineillä työskentely sekä pedagogisesti ohjatut leikit ja pelit (Opetushallitus, 2014, 130, 236). Liukkaan ja Mykkäsen (2014, 48, 86) mukaan ohjelmoinnillisia taitoja harjoitellaan kahdella ensimmäisellä luokalla leikkien kautta, joissa opetellaan antamaan yksikäsitteisiä komentoja. Leikeissä pyritään suorittamaan tietyt toimet tietyssä järjestyksessä tehtävän ratkaisemiseksi.

Matematiikan oppiaineen sisällöissä ja tavoitteissa (Opetushallitus, 2014, 128–129, 235–236) sekä laaja-alaisen osaamisen tavoitteissa (Opetushallitus, 2014, 99–101, 155–157) esiintyy useaan kertaan viittauksia asioista, joita löytyy myös ohjelmoinnin ja algoritmisen ajattelun taustalta. Matematiikan oppiaineen ohjelmointiin liittyvät tavoitteet ja sisällöt vuosiluokille 1–6 on esitetty kokonaisuudessaan yhteenvetotaulukossa (liite 1). Vuosiluokkien 1–2 matematiikan oppiaineen ohjelmointiin liittyvät sisällöt ja tavoitteet sekä laaja-alaisen osaamisen tavoitteet (taulukko 1) eroavat jokseenkin vuosiluokkien 3–6 vastaavista (taulukko 2). Molempien vuosiluokkakokonaisuuksien tavoitteissa korostuvat kuitenkin ongelmanratkaisutaidot, vaiheittaisten toimintaohjeiden laatiminen ja ymmärtäminen eli algoritmisen ajattelun taidot sekä päättelyn ja ratkaisujen esittäminen erilaisten konkreettisten välineiden avulla. Matematiikan Ajattelun taidot (S1) -sisällöissä nousevat esiin säännönmukaisuuksien löytämisen taidot, vertailu- ja luokittelutaidot sekä syy- ja seuraussuhteiden havaitseminen. 1–2 vuosiluokkien Geometria ja mittaaminen (S3) -sisältöalueissa mainitaan ohjelmoinnissakin hyödyllisten suunta- ja sijaintikäsitteiden harjoittelu.

## Taulukko 1

Matematiikan oppiaineen tavoitteet ja sisällöt vuosiluokille 1-2

### MATEMATIIKAN OPPIAINE 1.-2. lk.

TAVOITTEET	SISÄLLÖT	LAAJA-ALAISET TAVOITTEET
<p>T3 kannustaa oppilasta esittämään ratkaisujaan ja päättämään konkreettisin välinein, piirroksin, suullisesti ja kirjallisesti myös tieto- ja viestintäteknologiaa hyödyntäen</p> <p>T4 ohjata oppilasta kehittämään päättely- ja ongelmanratkaisutaitojaan</p> <p>T12 harjaannuttaa oppilasta laatimaan vaihteellaisia toimintaohjeita ja toimimaan ohjeen mukaan</p>	<p>S1 Ajattelun taidot</p> <p>Oppilaille tarjotaan mahdollisuuksia löytää yhtäläisyyksiä, eroja ja säännönmukaisuuksia. Vertaillaan, luokitellaan ja asetetaan järjestykseen sekä havaitaan syy- ja seuraussuhteita. Harjoitellaan tarkastelemaan matemaattisia tilanteita eri näkökulmista. Tutustuminen ohjelmoinnin alkeisiin alkaa laatimalla vaihteellaisia toimintaohjeita, joita myös testataan.</p> <p>S3 Geometria ja mittaaminen</p> <p>Kehitetään oppilaiden taitoa hahmottaa kolmiulotteista ympäristöä ja havaita siinä tason geometriaa. Harjoitellaan suunta- ja sijaintikäsitteiden käyttöä</p>	<p>L1 Ajattelu ja oppimaan oppiminen</p> <p>Ihmettelylle, oivaltamiselle, uuden löytämiselle ja keksimiselle, mielikuvitukselle sekä oppimisen ilolle on tilaa.</p> <p>Ikäkaudelle sopivien ongelmanratkaisu- ja tutkimustehtävien avulla viritetään uteliaisuutta ja kiinnostusta ympäröivään maailman ilmiöitä kohtaan sekä vahvistetaan taitoa jäsentää, nimetä ja kuvailla ympäristöä. Oppilaita rohkaistaan myös kyseenalaistamaan havaitsemiaan asioita ja huomaamaan, että tieto voi olla joskus ristiriitaista ja epäselvää</p> <p>Monipuolinen liikkuminen ja motoriset harjoitukset pelien ja leikkien avulla.</p> <p>L5 Tieto- ja viestintäteknologinen osaaminen</p> <p>Leikkiin perustuva työskentely on edelleen keskeistä. Tieto- ja viestintäteknologian perustaitoja harjoitellaan ja opitaan käyttämään niitä opiskelun välineinä. Samalla opitaan keskeistä käsitteistöä. Oppilaat pohtivat myös, mihin tarkoituksiin tieto- ja viestintäteknologiaa käytetään lähiympäristössä ja mikä sen merkitys on arjessa.</p> <p>Koulutyössä harjoitellaan laitteiden, ohjelmistojen ja palveluiden käyttöä ja opetellaan niiden keskeisiä käyttö- ja toimintaperiaatteita. Samoin harjoitellaan näppäintaitoja sekä muita tekstin tuottamisen ja käsittelyn perustaitoja. Oppilaat saavat ja jakavat keskenään kokemuksia digitaalisen median parissa työskentelystä sekä ikäkaudelle sopivasta ohjelmoinnista. Pelillisyyttä hyödynnetään oppimisen edistäjänä.</p>

3.–6. vuosiluokkien tavoitteissa (taulukko 2) innostetaan oppilaita laatimaan toimintaohjeita tietokoneohjelmina graafisessa ohjelmointiympäristössä. Graafinen, lohkopohjainen ohjelmointi on hyvin yleinen ja vakiintunut tapa tutustuttaa oppilaat ohjelmointiin (Weintrop, 2019). Laaja-alaisen osaamisen taidot korostavat ajattelua ja oppimaan oppimista (L1). Niissä painotetaan ihmettelyn, oivaltamisen, luovuuden ja mielikuvituksen merkitystä. Korhosen (2021) mukaan ohjelmointi ei ole vain tekniikkaa ja algoritmista ajattelua, vaan myös luovuuden kehittämistä. Myös toiminnallisuus näkyy laaja-alaisissa tavoitteissa motorististen harjoitusten kautta pelien ja leikkien avulla. Toiminnalliset harjoitukset, kuten reitti ohjeiden antaminen ja niiden mukaan toimiminen, ovat hyviä ohjelmoinnilliseen ajatteluun johdattelevia harjoituksia. Laaja-alaisen osaamisen tavoitteissa, Tieto- ja viestintäteknologinen osaaminen (L5) mainitaan, että ohjelmointia kokeillessaan oppilaat saavat kokemuksia siitä, miten teknologian toiminta riippuu ihmisen tekemistä ratkaisuista.



## Taulukko 2

Matematiikan oppiaineen tavoitteet ja sisällöt vuosiluokille 3-6

### MATEMATIIKAN OPPIAINE 3.–6. lk.

TAVOITTEET	SISÄLLÖT	LAAJA-ALAISET TAVOITTEET
<p>T3 ohjata oppilasta kehittämään taitoaan esittää kysymyksiä ja tehdä perusteltuja päätelmiä havaintojensa pohjalta</p> <p>T4 kannustaa oppilasta esittämään päättelyään ja ratkaisujaan muille konkreettisin välinein, piirroksin, suullisesti ja kirjallisesti myös tieto- ja viestintäteknologiaa hyödyntäen</p> <p>T5 ohjata ja tukea oppilasta ongelmanratkaisutaitojen kehittämisessä</p> <p>T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoina graafisessa ohjelmointiympäristössä</p>	<p>S1 Ajattelun taidot</p> <p>Kehitetään oppilaiden taitoja löytää yhtäläisyyksiä, eroja ja säännönmukaisuuksia. Syvennetään taitoa vertailla, luokitella ja asettaa järjestykseen, etsiä vaihtoehtoja systemaattisesti, havaita syy- ja seuraussuhteita sekä yhteyksiä matematiikassa. Suunnitellaan ja toteutetaan ohjelmia graafisessa ohjelmointiympäristössä.</p> <p>S3 Algebra</p> <p>Tutkitaan lukujonon säännönmukaisuutta sekä jatketaan lukujonoa säännön mukaan. Tutustutaan tuntemattoman käsitteeseen. Tutkitaan yhtälöä ja etsitään yhtälön ratkaisuja päättelämällä ja kokeilemalla.</p>	<p>L1 Ajattelu ja oppimaan oppiminen</p> <p>Oppilaita ohjataan huomaamaan asioiden välisiä vuorovaikutussuhteita ja keskinäisiä yhteyksiä.</p> <p>Ajattelun taitoja harjoitellaan ongelmanratkaisu- ja päättelytehtävin sekä uteliaisuutta, mielikuvitusta, kekseliäisyyttä ja toiminnallisuutta hyödyntävin ja edistävoin työskentelytavoin. Oppilaita rohkaistaan käyttämään kuvittelukykyään luoviin ratkaisuihin ja olemassa olevien rajojen ylittämiseen.</p> <p>L5 Tieto- ja viestintäteknologinen osaaminen</p> <p>Koulussa tutkitaan tieto- ja viestintäteknologian vaikutusta arkeen</p> <p>Oppilaat oppivat käyttämään erilaisia laitteita, ohjelmistoja ja palveluita sekä ymmärtämään niiden käyttö- ja toimintalogiikkaa.</p> <p>Oppilaita kannustetaan toteuttamaan tot:n avulla ideoitaan yksin ja yhdessä toisten kanssa. Ohjelmointia kokeillessaan oppilaat saavat kokemuksia siitä, miten teknologian toiminta riippuu ihmisen tekemistä ratkaisuista.</p>

Matematiikan oppiaineen lisäksi perusopetuksen opetussuunnitelman perusteista löytyy ohjelmointiin liittyvä sisältöalue 3.–6. -luokkien käsitöiden kohdalta: S3 Kokeilu "Harjoitellaan ohjelmoimalla aikaan saatuja toimintoja, joista esimerkkinä robotiikka ja automaatio." (Opetushallitus, 2014, 271). Tämä on ainut kohta perusopetuksen opetussuunnitelman perusteissa, jossa mainitaan termit robotiikka ja automaatio. Kuten aikaisemmin mainitsimme, ohjelmoitavien robottien avulla ohjelmoinnista saadaan näkyvää ja ohjelmoinnillisten taitojen kehittymistä voidaan näin seurata.

Arvioinnin yhteydessä ohjelmoinnista mainitaan erikseen ainoastaan 6. vuosiluokan päätteeksi matematiikan hyvää osaamista kuvaavissa sanallista arviota tai arvosanaa kahdeksan varten annetuissa arviointikriteereissä, jonne on T14 -tavoitteen osalta kirjattu seuraavasti: "Oppilas osaa ohjelmoida toimivan ohjelman graafisessa ohjelmointiympäristössä." (Opetushallitus, 2014, 239.)

Kuten edellä toimme esille, ohjelmointi ei ole oma oppiaineensa, vaan se on sisällytetty opetussuunnitelman laaja-alaisen osaamisen tavoitteisiin ja integroitu matematiikan ja käsitöiden oppiaineisiin. Ohjelmointi vuosiluokilla 1–2 tarkoittaa käytännössä sitä, että oppilaita ohjataan ratkaisemaan ongelmia, jakamaan ongelmia pienempiin osiin sekä antamaan ohjeita riittävän selkeästi ja yksiselitteisesti. Näiden harjoittelu ei välttämättä vaadi tietokonetta tai muuta tekniikkaa, vaan oppiminen tapahtuu leikkien ja pelien kautta. Kouluarjessa tapahtuva jokapäiväinen ohjeistaminen on esimerkki ikäkaudelle annettavasta ohjelmoinnista. Oppilaiden ohjeistaminen esimerkiksi liikunnan tunnin liikesarjoissa, monivaiheisten käsityötehtävien teossa ja erilaiset toiminnan ohjaukselliset ohjeistukset, kuten "kun saat tehtävän tehtyä, jatka edellistä keskeneräistä tehtävää" tai "jos saat tehtäväsi valmiiksi, siirry tekemään seuraavaa tehtävää." Näin ohjelmointiin liittyvää termistö tulee oppilaille tutuiksi jo alkuopetuksen arjessa. Opettajan antama malli ohjeistuksesta, erityisesti sen tarkkuudesta, yksiselitteisyydestä ja johdonmukaisuudesta harjaannuttavat oppilaita algoritmisen ajattelun kehittymisessä. Arjen koulutyöskentelyssä opettaja tarjoaa oppilaille mahdollisuuksia kehittää ongelmaratkaisutaitoja, ongelmien pilkkomista pienempiin osiin ja ajattelua

siitä, että useampien pienempien osavaiheiden kautta päästään haluttuun ratkaisuun.

Vuosiluokilla 3–6 ohjelmoinnissa pääpaino on edelleen algoritmisen ajattelun kehittämisessä, jota ovat tukemassa graafiset ohjelmointiympäristöt, kuten Scratch-ympäristö ja yleishyödyllisen Code.org -yhteisön ohjelmointiympäristö, joka soveltuu erityisesti alakouluikäisille lapsille ensimmäiseksi ohjelmoinnin harjoitteluympäristöksi (liite 8).

Opetussuunnitelman perusteissa korostetaan oppimisen toiminnallisuutta ja sen edistämisen tärkeyttä. Toiminnalliset työtavat, moniaistillisuus, kokemuksellisuus ja konkretia on nostettu vahvasti esiin opetussuunnitelman perusteissa. (Opetushallitus, 2014, 27, 30.) Toiminnallisuudella tarkoitetaan oppilaan toiminnan ja ajatuksen aktiivisuutta, osallistumista, kokemuksellisuutta ja oppilaiden välistä vuorovaikutusta (Sahlberg & Leppilampi, 1994). Myös Dewey ja Lewin pitivät toiminnallisessa oppimisessä vuorovaikutusta ja palautteen antoa tärkeänä (Öystilä, 2003, 69). Sahlberg ja Leppilampi (1994) kuvailevat toiminnallisiksi opetusmenetelmiksi esimerkiksi ryhmätyöt, ongelmanratkaisutehtävät, väittelyt, leikit, projektityöt, draaman, tutkimustehtävät ja yhteistoiminnallisen oppimisen. Nämä edellä mainitut opetusmenetelmät on todettu tehokkaiksi, koska niissä oppilas pääsee osallistumaan ja toimimaan oppimisprosessissa aktiivisesti.

Perusopetuksen opetussuunnitelmassa todetaan, että fyysinen aktiivisuus, leikit, kokeellisuus ja monipuoliset toiminnalliset työtavat lisäävät oppimisen iloa ja mahdollistavat luovan ajattelun kehittymisen sekä oppimisen elämyksellisyyttä. Toiminnallisten työtapojen ja liikkumisen todetaan lisäävän motivaatiota. Esimerkiksi monialaisten oppimiskokonaisuuksien yhteydessä on tarkoitus käsitellä toiminnallisoin keinoin oppilaan kokemusmaailmaan liittyviä kokonaisuuksia. Vuosiluokkien 1–2 työtapojen tavoitteissa korostetaan toiminnallisuuden, leikkien, pelien, tarinallisuuden ja mielikuvituksen käytön tärkeyttä osana opetusta (Opetushallitus, 2014, 17–32, 98).

Nykyisessä opetussuunnitelmassa (Opetushallitus, 2014, 17) korostetaan oppilaan aktiivisuutta ja osallisuutta oppimisprosessissa. Siinä oppilas on

aktiivinen toimija ja toimii vuorovaikutuksessa yhdessä toisten kanssa. Oppiminen nähdään kumuloituvana, oppilasta ohjataan rakentamaan tietoa aiemmin opittujen käsitteiden päälle ja syventämään ymmärrystään opituista asioista. Myös Deweyn (1938, 35) mukaan aiemmilla kokemuksilla on vaikutusta uuden oppimiseen ja kokemusten muodostumiseen. Opetussuunnitelman perusteissa painotetaan myös itseohjautuvuuden tärkeyttä, jota myös Deweyn ajattelu korostaa (Opetushallitus, 2014, 30; Dewey, 1938, 27–28). Dewey (1938, 38) kuvailee toiminnallista oppimista ajattelun ja toiminnan reflektoinnin kautta.

### 3.6 Toiminnallisuus ohjelmoinnissa

Luvussa 2.2 kuvasimme, mitä toiminnallisuus opetuksessa on. Samat periaatteet voidaan nähdä myös ohjelmoinnin opettamisessa. Toiminnallisuus ohjelmoinnin opettamisessa on oppilaan aktiivista toimimista ja ajattelemista oppimisprosessin aikana yhdessä muiden kanssa.

Kun teknologia jätetään pois ja oppimista edistetään ilman tietokonetta, voidaan keskittyä algoritmisen ajattelun kehittämiseen välttämällä tietotekniset haasteet. Tästä käytetään englannin kielessä termiä *unplugged*. Kun asiat on opittu ensin ilman teknologiaa, voidaan opittuja asioita soveltaa tietotekniikan avulla. Ilman tietokonetta toteutettavassa opetuksessa vältetään ruudun ääressä istumista ja itsenäistä työskentelyä, jolloin voidaan hyödyntää monia muita tapoja oppia yhdessä muiden kanssa, kuten toiminnallisuutta. (Caldwell & Smith, 2017, 4)

Ilman tietokonetta toteutettavat ohjelmointi ja algoritmista ajattelua kehittävät ohjelmointitehtävät ovat usein toiminnallisia. Harjoitteluun tarvitaan vain oma keho, kynä ja paperi tai erilaiset robotit. Ihmisrobotiharjoitteet ovat opettavaisia ja yhteistoiminnallisia, ja niissä oppilaat näkevät konkreettisesti sen, kuinka tärkeää on komentojen tarkkuus ja täsmällisyys. Samalla oppilaat kehittävät loogista ja luovaa ongelmanratkaisukykyään (Caldwell & Smith, 2017, 8–9). Ihmisrobotiharjoitteiden yhteydessä myös oppilaiden suunta- ja

sijaintikäsitteiden ymmärrys ja avaruudellinen hahmottamiskyky kehittyvät. Caldwell ja Smithin (2017, 9) mukaan lapset usein kehittävät itseohjautuvasti monimutkaisempia toimintaohjeita, joissa he käyttävät ehto- ja toistokomentoja, kun he aluksi ovat oivaltaneet, kuinka heidän antamansa peruskomennot toimivat ihmisrobotin ohjauksessa.

Tässä tutkimuksessa toiminnallisiksi ohjelmointiharjoitteiksi tai algoritmista ajattelua kehittäviksi ja tukeviksi tehtäviksi laskemme esimerkiksi kaikki vuorovaikutteisesti suoritettavat ohjeenantotehtävät, joissa esimerkiksi oppilas ohjaa toista oppilasta tai opettajaa (ihmisrobotti) toimimaan annetun toimintaohjeen mukaisesti. Toiminnallisia harjoitteita ovat lisäksi kaikenlaiset pelit ja leikit. Yhtenä toiminnallisuuden muotona näemme myös robotiikan hyödyntämisen ohjelmoinnin opetuksessa. Opetustarkoitukseen löytyy monenlaisia ohjelmoitavia robotteja. Lattiarobottien, esimerkiksi Bee-Bot, Blue-Bot, Pro-Bot ja Sphero BOLT (liite 8), avulla voi harjoitella suunta- ja sijaintikäsitteitä. Drone-lennokkien avulla käsitteitä voi harjoitella ilmasta käsin. Usein ohjelmoitavat robotit ohjelmoidaan bluetooth-yhteyden välityksellä graafisessa ohjelmointiympäristössä joko tietokonetta tai tablet-tietokonetta käyttäen. Tässä tutkimuksessa toiminnallisista harjoitteista rajauksemme ulkopuolelle jätettiin yksin tehtävät kynä-paperi-tehtävät.

## 4 OPPIMATERIAALEISTA JA NIIDEN TUTKIMUKSISTA

Perinteisessä mielessä oppilaan näkökulmasta oppimateriaalina toimii oppikirja ja opettajan näkökulmasta opettajan opas. Oppikirja on teos, jonka tulisi pohjautua perusopetuksen opetussuunnitelman perusteisiin ja se on laadittu opetustarkoituksiin. Sen ensisijainen tehtävä on välittää oppilaalle tietoa ja ymmärrystä opittavasta asiasta. (Heinonen, 2005, 29.) Opettajan opas toimii opettajan tukena opetuksessa ja sen suunnittelussa ajoitusehdotuksineen, toteutusvinkkeineen ja lisätehtävineen (Perkkilä, 2002, 87, 156). Heinosen (2005) mukaan nykyinen oppimateriaalikäsitys sisältää kaiken materiaalin, joka tukee opettamista ja oppimista.

Perkkilä (2002) toteaa väitöskirjassaan, että suomalaiset oppikirjat saavat paljon vaikutteita behavioristisesta opettamisen mallista. Perkkilän, Joutsenlahden ja Sareniuksen (2018, 344–345, 349) mukaan oppimateriaaleja pidetään lähes opetussuunnitelman roolissa. Tällöin kustantajilla on suuri vastuu materiaalien kehittämistyössä nykyisten oppimiskäsitysten suuntaisiksi, opetussisältöjen etenemisen loogisuudessa sekä matematiikan kiinnostavuuden lisäämisessä. Oppimisprosessin suunnittelussa ja toteutuksessa pitäisi kiinnittää huomiota oppilaiden yksilöllisiin tarpeisiin ja mahdollistaa oppilaalle matemaattisen tietorakenteen rakentaminen yhdessä toisten oppilaiden kanssa. Oppikirjoja tulisi pitää vain yhtenä osana matemaattisen tiedon rakentelun prosessia. Myös Rauste-von Wright, von Wright ja Soini (2003, 142, 200) löytävät suomalaisista oppikirjoista vaikutteita behavioristisesta oppimiskäsityksestä, jossa oppijaa pidetään passiivisena tiedon vastaanottajana. Perkkilän, Joutsenlahden ja Sareniuksen (2018, 347) mukaan oppimisympäristöt tulisi olla vaihtelevia ja oppimista tulisi tapahtua myös muulla tavoin kuin kirjajohtoisesti. Järvillehto (2014, 18, 185) kannustaakin ottamaan huomioon oppilaiden osallistamisen oppimisympäristöjä ja -sisältöjä suunnitellessa. Hän mainitsee, miten kiinnostusta herättävä oppisisältö innostaa ja sitouttaa oppilaita

oppimisprosessiin ja mahdollistaa tehokkaan oppimisen. Oppikirjan roolia tulisi siis pitää ohjaavana työkaluna.

Oppikirjan rooli peruskoulun matematiikan opetuksessa on tutkimusten mukaan suuri, ja oppitunnin rakenteeseen ja sisältöihin vaikuttavat paljon oppikirjojen anti. Esimerkiksi Joutsenlahti ja Vainionpää (2010, 138) toteavat, että erityisesti painettu oppimateriaali on matematiikan opetuksessa keskeisessä asemassa. Myös Perkkilän, Joutsenlahden ja Sareniuksen (2018, 346) artikkelissa tulee esille, että matematiikan opetuksessa korostuu oppikirjan rooli ja että matematiikan oppitunnin rakennetta ohjaa hyvin pitkälti juuri oppikirjan struktuuri. Matematiikan oppikirjat mielletään usein asiantuntijoiden laatimiksi ja usein opettaja luottaa oppikirjan struktuuriin ja noudattaa sitä orjallisesti. Sisältöjen karsimista ja oppikirjojen struktuurin rikkomista pidetään jopa riskinä (Heinonen, 2005, 35.)

Matematiikan oppituntien rakenne on usein oppikirjan struktuurin ”aukeama tunnissa” mukainen. Oppitunnin alussa opettaja opettaa uuden asian, ja lopputunnin oppilaat harjoittelevat sitä itsenäisesti käyden välillä tarkistamassa ratkaisujen oikeellisuutta tarkistuspisteellä olevista tuloskirjoista. Lopputulosta pidetään tärkeämpänä kuin erilaisia ratkaisutapoja. (Perkkilä, 2002, 11, 21–22; Perkkilä ym., 2018.) Tikkanen (2008, 49) lisää kaavamaisuuteen vielä oppituntien alussa tarkistettavat kotitehtävät ja oppitunnin päätteeksi annettavat uudet kotitehtävät.

Matematiikan oppikirjasarjojen sisällölliset ja rakenteelliset erot ovat suhteellisen pieniä (Joutsenlahti & Vainionpää, 2010, 143). Matematiikan oppikirjan rakenne on hyvin samankaltainen kustantajasta riippumatta. Uuden opetettavan asian käsittely aloitetaan sivun ylälaudassa opetusruudussa, jonka jälkeen asiaa opiskellaan tehtävien avulla mekaanisista tehtävistä kohti soveltavimpia tehtäviä. Seuraavalla aukeamalla jatkuu soveltavat ja vaativammat tehtävät ja lopuksi tarjotaan muutama kotitehtäväksi soveltuva tehtävä. Joutsenlahden ja Vainionpään (2010) mukaan oppilaiden osaamistasoa ei voi siten perustella pelkästään oppikirjasarjojen sisältö- ja rakenneratkaisuilla.

Vuoteen 1990 saakka oppikirjojen tuli saada Kouluhallituksen hyväksyntä tarkastusmenettelyssä, jossa Kouluhallitus pyrki valvomaan, että oppikirjojen sisältö vastasi meneillään olevaa opetussuunnitelmaa (Heinonen 2005, 30). Heinonen (2005, 35) toteaaakin, että oppikirja on ottanut opetussuunnitelman aseman virallisen opetussuunnitelman sijaan. Kirjakustantajilla on siis vahva epävirallinen asema ohjaavana tekijänä opettajien suunnittelutyössä.

Joutsenlahti ja Vainionpää (2007) tuovat tutkimuksessaan esille, että oppikirjoissa toistuu usein kaavamaisuus, jolla käsiteltävä asia käydään läpi. Tämä näkyy muun muassa sisältöjen jaksottamisessa, oppitunnin struktuurissa ja ehdotetuissa työtavoissa. Kaavamaisuus toistuu vuosiluokalta toiselle hyvin samanlaisena eri kustantajilla ja kustantajien välilläkin. Tämä saattaa helpottaa opettajan työtä, mutta voi oppilaan näkökulmasta näyttäytyä enemmän suorittamisena ja oppikirjan täyttämisenä kuin uteliaalta ymmärrystä kasvattavana oppimisena.

Digitaaliset oppimateriaalit ovat tulleet olennaiseksi osaksi oppimateriaaleja 2000-luvulla tekniikan kehittyessä. Digitaaliset materiaalit perustuvat uudenlaiseen oppimisen ymmärtämiseen, esimerkiksi opiskelun ja vuorovaikutuksen tapoihin. Niissä korostuvat nykyään erilaiset oppimispelit ja oppimistapahtumissa hyödynnettävät osallistumisvälineet. (Hiidenmaa, 2015, 35.) Verkkopohjainen digitaalinen oppimateriaali antaa monipuolisia mahdollisuuksia opetukseen ja se vastaa myös paremmin oppimateriaalina konstruktivismin tuomiin haasteisiin. (Heinonen, 2005, 42.) Digitaaliset oppimateriaalit sisältävät usein kaiken oppimateriaalin mukaan lukien oppilaan oppikirjan, opettajan oppaat, digitaaliset oppimispelit ja -tehtävät. Oppilaan oppikirjan digitaalinen versio mahdollistaa esimerkiksi tekstin suurentamisen, oman äänen tallentamisen, vuorovaikutteiset oppimistavat ja erilaisia aiheeseen liittyviä animaatioita (Lerkanen, 2015, 100). Digitaalisten oppimateriaalien tarkoitus onkin tuoda opetukseen painettuja materiaaleja enemmän syvyyttä ja ulottuvuuksia. Digitaalisten materiaalien alkuaikoina materiaalien suunnittelu perustui liiaksi teknologian mahdollisuuksiin eikä niinkään pedagogisiin periaatteisiin. Tällöin suunnittelun tulos on huonoimmillaan vain ”kirja



verkossa” (Ilomäki, 2012, 10).

Tässä tutkimuksessa kutsumme oppimateriaaliksi oppilaan oppikirjaa ja opettajan opasta. Saimme kahdelta kustantajalta, Sanoma Pro Oy:ltä ja Otava Oy:ltä, käyttöömme digitaaliset 1.-4. luokkien matematiikan oppimateriaalit, jotka sisälsivät oppilaiden oppikirjat, digiopetusmateriaalin ja opettajan oppaat. Tutkimuksen kvantitatiiviseen osuuden arvot on laskettu oppilaan oppikirjoista ja opettajan oppaista. Digiopetusmateriaalien sisältö on hyvin samankaltainen kuin opettajan oppaissa. Ne ovat vain erilaisessa esitysmuodossa, joten toiston välttämiseksi digiopetusmateriaalit on rajattu pois laskennasta.

## 5 TUTKIMUSTEHTÄVÄ JA -KYSYMYKSET

Nyky-yhteiskunnassa algoritmisen ajattelutaidon kehittäminen on välttämätöntä, sillä iso osa ympäristöstämme toimii tietotekniikan avulla. Monipuolinen ohjelmoinnin opetus tukee ja kehittää algoritmisen ajattelutaidon kehittymistä. Perusopetuksen opetussuunnitelmassa ohjelmointiosaaminen on ollut mukana matematiikan tavoitteissa vuodesta 2014 alkaen, mutta se voi edelleen olla hieman vieras aihe osalle opettajista. Toiminnallista oppimista on hyödynnetty opetuksessa jo pitkään, ja opetusmenetelmä mainitaankin useasti myös perusopetuksen opetussuunnitelmassa. Ohjelmointi aiheena on ajankohtainen ja mielenkiintoinen ja tästä syystä halusimme lähteä tutkimaan ohjelmoinnin opetusta alakoulussa ja sitä, miten toiminnallisuus näyttäytyy opetuksessa.

Tutkimuksemme tarkoituksena oli selvittää, miten ohjelmointia opetetaan ja kuinka paljon sitä löytyy matematiikan oppimateriaaleista vuosiluokilla 1–4. Lisäksi selvitimme, miten toiminnallisuus on huomioitu ohjelmoinnin opetuksessa. Tutkimuksessa kiinnostuksemme kohdentuu varsinaisten ohjelmointitehtävien lisäksi myös matematiikan oppimateriaaleissa esiintyvien algoritmista ajattelua kehittävien ja tukevien tehtävien määrään. Numeeristen arvojen tueksi haastattelimme kahta luokanopettajaa, joilla oli runsaasti kokemusta ohjelmoinnin opettamisesta alakoulussa.

Tutkimuskysymyksemme ovat:

1. Miten ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4?
  - Kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4?
2. Miten toiminnallisuus on otettu huomioon ohjelmoinnin opetuksessa matematiikan oppimateriaaleissa vuosiluokilla 1–4?
3. Minkälaisia kokemuksia luokanopettajilla on ohjelmoinnin opettamisesta?

## 6 TUTKIMUKSEN TOTEUTTAMINEN

### 6.1 Tutkimuksen metodologiset valinnat

#### 6.1.1 Mixed methods

Tämä tutkimus toteutettiin mixed methods -tutkimuksena. Hesse-Biberin (2010, 3) mukaan mixed methods -menetelmä on tutkimus, jossa pyritään vastaamaan tutkimuskysymyksiin laadullisen ja määrällisen aineiston avulla. Näitä aineistoja voidaan yhdistellä monella eri tavalla, esimerkiksi tutkimustuloksissa tai apuna kehittämässä toista aineistonkeruumenetelmää ja täydentämään toisiaan. (Creswell & Plano Clark, 2007, 62–79; Hirsjärvi ym., 2009, 136–137.) Mixed methods -tutkimuksessa aineisto kerätään ja analysoidaan laadullisesti ja määrällisesti. Sen käyttöä perustellaan erilaisilla hyödyillä ja tarkoituksilla, sillä usein tämä tutkimusmalli tuo laajuutta ja vahvistusta tutkimukselle. Näin tutkimusaiheesta voidaan mahdollisesti saada runsaampi ymmärrys kuin yksittäisellä menetelmällä. (Johnson ym., 2007, 122; Creswell & Plano Clark, 2007, 5.)

#### 6.1.2 Kvalitatiivinen osuus

Kvalitatiivisessa eli laadullisessa tutkimuksessa pyritään kuvaamaan todellista elämää tutkimalla kohdetta mahdollisimman kokonaisvaltaisesti. Kvalitatiivisen tutkimuksen tavoitteena on löytää tai paljastaa tosiasioita eikä tyytyä pelkästään todentamaan jo olemassa olevia väittämiä. (Hirsjärvi ym., 2009, 161.) Kvalitatiivinen tutkimusote soveltuu tutkimukseen erityisesti silloin, kun ollaan kiinnostuneita tietyssä tapahtumassa mukana olleiden yksittäisten toimijoiden merkitysrakenteista (Metsämuuronen, 2009, 203). Tutkimuksemme tarkastelun kohteena olivat luokanopettajien käsitykset ja kokemukset ohjelmoinnin opettamisesta alakoulussa ja heidän ajatuksensa, miten matematiikan oppimateriaalit tukevat sitä. Lisäksi halusimme selvittää, miten luokanopettajat ovat huomioineet toiminnallisuuden ohjelmoinnin opettamisessa. Tutkittavien ilmauksista etsittiin erilaisia käsityksiä tutkittavasta aiheesta. Luokanopettajilta

saatu haastatteluaineisto analysoitiin teorialähtöisellä sisällönanalyysillä. Tutkimuksemme teorialähtöisellä sisällönanalyysillä tarkoitetaan aineiston analysointia teorian pohjalta laaditun käsitejärjestelmän mukaisesti (Tuomi & Sarajärvi, 2018, 127).

Tutkimuksemme laadullinen osuus toteutettiin haastattelemalla kahta luokanopettajaa heidän kokemuksistaan ohjelmoinnin opettamisesta alakoulussa. Haastattelussa käyttämämme haastattelurunko löytyy liitteestä 4. Haastattelu on hyvä ja paljon käytetty aineistonkeruumenetelmä, kun halutaan tietää, mitä ihminen ajattelee jostakin asiasta, vaikka se onkin aineistonkeruumenetelmänä melko aikaa vievä (Tuomi & Sarajärvi, 2018 83–86). Haastattelun idea on esittää kysymyksiä ja saada vastauksia niille. Haastattelijan tehtävä on ohjata keskustelua, mutta välttää esittämästä omaa tietämystään aiheesta kommentoimalla haastateltavan vastauksia haastattelutilanteessa. (Hirsjärvi & Hurme, 2008, 102–103; Tuomi & Sarajärvi, 2018, 84–86.) Hirsjärven ja Hurmeen (2008) mukaan haastattelulla on tarkoitus saada selville tutkittavan käsityksiä, ajatuksia, tunteita ja kokemuksia tutkittavasta aiheesta. Tässä tutkimuksessa pyrimme saamaan käsityksen luokanopettajien ajatuksista ja kokemuksista toiminnallisen ohjelmoinnin opettamisesta alakoulussa sekä siitä, miten he kokevat matematiikan oppimateriaalit osana ohjelmoinnin opettamista. Haastattelukysymykset ja teemat annettiin tutkittaville jo hyvissä ajoin ennen haastattelua, jotta varmistimme runsaan tutkimusmateriaalin saannin. Haastattelukysymysten ja teemojen pohjana toimi valitsemamme tutkimuskysymykset.

Haastattelun etuina ovat erityisesti sen joustavuus ja monipuolisuus. Tutkija voi tarvittaessa toistaa kysymyksen haastattelun edetessä tai selventää ilmausten sanamuotoa tutkittavalleen. Myös kysymysten esittäminen voidaan toteuttaa tutkijan haluamassa järjestyksessä, mikä tuo joustavuutta haastatteluun. Valitsimme tutkimukseen henkilöitä, joilla tiesimme olevan paljon tietoa tutkittavasta aiheesta, mikä on laadullisen tutkimuksen kannalta tärkeää. (Hirsjärvi ym., 2009, 205; Tuomi & Sarajärvi, 2018, 83–86.)

### 6.1.3 Kvantitatiivinen osuus

Kvantitatiivinen eli määrällinen tutkimusmenetelmä pohjautuu tilastotieteen käyttöön eli numeeriseen aineistoon ja sen analysointiin. Nykyään määrällisen tutkimuksen tekeminen on matemaattisesti vaivatonta, koska tietokoneet hoitavat laskemisen silmänräpäyksessä. Tutkijan tehtäväksi jää aineistonkeruun lisäksi kuitenkin laskelmien ja tilastojen analysointi sekä raportointi ja johtopäätökset. Vallin (2015) mukaan tilastollisessa tutkimuksessa tärkeintä on aihesisällön tunteminen ja oikea tulkinta sekä niiden edellyttämien menetelmien ymmärtäminen.

Tämän tutkimuksen kvantitatiivinen aineisto koostuu kahden kustannusyhtiön matematiikan oppimateriaalien tehtävien kokonaislukumääristä, ohjelmointitehtävien lukumääristä, algoritmista ajattelua kehittävien ja tukevien tehtävien lukumääristä sekä toiminnallisten ohjelmointitehtävien ja algoritmista ajattelua kehittävien ja tukevien tehtävien lukumääristä. Ohjelmointitehtävien ja algoritmista ajattelua kehittävien ja tukevien tehtävien lukumääriä verrattiin tehtävien kokonaismäärään, jolloin saatiin niiden suhteelliset osuudet kokonaisuudesta selville.

Tutkimuksemme määrällinen osuus liittyy siis matematiikan oppimateriaaleissa esiintyvien tehtävätyyppien (kaikki tehtävät, ohjelmointitehtävät ja algoritmista ajattelua kehittävät ja tukevat tehtävät) esiintyvyyteen eli niiden lukumääriin ja suhteellisiin osuuksiin. Lisäksi ohjelmointitehtävistä ja algoritmista ajattelua kehittävästä ja tukevista tehtävistä on laskettu erikseen toiminnalliset tehtävät. Kvantitatiivinen osuus antaa lukijalle selkeän kuvan numeroin ilmaistuna, kuinka paljon kahden kirjakustantajan, yhteensä neljän matematiikan oppimateriaalisarjan tehtävistä sisältää varsinaisia ohjelmointitehtäviä ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä. Lisäksi näiden tehtävien osalta selviää, kuinka monta niistä on luokiteltu toiminnallisiksi.

## 6.2 Aineistonkeruu ja tutkimusaineistot

Tutkimuksemme sisältää sekä laadullista että määrällistä aineistoa. Laadullinen aineisto on kerätty kahdelta tutkimukseen osallistuneelta luokanopettajalta ja määrällinen aineisto koostuu oppimateriaalien kustantajilta saaduista aineistoista. Kuvaamme seuraavaksi laadulliseen ja määrälliseen aineistoon liittyviä asioita erikseen.

### *Laadullisen aineiston tutkittavat ja aineistokeruu*

Kartoitimme kvalitatiivista tutkimusotostamme varten luokanopettajia, joilla oli kokemusta ohjelmoinnin opettamisesta alakoulussa. Otimme yhteyttä luokanopettajiin sähköpostitse ja saimme kahdelta kontaktoidulta myöntyvän vastauksen osallistumisesta tutkimukseen. Haastattelupyyntöihin liittyvät asiakirjat löytyvät liitteistä: saatekirje (liite 5), tutkimustiedote (liite 6) ja tietosuojailmoitus (liite 7).

Tutkimuksessamme haastateltiin kahta asiantuntijaopettajaa sekä tutkittiin Otava Oy:n ja Sanoma Pro Oy:n matematiikan oppimateriaaleja. Asiantuntijaopettajalla tarkoitamme opettajaa, joka on luokanopettajan työssään opettanut ohjelmointia useamman vuoden ajan osana matematiikan oppiainetta ja hänellä on runsaasti kokemuksia toiminnallisuudesta ohjelmoinnin opetuksessa. Molemmilla haastateltavilla oli runsaasti kokemusta luokanopettajatyöstä sekä ohjelmoinnin opettamisesta. Toinen haastateltava työskenteli haastatteluhetkellä luokanopettajakouluttajana. Ohjelmointikoulutuksen haastateltavat olivat saaneet erilaisten ohjelmointiin liittyvien LUMA-koulutusten ja oman perehtymisen kautta. Toisella haastateltavalla oli lisäksi Applen tarjoama koulutus ja hän toimi myös itse Apple-kouluttajana. Taulukossa 3 on esitelty tutkimuksemme osallistuneiden luokanopettajien perustiedot.

### Taulukko 3

#### *Tutkimuksen haastateltavat*

Haastateltava	Työkokemus	Ohjelmoinnin opetuskokemus	Ohjelmointikoulutus
H1	yli 20 v, LO 2 v, LO-koulutus	7 v	omatoiminen perehtyminen, LUMA-koulutukset, Apple-koulutukset
H2	alle 10 v, LO	9 v	omatoiminen perehtyminen, LUMA-koulutukset, LUMATIikka- ohjelmointikurssi (15 op)

Haastattelujen tarkoituksena oli selvittää opettajien kokemuksia ohjelmoinnin opettamisesta sekä matematiikan oppimateriaalien tarjoamasta ohjelmoinnin opetuksesta ja siitä, miten he ovat kokeneet toiminnallisuuden osana sitä.

Tutkimuksessa voidaan haastattelumenetelmänä käyttää lomakehaastattelua, avointa haastattelua tai teemahaastattelua (Hirsjärvi ym., 2009, 29). Päädyimme valitsemaan haastattelumenetelmäksi teemahaastattelun. Se on puolistrukturoitu haastattelu, jossa edetään valittujen teemojen ja tarkentavien kysymysten mukaan. Teemahaastattelun etuina voidaan nähdä se, että kysymyksiä pystytään täsmentää ja syventää saatujen vastausten perusteella. Puolistrukturoitu teemahaastattelu sopi tässä tutkimuksessa haastattelumenetelmäksi, koska halusimme saada tutkittavien äänen kuuluviin mahdollisimman hyvin. (Hirsjärvi & Hurme, 2008, 47–48.) Kävimme läpi tutkimushaastattelun eri vaihtoehdot ja pohdimme niiden soveltuvuutta tämän tutkimuksen tiedonhankinnassa. Teimme haastatteluja varten haastattelurungon, johon oli kirjattu pääkysymyksiä eri teemoista sekä muita

tarkentavia kysymyksiä, jotka jaoin tutkimukseen osallistuville tutustuttavaksi etukäteen varmistaaksemme mahdollisimman kattavan ja laajan aineiston. Haastattelun runko löytyy liitteestä 4. Puolistrukturoidussa haastattelussa kysymykset olivat samat molemmille haastateltaville, mutta kumpikin sai vastata niihin omin sanoin (Eskola & Suoranta, 2014, 87).

Haastattelujen aineistonkeruu tapahtui toukokuussa 2022 ja ne toteutettiin yksilöllisesti haastateltavien työpaikoilla koulun omissa tiloissa, koska halusimme paikan olevan rauhallinen ja turvallinen haastateltaville. Haastattelutilanteet olivat luontevia ja tutkittavat vastasivat kysymyksiin avoimesti ja hyvinkin laajasti, sillä he olivat saaneet haastattelurungon jo etukäteen tutustuttavaksi. Hirsjärven ja Hurmeen (2008, 74) mukaan teemahaastattelu myös edellyttää hyvää kontaktia tutkittavaan. Pidimme huolta siitä, että tutkittavien oli helppo lähestyä meitä ja kysyä, jos mieltä askarrutti jokin asia. Haastattelujen yhteydessä pyysimme opettajilta suostumuksen haastatteluun ja annoimme kirjallisesti lupauksen siitä, että aineistoa käsitellään luotettavia tutkimuskäytäntöjä noudattaen. Haastattelut nauhoitettiin äänityslaitteella. Haastattelumenetelmänä puolistrukturoitu teemahaastattelu oli hyvä, sillä saimme monipuolista aineistoa ja pystyimme tarkentamaan kysymyksiä epäselvyyksien välttämiseksi, mikä vahvisti aineiston luotettavuutta ja sen riittävyttä. Ennen aineiston analyysin aloittamista haastatteluaineisto litteroitiin eli kirjoitettiin puhtaaksi tekstinkäsittelyohjelmalla. Litteroitua haastatteluaineistoa kertyi yhteensä 24 A4-sivua (Book Antiqua, fonttikoko 12, riviväli 1,15) Haastattelut kestivät yhteensä kaksi tuntia ja 15 minuuttia. Litterointivaiheessa tutkittavista käytettiin tunnistekoodia H1 ja H2 anonyymiteetin säilyttämiseksi.

#### *Määrällisen aineiston keruu ja kuvaus*

Tutkimuksemme määrällinen aineisto koostuu kahden suomalaisen kustantajan matematiikan oppimateriaaleista. Aineiston muodostavat Sanoma Pro Oy:n 1–4 vuosiluokkien Kymppi-oppimateriaalit ja 1–4 vuosiluokkien Milli-oppimateriaalit sekä Otava Oy:n 1–3 vuosiluokkien Tuhattaituri-oppimateriaalit



sekä 1- ja 3-luokkien Oivaltaja-oppimateriaalit. Kaikki oppimateriaalit sisältävät oppilaiden oppikirjat ja opettajan oppaat. Oppimateriaalit sisältävät omat versionsa vuosiluokkien syys- ja kevätlukukausille. Oppimateriaalit poikkeavat toisistaan sisällöllisesti, aiheiden käsittelyn järjestyksessä ja opetustavoissa. Tutkimuksemme tarkoitus ei ole arvottaa tai arvostella eri oppimateriaaleja keskenään, vaan pelkästään selvittää, miten ja kuinka paljon ohjelmointia opetetaan niissä.

Seuraavaksi esittelemme tutkimuksemme aineiston oppimateriaalit. Jokaisesta kirjasarjasta tarkasteltiin oppilaan oppikirjaa ja opettajan opasta. Kirjasarjoista on avattu rakenteita ja kvantitatiivisen aineiston keruuseen liittyviä ominaispiirteitä sekä esitelty jokaisesta kirjasarjasta tyypillisiä ohjelmointitehtäviä ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä niin oppilaan oppikirjan kuin opettajan oppaan sisällöistä. Oppimateriaalien esimerkkitehtävät perustuvat teorian pohjalta nousseeseen luokittelurunkoon, jonka olemme laatineet algoritmista ajattelua kehittävien ja tukevien tehtävätyyppien pohjalta. Tehtävätyypit ovat: ongelmanratkaisu ja looginen päättely, toimintaohjeen noudattaminen, suunta- ja sijaintikäsite ja kielentämistehtävät. Luokittelurungon esittelemme tarkemmin aineiston analyysiluvussa. Luettavuuden selvennykseksi mainittakoon, että oppikirjaesimerkeissä näkyvät myös malliratkaisut.

#### *Kymppi-oppimateriaali (Sanoma Pro Oy)*

Kymppi-oppimateriaalista aineistossamme ovat materiaalit vuosiluokilta 1-4. Jokaiselle vuosiluokalle on kaksi osaa (Syksy ja Kevät) eli Kymppi-sarja sisältää yhteensä kahdeksan oppimateriaalikonaisuutta. Kustantajan ohjeistuksen mukaan Syksy on tarkoitettu suoritettavaksi syyslukukaudella ja Kevät kevätlukukaudella. Kymppi-sarja on jaettu teemoittain jaksoihin ja jokaisesta oppikirjasta löytyvät varsinaiset ohjelmointitehtävät "Kohti koodausta" -jaksoista (1.-2. luokat) ja "Tutustutaan koodaukseen" -jaksoista (3.-4. luokat). Lisäksi kirjoista löytyvät erilliset kertausjaksot.

Kymppi-sarjassa toistuu lähes jokaisen jakson päätteeksi

kappale ”Testataan ja toimitaan”, joista löytyy myös algoritmiseen ajatteluun liittyviä toiminnallisia tehtäviä. Ohjelmointia tukevia tehtäviä on tasaisesti jaksosta riippumatta, painottuen kuitenkin esimerkiksi geometrian osuuksiin ja ”Testataan ja toimitaan”-kappaleisiin. Opettajan oppaasta löytyy kappalekohtaisesti ”Toiminnallisia harjoituksia” -osio, joista osa on myös ohjelmointiin liittyviä.

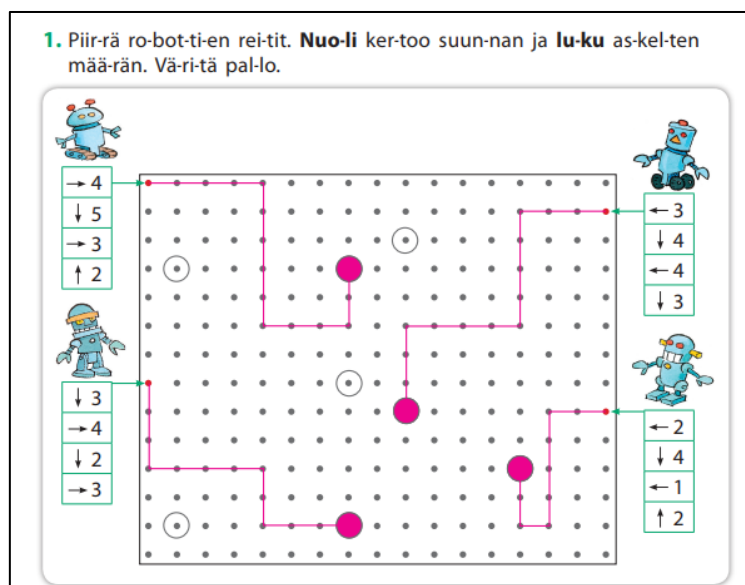
Kymppi-sarjan tehtävät ovat numeroitu juoksevin numeroin. Saman juoksevan numeron alla voi olla useampi osatehtävä, mutta tutkimuksen numeeriseen aineistoon on laskettu vain juoksevan numeroinnin mukaiset tehtävät.

Kymppi-oppimateriaalin opettajanoppaan osasta tutkimukseen on laskettu mukaan ohjelmointiin ja algoritmista ajattelua tukevia tehtäviä ja harjoitteita kohdista ”Toiminnallisia harjoituksia” ja ”Pohdittavaa”. Molemmissa osissa on runsaasti ongelmanratkaisutaitoa ja loogista päättelykykyä kehittäviä tehtäviä sekä toimintaohjeiden noudattamista harjoittavia tehtäviä. ”Toiminnallisia harjoituksia”-osan tehtävistä lähes kaikki ovat toiminnallisia.

Kymppi-oppimateriaalissa varsinaisia ohjelmointitehtäviä löytyy ”Kohti koodausta” - ja ”Tutustutaan koodaukseen”-kappaleista (kuva 1).

## Kuva 1

*Robotin ohjaus (Kymppi 1 kevät, 192)*



Opettajanoppaan ”Toiminnallisia harjoituksia” -osuuden tehtävät ovat lähes kaikki toiminnallisia (kuva 2).

## Kuva 2

*Ihmisrobotin ohjaus (Kymppi 1 kevät opettajanopas, 125)*

**Robottinäytös**

- Viisi vapaaehtoista robottia seisoo kasvot muihin oppilaisiin päin.
- Numeroidaan robotit 1–5.
- Opettaja antaa aluksi malliksi käskyjä ja oppilaat katsovat, toimivatko robotit oikein.
  - nosta oikea käsi ylös
  - nosta vasen käsi sivulle
  - vie oikea polvi maahan
  - käännä pää vasemmalle
  - pyörähdä puoli kierrosta ympäri.
- Sitten joku oppilaista saa antaa roboteille käskyjä ja muut tarkkailevat.
- Kun joku roboteista tekee väärin, oppilaat saavat sanoa esim.: Robotti 3, virhe.
- Vaihdetaan myös robotteja välillä.

Algoritmista ajattelua tukevia tehtäviä Kymppi-sarjassa on oppilaan oppikirjassa ja opettajan oppaassa lähes jokaisessa kappaleessa. Esimerkiksi oppilaan oppikirjassa harjoitellaan ongelmanratkaisutaitoja (kuva 3) ja opettajanoppaan puolella pohditaan säännön mukaisuutta (kuva 4).

### Kuva 3

Ongelmanratkaisutehtävä (Kymppi 3 syksy, 57)

**2. Ketä tarkoitan?**

Meri	Mikko	Milla
Minttu	Miro	Niilo
Mikael	Matilda	Nea

- Hänen nimensä alkaa M-kirjaimella.
- Nimessä on enemmän kuin neljä kirjainta.
- Nimessä on a-kirjain.
- Hän ei ole poika.
- Nimessä ei ole d-kirjainta.

v: Milla

Veeti	Veera	Leevi
Viivi	Luukas	Lotta
Ville	Lilja	Väinö

- Nimessä on viisi kirjainta.
- Nimessä on kaksi samaa kirjainta.
- Nimessä on i-kirjain.
- Hän ei ole tyttö.
- Hänen nimensä alkukirjain ei ole V.


v: Leevi

### Kuva 4

Säännönmukaisuuden ratkaiseminen (Kymppi 3 syksy, opettajanopas, 55)

Kuviojono jatkuu saman säännön mukaan. Minkä muotoisia ja värisiä ovat seuraavat kolme kuviota?

1. 

(  Joka toinen kuvio on sininen ja joka toinen punainen. Kahta neliötä seuraa aina yksi ympyrä.)

---

2. 

(  Joka kolmas kuvio on punainen. Joka toinen kuvio on kolmio ja joka toinen neliö.)

Milli (Sanoma Pro Oy)

Milli-oppimateriaalista aineistossamme on materiaalit vuosiluokilta 1–4. Jokaiselle vuosiluokalle on kaksi osaa (A ja B) eli Milli-sarja sisältää yhteensä kahdeksan oppimateriaalikokonaisuutta. Kustantajan ohjeistuksen mukaan A on tarkoitettu suoritettavaksi syyslukukaudella ja B kevätlukukaudella. Milli-sarja

on jaettu teemoittain jaksoihin ja jokaisesta oppikirjasta löytyvät varsinaiset ohjelmointitehtävät "Ohjelmointia" -jaksosta. Erillinen kertausjakso löytyy 3. ja 4. vuosiluokan oppikirjoista.

Milli-sarjassa jaksot päättyvät yleensä "Katsastus"- ja "Projekti"-kappaleisiin, joista löytyy myös algoritmista ajattelua kehittäviä ja tukevia tehtäviä. "Katsastus"-kappaleen itsearviointia ei ole huomioitu ja "Projekti"-kappaleen jokainen tehtävä on erikseen laskettu.

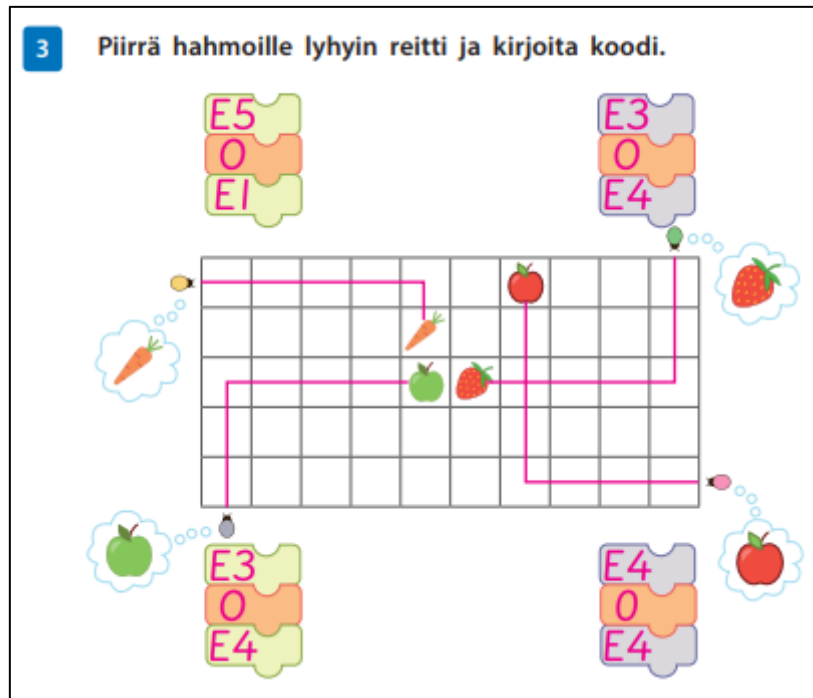
Milli-sarjan tehtävät on numeroitu juoksevin numeroin. Saman juoksevan numeron alla voi olla useampi osatehtävä, mutta tutkimuksen numeeriseen aineistoon on laskettu vain juoksevan numeroinnin mukaiset tehtävät. Poikkeuksena Millimittaritehtävät, joita ei ole numeroitu juoksevin numeroin. Millimittaritehtävä on laskettu mukaan yhtenä tehtävänä riippumatta sen osatehtävien määrästä.

Milli-oppimateriaalien opettajanoppaan osasta tutkimukseen on laskettu mukaan ohjelmointiin ja algoritmista ajattelua tukevia tehtäviä ja harjoitteita kohdista "Harjoittelu" ja "Pulmat". Molemmista osista löytyy runsaasti ongelmanratkaisutaitoa ja loogista päättelykykyä kehittäviä tehtäviä sekä toimintaohjeiden noudattamista harjoittavia tehtäviä. "Harjoittelu"-osan tehtävistä lähes kaikki ovat toiminnallisia.

Milli-oppimateriaalissa varsinaisia ohjelmointitehtäviä löytyy "Ohjelmointia"-jaksosta (kuva 5). Opettajanoppaan "Harjoittelu" -osuuden tehtävät ovat lähes kaikki toiminnallisia, joissa harjoitellaan esimerkiksi vaiheittaisen toimintaohjeen eli algoritmin muodostamista (kuva 6).

## Kuva 5

Koodin kirjoittamista (Milli 2A, 205)



## Kuva 6

Toimintaohjeen laatiminen (Milli 2A, opettajanopas, 129)

### Vaiheittaisen toimintaohjeen laatiminen vesilasillisen juomiseen

Asetetaan juomalasit johonkin näkyvään paikkaan luokassa. Oppilaat työskentelevät pareittain. Heidän tehtävänä on muodostaa vaiheittainen toimintaohje vesilasillisen juomiseen. Ensimmäisessä työskentelyn vaiheessa he laativat ohjeen ilman toimintaohjeen testaamista. Tarvittaessa näkyvisä voi olla myös sanoja, joilla ohjeen voi aloittaa, esimerkiksi: tartu, nosta, pidä kiinni, laita, avaa jne. Kun oppilaspari on saanut ohjeensa valmiiksi, tulee heidän testata laatimaansa ohjetta. Toimintaohjeen testaamisen aikana ja perusteella ohjetta muokataan tarvittavalla tavalla. Lopuksi verrataan oppilasparien laatimia ohjeita. Mitkä asiat ovat samalla tavalla ohjeistettu ja mitä eroja ohjeissa on? Lyhennetään yksi toimintaohje mahdollisimman yksinkertaiseksi yhdessä.

Algoritmista ajattelua tukevia tehtäviä Milli-sarjassa on oppilaan oppikirjassa ja opettajan oppaassa lähes jokaisessa kappaleessa. Esimerkiksi oppilaan oppikirjassa harjoitellaan loogista päättelykykyä (kuva 7) ja opettajanoppaan puolella harjoitellaan ongelmanratkaisua vaiheittain (kuva 8)

### Kuva 7

*Loogista päättelykykyä kehittävä tehtävä (Milli 4B, 77)*

**8** Koulussa pelataan luokkien välinen polttopalloturnaus. Päättele oppilaiden tekemien polttoheittojen määrät.

- Eena polttaa 8 pelaajaa enemmän kuin Olli.
- Kati polttaa 7 pelaajaa vähemmän kuin Olli.
- Jalo polttaa kaksi pelaajaa enemmän kuin Kati.
- Jalo ja Kati polttavat yhteensä 12 vastustajaa.

Eena 20   Kati 5   Olli 12   Jalo 7

### Kuva 8

*Ongelmanratkaisua vaiheittain (Milli 4B, opettajanopas, 43)*

**Ongelmanratkaisua**

Harjoitellaan monivaiheisten tehtävien ratkaisemista. Menetelmässä kannattaa noudattaa esimerkiksi viisiportaista ratkaisumallia.

1. Lue tehtävä.
2. Alleviivaa **kysymys**.
3. Ympyröi tehtävän kannalta **oikeat luvut**.
4. Kirjoita laskut, joko yhtenä lausekkeena tai useampana vaiheena.
5. Laske laskut ja vastaa kysymyseen.

Tehtävien ratkaisemisessa on kiinnitettävä huomiota oleellisiin asioihin:

- Mihin halutaan vastaus?
- Mitä tietoja on annettu?
- Millä laskuilla tehtävän voi ratkaista?

Ratkaistaan yhdessä esimerkkitehtävä:

- Iso tynnyri painaa 300 kg, josta puolet on hiekan painoa. Pieni tynnyri painaa 120 kg, josta puolet on hiekan painoa. Kuinka paljon tynnyreissä oleva hiekka painaa yhteensä?

(300 kg : 2 + 120 kg : 2  
= 150 kg + 60 kg  
= 210 kg  
V: 120 kg)

*Tuhattaituri (Otava Oy)*

Tuhattaituri-oppimateriaalista aineistossamme on materiaalit vuosiluokilta 1–3. Jokaiselle vuosiluokalle on kaksi osaa (a ja b) eli Milli-sarja sisältää yhteensä kuusi oppimateriaalikokonaisuutta. Kustantajan ohjeistuksen mukaan a on tarkoitettu suoritettavaksi syyslukukaudella ja b kevätlukukaudella. Tuhattaituri-sarja on jaettu teemoittain jaksoihin. Tuhattaituri-sarjaan ei sisälly erityistä ohjelmointijaksoa, jossa olisi varsinaisia ohjelmointitehtäviä, vaan ohjelmointitehtävät ovat sisällytetty muiden teemojen alle, etenkin geometriateemaan. Lisäksi sarjaan on merkitty erikseen, mikäli tehtävä on ohjelmointia tukeva tehtävä.

Tuhattaituri-sarjassa on erilliset jaksojen päättävät ”Tähtipysäkki”-kokonaisuudet, jaksokohtaiset ”Toimintatunti”-kappaleet sekä kertaavat ”Tähtitehtävät”- ja ”Taitorastit”-kappaleet. Kolmannen vuosiluokan oppikirjoista löytyy myös erillinen toiminnallinen ”Projektitehtävät”-osio.

Tuhattaituri-sarjan tehtävät on numeroitu juoksevin numeroin. Osa tehtävistä sisältää osatehtäviä, jotka ovat nimetty kirjaimin (a, b, c...). Nämä tehtävät ovat laskettu yhtenä tehtävänä juoksevan numeroinnin perusteella. Ensimmäisen vuosiluokan oppikirjoissa kotitehtävätösköön tehtävät eivät ole numeroitu ja nämä on laskettu yhtenä tehtävänä osatehtävien lukumäärästä riippumatta. Opettajan oppaassa on ”Toiminta”-kohta, josta löytyy myös ohjelmointiin liittyviä tai algoritmista ajattelua tukevia tehtäviä.

Tuhattaituri-oppimateriaalien opettajanoppaan osasta tutkimukseen on laskettu mukaan ohjelmointiin ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä ja harjoitteita kohdista ”Toiminta” ja ”Pulmat”. ”Toiminta”-osiossa on paljon ongelmanratkaisutaitoa ja loogista päättelykykyä kehittäviä tehtäviä sekä toimintaohjeiden noudattamista harjoittavia tehtäviä. ”Toiminta”-osion tehtävistä lähes kaikki ovat toiminnallisia.

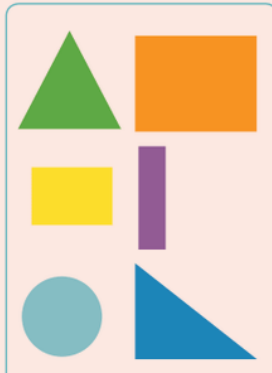
Tuhattaituri-oppimateriaalissa varsinaisia ohjelmointitehtäviä löytyy ”Ohjelmointia”-kappaleesta (kuva 9). Opettajanoppaan ”Toiminta” -osuuden tehtävät ovat lähes kaikki toiminnallisia, joissa harjoitellaan esimerkiksi ihmisrobotin avulla koodin purkamista (kuva 10).









## Kuva 9

Toimintaohjeen mukaan toiminta (Tuhattaituri 1b, 109)

**5. Piirrä ku-vi-o ruu-tuun oh-jeen mu-kaan. Vä-ri-tä oi-kein.**



- Vih-re-än kol-mi-on a-la-puo-lel-la on 
- Tum-mansi-ni-sen kol-mi-on y-lä-puo-lel-la on 
- Kel-tai-sen ne-li-kul-mi-on oi-ke-al-la puo-lel-la on 
- O-rans-sin ne-li-kul-mi-on va-sem-mal-la puo-lel-la on 
- A-lim-man vaa-ka-ri-vin vii-me-se-nä on 
- Oi-ke-an pysty-ri-vin y-lim-pä-nä on 

## Kuva 10

Koodin purkaminen ihmisrobotin avulla (Tuhattaituri 1b, opettajanopas, 107)

**3. Koodin purkaminen**

Tarvikkeet: hattu sekä 20–30 korttia, joissa numerot 1–5. Toimitaan pareittain tai luokkana. Oppilaille esitellään, mitä toimintoa numerot tarkoittavat. Valitaan yksi robotti. Robotti erottuu muista vaikkapa hatun avulla. Leikin vetäjällä on korttiviuhka, joista arvotaan 4 korttia. Ne annetaan robotille. Robotti lähtee toimimaan kortissa olevien numeroiden mukaan. Muiden tehtävänä on purkaa robotin toistama koodi.

- 1 – lantion keinutus
- 2 – pään pyöritys
- 3 – sormien napsutus
- 4 – jalkojen tömistys
- 5 – huuto: "Jee-jee!"

Algoritmista ajattelua kehittäviä ja tukevia tehtäviä Tuhattaituri-sarjassa on oppilaan oppikirjassa ja opettajan oppaassa lähes jokaisessa kappaleessa. Opettajan oppaasta löytyy myös erikseen listaus tehtävistä, jotka on luokiteltu ohjelmoinnista ajattelua tukeviksi tehtäviksi. Esimerkiksi oppilaan oppikirjassa harjoitellaan salakirjoitussymboliikan avaamista (kuva 11) ja opettajanoppaan puolella kehitetään loogista päättelykykyä (kuva 12).

## Kuva 11

Salakirjoituksen avaaminen (Tuhattaituri 1b, 121)

8. Selvi-tä Kas-sen ja Kur-ren sa-la-kieli.



(1,1)	(3,4)	(2,5)	(2,5)	(5,2)
K	U	R	R	E

(5,5)	(3,4)	(1,3)	(5,2)
T	U	L	E

(5,5)	(2,2)	(4,3)	(4,3)	(5,2)
T	Ä	N	N	E

(5,5)	(3,4)	(1,3)	(4,1)	(0,6)	(0,6)	(6,4)
T	U	L	O	S	S	A

(4,1)	(1,3)	(1,3)	(6,4)	(6,4)	(4,3)
O	L	L	A	A	N

## Kuva 12

Looginen päättelykyky (Tuhattaituri 2a, opettajanopas, 65)

### 5. Arvaa salainen lukuni

Laitetaan taululle lukujono 0–100. Kaksi oppilasta tulee luokan eteen ja kumpikin valitsee lukusuoralta oman luvun. He antavat vihjeitä omasta luvustaan, esim. "Minun luvussani on kymmeniä 6." "Minun lukuni on suurempi kuin  $58 + 5$ ." "Minun lukuni on pienempi kuin  $70 - 3$ ." Muut oppilaat arvaavat luvut.

*Oivaltaja (Otava Oy)*

Oivaltaja-oppimateriaalista aineistossamme on materiaalit vuosiluokilta 1 ja 3. Molemmille vuosiluokille on kaksi osaa (a ja b) eli Oivaltaja-sarja sisältää yhteensä neljä oppimateriaalikokonaisuutta. Kustantajan ohjeistuksen mukaan a on tarkoitettu suoritettavaksi syyslukukaudella ja b kevätlukukaudella. Oivaltaja-sarja on jaettu teemoittain jaksoihin. Oivaltaja-sarjasta löytyy varsinaisia ohjelmointitehtäviä "Oivallus"-jakson yhteydestä "Ohjeita ohjelmointiin"-kappaleesta. Lisäksi sarjasta löytyy erikseen "Hoksaus"-jakso, josta löytyy ohjelmointia tukevia tehtäviä. Oivaltaja-sarjassa kappaleen päättää

kotitehtävätyyppinen ”Palauta mieleen”-osio. Jokaisen jakson päättää kertauskappale.

Oivaltaja-sarjan tehtävät on numeroitu juoksevin numeroin. Osa tehtävistä sisältää osatehtäviä, jotka ovat nimetty kirjaimin (a, b, c...). Nämä tehtävät ovat laskettu yhtenä tehtävänä juoksevan numeroinnin perusteella.

Oivaltaja-oppimateriaalien opettajanoppaan osasta tutkimukseen on laskettu mukaan ohjelmointiin ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä ja harjoitteita kohdista ”Opettavat toimintavinkit” ja ”Harjoittavat toimintavinkit”. Molemmista osista löytyy runsaasti ongelmanratkaisutaitoa ja loogista päättelykykyä kehittäviä tehtäviä sekä toimintaohjeiden noudattamista harjoittavia tehtäviä. Osioden tehtävistä lähes kaikki ovat toiminnallisia.

Opettajanoppaan ”Toiminta” -osuuden tehtävät ovat lähes kaikki toiminnallisia, joissa harjoitellaan esimerkiksi ihmisrobotin avulla koodin purkamista (kuva 10).

”Ohjeita ohjelmointiin”-kappaleessa tutkitaan esimerkiksi lohkokoodauskielen ohjelmaa ja pyritään ymmärtämään sen toimintaa (kuva 13).

### Kuva 13

*Ohjelmakoodin tutkiminen (Oivaltaja 3a, 114)*

**1 Tutki ohjelmaa.**

**a.**

<div style="background-color: #f9e79f; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">odota 5 sekuntia</div>	Kuinka monta käskyä ohjelmassa on? <u>2</u>
<div style="background-color: #d9ead3; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">liiku 5 askelta</div>	Kuinka monta sekuntia 🐱 odottaa? <u>5</u>
	Kuinka monta askelta 🐱 liikkuu? <u>5</u>

**b.**

<div style="background-color: #f9e79f; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">liiku 3 askelta</div>	Kuinka monta käskyä ohjelmassa on? <u>5</u>
<div style="background-color: #d9ead3; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">odota 2 sekuntia</div>	Kuinka monta sekuntia 🐱 odottaa yhteensä? <u>4</u>
<div style="background-color: #d9ead3; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">liiku 3 askelta</div>	Kuinka monta askelta 🐱 liikkuu yhteensä? <u>9</u>
<div style="background-color: #d9ead3; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">odota 2 sekuntia</div>	
<div style="background-color: #d9ead3; padding: 2px; border: 1px solid #ccc; border-radius: 5px;">liiku 3 askelta</div>	

Opettajanoppaan ”Harjoittava toimintavinkki”-osiosta löytyy toiminnallisia harjoituksia, esimerkiksi ihmisrobotin ohjausta (kuva 14):

## Kuva 14

*Ihmisrobotin ohjaus (Oivaltaja 3a, opettajanopas, 68)*

### Liikkeestä ohjelmaksi

**Tarvikkeet:** turvallinen tila, esimerkiksi välituntialue

Toimitaan pareittain. Toinen parista on ohjelmoija ja toinen robotti.

1. Robotti suorittaa ensin 4 keksimäänsä käskyä.

Sopivia käskyjä ovat esimerkiksi:

- Odota 5 sekuntia.
- Sano hei.
- Liiku 4 askelta.
- Käänny vasempaan.

2. Ohjelmoija tarkkailee robotin toimintaa ja yrittää kääntää toiminnan käskyiksi.

3. Lopuksi pari tarkistaa käskyt ja vaihdetaan rooleja.

Algoritmista ajattelua kehittäviä ja tukevia tehtäviä Oivaltaja-sarjassa on oppilaan oppikirjassa ja opettajan oppaassa lähes jokaisessa kappaleessa. Esimerkiksi oppilaan oppikirjassa harjoitellaan ongelmanratkaisua ja loogista päättelyä salakirjoitussymboliikan avulla (kuva 15) ja opettajanoppaan puolella harjoitellaan toimintaohjeiden noudattamista tarinan avulla (kuva 16).

## Kuva 15

*Ongelmanratkaisua ja loogista päättelyä (Oivaltaja 1a, 81)*

**3** Päät-te-le. Mi-tä kirjai-mi-a ku-vi-ot tarkoit-ta-vat?  
Kirjoi-ta last-en ni-met.

IINA				
ANNI				
EINO				
ENNA				

## Kuva 16

*Toimintaohjeen noudattaminen tarinan avulla (Oivaltaja 1a, 64)*

### Vähennetään 0, 1 tai 2

A. Opettaja kertoo vähennyslaskusta matikkatarinan. Oppilaat esittävät tarinan vaiheittain. Ensimmäisessä seisoo vähenevän verran oppilaita (alussa on). Sitten oppilaita lähtee pois vähentäjän verran (otetaan pois). Lopuksi lasketaan, kuinka monta jää jäljelle (jäljelle jää). Kirjoitetaan lasku näkyviin.

#### Esimerkkitarinat:

- Jalkapallokentällä on 4 pelaajaa. 1 pelaaja lähtee pois. Kuinka monta pelaajaa jää jäljelle? ( $4 - 1 = 3$ )
- Epulla on 6 jalkapalloa. Hän antaa niistä 2 pois. Kuinka monta jalkapalloa jää jäljelle? ( $6 - 2 = 4$ )

Esimerkin jälkeen oppilaat keksivät ja esittävät vähennyslaskutarinoita ryhmissä. Toiminnan voi tehdä myös värikiekoilla tai pienillä erilaisilla esineillä.

Edellä esitettyjen oppimateriaalikohtaisten esimerkkitehtävien lisäksi oppimateriaaleista löytyy runsaasti myös kielentämistehtäviä, jotka on laskettu mukaan algoritmista ajattelua kehittäviksi ja tukeviksi tehtäviksi. Lisäksi laskentaan on huomioitu pelit ja leikit, jotka kehittävät erityisesti (vaiheittaisten) toimintaohjeiden noudattamista ja ymmärtämistä. Pelejä ja leikkejä löytyy sekä oppilaan oppikirjojen että opettajanoppaiden puolelta.

## 6.3 Aineiston analyysi

Tärkeä osa tutkimusprosessia on aineiston analyysin suunnittelu ja toteuttaminen. Siinä tutkija pyrkii selvittämään, minkälaisia vastauksia hän sai tutkimuskysymyksiinsä. (Hirsjärvi ym., 2009, 221.) Seuraavaksi kuvaamme laadullisen ja määrällisen aineiston analyysit erikseen.

### *Laadullisen aineiston analyysi*

Haastatteluaineisto analysoitiin ja tulokset raportoitiin touko-heinäkuussa 2022. Analyysi tapahtui teorialähtöistä sisällönanalyysiä käyttäen, jolla pyrittiin tavoittamaan haastateltavien käsityksiä ja kokemuksia sekä niille annettuja merkityksiä. Sisällönanalyysissä tarkastellaan tekstimuotoista aineistoa (Tuomi & Sarajärvi, 2018, 117), joka tässä tutkimuksessa on litteroitu haastatteluaineisto. Tuomen ja Sarajärven (2018, 127) mukaan teorialähtöinen sisällönanalyysi

perustuu jo ennalta olevaan käsitejärjestelmään, joka voi olla esimerkiksi teoria.

Luimme litteroidun aineiston useaan kertaan läpi, jotta löysimme sieltä tutkimuksen kannalta merkittäviä ilmauksia. (Eskola, 2018, 219–221.) Teorialähtöinen sisällönanalyysi etenee hyvinkin samanlaisesti kuin aineistolähtöinen sisällönanalyysi. Teorian pohjalta on tarkoitus muodostaa kategorioita ja luokituksia. Analyysi jatkuu poimimalla haastatteluaineistosta analyysirungon mukaisia ilmauksia eli analyysiyksiköitä. Analyysiyksiköllä tarkoitetaan sanaa, lausetta tai useamman lauseen kokonaisuutta, mitä kutsutaan pelkistämiseksi. Tämän jälkeen analyysissä pelkistetyt ilmaukset lajitellaan analyysiluokkiin analyysirungon mukaisesti. (Tuomi & Sarajärvi, 2018, 128–129.)

Aloitimme aineiston analyysin kokoamalla analyysirungon. Tuomen ja Sarajärven (2018, 128) mukaan aineistosta muodostettu analyysirunko voi olla joko väljä tai tarkemmin strukturoitu. Väljällä analyysirungolla tarkoitetaan sitä, että aineistosta kerätään myös ne tiedot, jotka jäävät analyysirungon ulkopuolelle ja mistä tehdään uusia luokkia induktiivisen sisällönanalyysin mukaan. Käytimme tarkasti strukturoitua analyysirunkoa ja muodostimme haastattelukysymykset myös sen mukaisesti. Analyysirunkomme muodostui algoritmista ajattelua kehittävästä ja tukevasta tehtävätyypeistä, jotka nousivat esiin teoriasta (ks. 3.4 Algoritmisen ajattelu). Analyysirungon pääkategoriat ovat: ongelmanratkaisu ja looginen päättely, toimintaohjeen noudattaminen, suunta- ja sijaintikäsite sekä kielentämistehtävät. Nämä pääkategoriat on esitelty taulukossa 4.

## Taulukko 4

### Algoritmisen ajattelun kehittäminen

Algoritmisen ajattelun kehittäminen			
Ongelmanratkaisu ja looginen päättely	Toimintaohjeen noudattaminen	Suunta- ja sijaintikäsite	Kielentämistehtävät
tuntemattoman ratkaiseminen, hahmottaminen, luokittelu- ja sarjoittamistehtävät	pelit, leikit, vaihteelliset toimintaohjeet	avaruudellinen hahmottaminen	tarinasta lauseke, lausekkeesta tarina
<p>”Pitää antaa joku haaste, et ratkaise tämä ongelma ja sitten ne joutuu itse miettimään ja sitten tavallaan hoksata sekin, että ratkaisuja on monia päästä samaan lopputulokseen, sitten voi haastaa keksimään toisen tavan tehdä sama asia erilaisilla koodipalikka-yhdistelmillä tai keksitkö vähän helpomman tavan että missä olisi ehkä vähemmän bugeja. Keksi siis mahdollisimman yksinkertainen tapa ratkaista tämä asia mahdollisimman vähillä koodipalasoilla.” (H2)</p> <p>”Semmoiset tehtävät, joissa on jätetty sitä avoimuutta. Mun mielestä semmoiset tehtävät tukee parhaiten.” (H2)</p> <p>”Ongelmanratkaisutaidot on siis mun mielestä kaikista kivoimpia näissä tehtävissä. Tehdään niitä pieniä haasteita ja tietysti jokainen bugin ratkaisu on yhdenlainen ongelmaratkaisutehtävä, että mikä tässä on nyt ongelma kun se ei toimi.” (H1)</p>	<p>”Voisi olla ihan hyvä että sulla on joku ”korjaa koodi” tai joku semmoinen, että pitää tarkasti lukea jotain, että miksi tämä ei nyt toimi.” (H2)</p> <p>”Kun saadaan yhdellä käskyllä menemään pitkä rimpsu asioita läpi, yhdellä ainoalla napin painalluksella.” (H1)</p>	<p>”Minusta tämä suunta ja sijainti on nimenomaan sen ajattelun opettaminen ja ne kehittää tosi paljon, kun sitä joutuu aika pitkälle miettimään, että miten se homma toimii.” (H1)</p> <p>”Ne code.orgin tehtävät tai jopa ne kirjatehtävät, mitä siellä on tai sitten justiinnsa vaikka sillä Beebotilla tai Bluebotilla, missä on se matto tässä näin, että sä voit ajaa niinku ruudusta toiseen. Sitten myös se ihmisrobotin ohjelmointi, että sen tyyppiset missä on paljon juuri tätä kääntymistä ja kulkemista niin sen tyyppiset tehtävät tukee tuota.” (H2)</p>	<p>”Nämä on ollut tosi mahtavia, että on tarina, että meillä on tuolla vuorella semmoinen myrkyllinen alue ja sinne ei voi lentää, että sieltä pitää käydä pelastamassa äkkiä tyyppit ja sitten ohjelmoidaan täältä drone, joka menee ja laskeutuu, ja sit se nousee, kääntyy, ottaa kuvan ja lentää tänne. Ja se pitäisi kaikki tapahtua ja toteuttaa yhdellä napinpainalluksella.” (H1)</p> <p>”Ennen kaikkea, että näistä saa toiminnallisia kokonaisuuksia, joihin liittyy joku tarina, joka sitä kautta kiehtoo.” (H1)</p>

Taulukossa 4 algoritmisen ajattelun kehittäminen jakautuu neljään edellä mainittuun pääkategoriaan. Jokaisen pääkategorian alle on listattu kyseiseen kategoriaan liittyviä tehtävätyyppejä. Tehtävätyyppien jälkeen olemme esitelleet muutamia kategorian aiheeseen sopivia haastateltavien lainauksia.

#### *Määrällisen aineiston analyysi*

Oppimateriaaliaineiston sisällöissä, struktuureissa ja esitystavoissa on jonkin verran eroja kahden eri kustantajan välillä, mutta myös saman kustantajan eri kirjasarjojen oppimateriaaleissa. Määrällistä aineistoa laskettaessa on jouduttu tekemään päätöksiä esimerkiksi siitä, miten rajataan yksittäinen tehtävä. Osassa oppilaan oppikirjojen tehtävät on numeroitu selkeästi, osassa jokaista kappaleen tehtävää ei ole aina numeroitu. Tehtävät eivät myöskään ole keskenään tasavertaisia eri materiaalien välillä, sillä välillä yhden tehtävänumeron alla saattaa olla useampikin kohta merkattuna aakkosin, kun taas toisessa aineistossa jokaiselle tehtävälle on annettu oma kohtansa juoksevasta numeroinnista.

Varsinaiset ohjelmointiin liittyvät tehtävät löytyvät kirjasarjoista yleensä loppupuolelta, joskus sisällytettynä esimerkiksi geometriajakson yhteyteen. Näiden ohjelmointiin liittyvien tehtävien lisäksi matematiikan kirjoista löytyy runsaasti ohjelmointia ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä. Näitä ovat ongelmanratkaisua ja loogista päättelyä vaativat tehtävät, kielentämistehtävät sekä suunta- ja sijaintitaitoja ja toimintaohjeiden noudattamista kehittävät tehtävät (ks. taulukko 4). Esimerkkejä tehtävätyypeistä olemme esitelleet luvussa 6.2 kohdassa *Määrällisen aineiston keruu ja kuvaus*. Oppilaan oppikirjan ohjelmointitehtävät ovat kauttaaltaan ei-toiminnallisia tehtäviä, kun taas opettajan oppaan tehtävät ja vinkit oppitunneille ovat suurimmaksi osaksi toiminnallisia harjoituksia. Yhden kirjasarjan opettajan oppaassa on varsinaisten ohjelmointitehtävien lisäksi listattu myös erikseen oppikirjasta löytyvät ohjelmointia tukevat tehtävät.

Yhteensä aineistossa oli 26 oppilaan oppikirjaa ja 26 opettajanopasta. Oppilaan oppikirjoista ja opettajanoppaista on laskettu tehtävien



esiintymislukumäärät kokonaisuutena, varsinaiset ohjelmointitehtävät ja algoritmista ajattelua kehittävät ja tukevat tehtävät. Algoritmista ajattelua kehittäviin ja tukeviin tehtäviin laskimme kokonaisesiintymismäärät, emme siis erotelleet esiintymismääriä tehtävätyypeittäin. Lisäksi erikseen on laskettu toiminnallisuutta sisältävät ohjelmointitehtävät ja algoritmista ajattelua kehittävät ja tukevat tehtävät. Laskettuja tehtävämääriä on verrattu kokonaismääriin sekä oppilaan oppikirjoissa että opettajan oppaissa, jolloin on saatu selville niiden suhteelliset osuudet. Erilliset kertaavat lisätehtävämonisteet on rajattu tarkasteltavan aineiston ulkopuolelle. Niin ikään aineiston ulkopuolelle rajattiin toiston välttämiseksi digimateriaalit ja niiden sisältämät mahdolliset laulut, aloitustarinat ja videot.

#### **6.4 Eettiset ratkaisut**

Tutkimusta tehdessä tutkijan pitää kiinnittää huomiota tutkimuksen eettisiin ratkaisuihin. Tutkimustyössä rehellisyys, huolellisuus, tarkkuus ja avoimuus kuuluvat hyvään tieteelliseen käytäntöön. Näistä edellä mainituista eettisistä ratkaisuista on huolehdittava tutkimusprosessin tiedonhankinnasta raportointiin ja arviointiin saakka. (Tuomi & Sarajärvi, 2018, 149–151.)

Hankimme tutkimuksellemme tarvittavat tutkimusluvut hyvää tieteellistä käytäntöä noudattaen (TENK, 2012, 6). Pidimme myös huolta siitä, että tutkittavien tietosuoja ja yksityisyys pysyy salassa. Haastateltavien kommentteja käytettiin tulosluvussa paljon ja haastateltavat koodattiin tunnisteilla H1 ja H2 anonymiteetin säilyttämiseksi. Tässä tutkimuksessa ei ole kyse arkaluontoisesta aiheesta, mutta on tärkeä tiedostaa, että henkilökohtaisten käsitysten ja ajatusten paljastaminen voi olla arkaluontoista tutkittavalle. Hirsjärven, Remeksen ja Sajavaaran (2009, 23–25) mukaan tutkimukseen osallistuvilla on oikeus tietää mihin haastatteluaineistoa käytetään. Tutkijan on tärkeä pitää mielessä, että tutkittavia ei saa loukata tai heille ei saa aiheuttaa minkäänlaista vahinkoa. Saatekirjeessä (liite 5) kerroimme tutkittaville, mitä tutkimuksemme koskee ja millaisia tietoja tulemme kysymään sekä mainitsimme tietojen hävittämisestä

tutkimuksen valmistuttua. Korostimme myös tutkimukseen osallistumisen vapaaehtoisuutta. Tutkijan tulee tavoitella objektiivisuutta tutkimusta tehdessään, vaikkei se ole aina kovin yksiselitteistä. Tutkijan on asianmukaista tarkastella tuloksia tasapuolisesti ja etäännyä tutkija-subjektistaan, jotta aineiston analyysi ei kärsi ja tutkijan ilmaisut ovat näkymättömissä. (Hirsjärvi ym., 2009, 310.) Pidimme tästä huolta tutkimusta tehdessä, sillä toisella tutkijoista on ohjelmointiaiheeseen liittyvää koulutusta ja varsin pitkä työkokemus ohjelmoinnin opettamisesta alakoulussa, joten tutkijan subjektiivisuus aineistoa analysoitaessa oli otettava huomioon. Haastatteluiden sisällöt on käsiteltävä mahdollisimman autenttisinä ja tutkijan oma tausta ei voi johdatella tulkintaa ja analyysiä.

Olemme toimineet koko tutkimuksen ajan avoimesti ja kuvanneet aineistosta kaikki vaiheet monipuolisesti. Ääninauhurin tiedostot ylikirjoitettiin ja litteroinnit poistettiin. Muut kirjalliset muistiinpanot ja haastateltavien kanssa käytyt sähköpostit hävitettiin yliopiston ohjeiden mukaisesti. Tutkittavilta saatuja tietoja käsiteltiin luottamuksella ja kunnioituksella koko tutkimuksen läpi.

## 7 TULOKSET

### 7.1 Tuloksien taustaa

Tutkimuksemme tehtävä oli selvittää, miten ja kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4 ja miten toiminnallisuus on otettu huomioon oppimateriaalien tehtävissä. Haastatteluissa selvitimme kahden luokanopettajan käsityksiä ja kokemuksia ohjelmoinnin opettamisesta alakoulussa.

Tutkimuksen kvantitatiiviset tulokset koostuvat numeerisista arvoista, jotka on laskettu kahden eri kirjakustantajan (Sanoma Pro Oy ja Otava Oy) yhteensä neljästä matematiikan oppimateriaalisarjasta (Milli, Kymppi, Tuhattaituri ja Oivaltaja) vuosiluokille 1–4. Tehtävien esiintymismääriin laskettiin varsinaiset ohjelmointitehtävät ja algoritmista ajattelua kehittävät ja tukevat tehtävät sekä toiminnallisuutta sisältävät tehtävät ohjelmoinnin yhteydessä.

Laadullisen ja määrällisen aineiston tulosten tarkastelussa tutkimuksen kvantitatiivisesta osuudesta saatuja lukumääriä on peilattu kvalitatiivisen osuuden haastattelujen sisältöihin ja niistä on pyritty löytämään yhtäläisyyksiä ja samankaltaisuutta. Oppimateriaalien tehtävät ja haastateltavien kommentit vuoropuhelevat keskenään. Seuraavaksi esittelemme tutkimustulokset tutkimuskysymysten mukaisesti.

### 7.2 Miten ja kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4?

Varsinaisten ohjelmointitehtävien määrä oppilaiden oppikirjoissa oli 326 kappaletta ja algoritmista ajattelua kehittäviä ja tukevia tehtävien määrä oli 3472 kappaletta. Yhteensä oppikirjoissa oli tehtäviä 13104 kappaletta. Varsinaisia ohjelmointitehtäviä ja algoritmista ajattelua tukevia tehtäviä verratessa kokonaismäärään saadaan niiden suhteelliset osuudet seuraavasti: varsinaisia ohjelmointitehtäviä 2,5 % ja algoritmista ajattelua kehittäviä ja tukevia tehtäviä

26,5 %. Laskennan kaikkien oppimateriaalien tulokset koottuna on luettavissa liitteestä 2. Oppikirjojen kappale- ja jaksokohtaiset tehtävämäärät ovat luettavissa liitteestä 3, joissa laskennat on eritelty kirjasarjoittain ja vuosiluokittain. Oppilaan oppikirjatehtävien lukumäärällisistä arvoista voidaan päätellä, että varsinaisten ohjelmointitehtävien suhteellinen lukumäärä (2,5 %) kokonaismäärästä on varsin pieni. Huomioitavaa on kuitenkin se, että ohjelmointia tukevia algoritmista ajattelua kehittäviä ja tukevia tehtäviä on oppimateriaaleissa melko runsaasti, yli neljäsosa. Vaikka varsinaisia ohjelmointitehtäviä on niukasti, on ajattelun kehittämistä tukevat tehtävät tärkeässä roolissa ohjelmoinnin opetuksessa, kuten haastattelemamme luokanopettajatkin toteavat:

Ohjelmointi, jos miettii ohjelmoinnillista ajattelua, niin se on musta se perusta, että miten se ajattelu siellä tapahtuu eli syy-seuraus-juttuja eli kausalityttöjuttuja. Se ajattelu ja sen kehittäminen on tässä vaiheessa riittänyt hyvin. (H1)

Jotenkin mä oon yrittänyt terottaa sitä oppilaille, että se tekee just niinku sä pyydät, eli se ei lue sun ajatuksia, eikä tee mitään ylimääräistä, että tavallaan niinku semmoinen ajatteluntaito tulee kyllä hyvin vahvasti. (H2)

Myös Liukas ja Mykkänen (2014) korostavat, että ohjelmoinnin opetus peruskoulutasolla on algoritmisen ajattelutaidon kehittämistä.

Tyypillinen varsinainen ohjelmointitehtävä oppilaan oppikirjassa harjoituttaa oppilasta liikkumaan ohjeiden mukaan kynä-paperi-tehtävässä. Esimerkiksi Milli 2B -oppikirjassa oppilaan täytyy kulkea koodin mukainen reitti (kuva 17).

## Kuva 17

Koodin mukaan toimiminen (Milli 2B, 200)

**1** Hahmot liikkuvat koodin mukaan. Piirrä reitit ja poimi kirjaimet.

**HILLA** **MUURAIN**  
**SUO** **LAKKA**

Toinen tyyppinen ohjelmointitehtävätyyppi oppikirjoissa harjoituttaa oppilasta purkamaan salakieltä (kuva 18). Tehtävässä oppilaan tulee selvittää viesti poimien taulukosta oikeat kirjaimet perinteisen laivanupotus pelin tyyliin.

## Kuva 18

Viestin selvittäminen (Kymppi 1S, 186)

**4.** Sel-vi-tä vies-ti.

▲	T	S	O	L	U
■	P	E	Ä	K	A
●	V	I	N	R	V
	1	2	3	4	5

1 ● 2 ■ 3 ● 4 ▲ 5 ■      1 ▲ 3 ▲ 2 ● 1 ● 3 ▲ 3 ▲  
V E N L A      T O I V O O

1 ■ 5 ▲ 4 ■ 2 ● 4 ▲ 1 ▲ 5 ■      1 ■ 5 ■ 4 ▲ 5 ■  
P U K I L T A      P A L A -

1 ■ 2 ■ 4 ▲ 2 ● 3 ■  
P E L I Ä.

Oppimateriaalien opettajan oppaiden tyypillisin ohjelmointitehtävä on ihmisrobotitehtävät (kuvat 19 ja 20). Tehtävissä ohjataan toisia oppilaita tai opettajaa antamalla selkeitä ja yksiselitteisiä käskyjä eli komentoja.

### Kuva 19

*Ihmisrobotitehtävä lattiaruudukossa (Kymppi 4K, opettajan opas, 123)*

<p><b>Lattiaruudut</b></p> <ul style="list-style-type: none"> <li>• Levitetään lattialle toisiinsa kiinni noin 30 cm x 30 cm ruutuja, joissa on lukuja, kirjaimia, kuvioita jne.</li> <li>• Oppilas seisoo ruudussa ja toinen oppilas antaa ohjeita käännoksistä ja liikkumisesta. Siirtyminen seuraavaan ruutuun on yksi askel.</li> </ul> <p>Esim.</p> <ul style="list-style-type: none"> <li>– Käänny vasemmalle.</li> <li>– Liiku 3 askelta eteenpäin.</li> <li>– Käänny vasemmalle.</li> <li>– Liiku yksi askel eteenpäin.</li> </ul> <p>Missä ruudussa seisot?</p>
--

### Kuva 20

*Opettajan ohjaaminen tarkoilla toimintaohjeilla (Milli 3A, opettajan opas, 127)*

<p><b>Opettajan ohjelmoiminen</b></p> <p>Opettaja asettaa näkyville terättömän lyijykynän ja teroittimen ja asettuu itse pienen välimatkan päähän niistä. Oppilaiden tehtävänä on antaa opettajalle tarkat toimintaohjeet kynän teroittamiseen:</p> <p>esim.</p> <ul style="list-style-type: none"> <li>• käänny oikealle</li> <li>• liiku viisi askelta/ruutua</li> <li>• ota lyijykynä oikealla kädellä</li> <li>• ota teroitin vasemmalla kädellä</li> </ul> <p>jne.</p> <p>Jos oppilaiden ohjeet ovat epätarkkoja, opettaja tekee tahallaan siten, että teroittaminen ei onnistu. Esim. kynä ja teroitin samassa kädessä, kynä ei ole reiässä perillä asti tai pyörittää kynää väärään suuntaan. Opettaja voi toimia heti jokaisen oppilaan ehdottaman käskyn mukaan tai kaikki käskyt voidaan ensin koota taululle, ja kun käsikirjoitus on oppilaiden mielestä valmis, opettaja suorittaa käskyt.</p>
---

Tarkkojen toimintaohjeiden antaminen ja niiden mukaan toimiminen kehittävät algoritmisen ajattelun taitoja. Fantin (1993, 4) mukaan algoritmin jokainen askel tulee olla täsmällisesti määritelty. Esimerkiksi yllä olevan (kuva 20) mukainen harjoitus kehittää oppilaita myös huomaamaan ohjelmassa olevan virheen, mikäli toimintaohje ei ole riittävän täsmällinen. Virheen eli bugin huomaaminen ja sen korjaaminen on myös haastateltavan (H1) mukaan hyvä keino opettaa ohjelmointia.

Kun lähetät sinne käskyjä, että toteuta tämä mun käsky, ja sit sä näet, että ei nyt meni pieleen ja sä keskeytät, että nyt on joku bugi tässä koodissa, niin nää on semmosia, mistä oppii ja kasvaa kaikessa tässä. (H1)

Opettajan oppaiden toimintavinkkien ja -tehtävien osiosta laskettiin tehtävien kokonaismääräksi 6209 kappaletta, joista 3689 kappaletta luokiteltiin algoritmista ajattelua kehittäviksi ja tukeviksi tehtäviksi. Kokonaismäärään suhteutettuna algoritmista ajattelua kehittävien ja tukevien tehtävien osuus oli yli puolet eli tarkalleen 59,4 %.

Ryhmittelimme luokanopettajien kokemuksia neljään pääkategoriaan, jotka kehittävät ja tukevat algoritmisen ajattelun kehittymistä. Pääkategoriat ovat ongelmanratkaisu ja looginen päättely, toimintaohjeen noudattaminen, suunta- ja sijaintikäsite sekä kielentämistehtävät. Nämä pääkategoriat ja niiden alakategoriat on esitelty taulukossa 4. Taulukosta löytyy myös esimerkkejä kategorioiden syntyyn vaikuttaneita haastateltavien lainauksia.

Seuraavaksi käymme läpi pääkategorioiden kannalta tärkeimmät tulokset, jotka nousivat luokanopettajien haastatteluissa esille. Samalla esittelemme myös kategorioihin liittyviä algoritmista ajattelua kehittäviä ja tukevia tehtäviä tutkimistamme oppimateriaaleista ja peilaamme tuloksia tieteellisiin tutkimuksiin kyseisestä aiheesta.

#### *Ongelmanratkaisu- ja looginen päättelykyky*

Ongelmanratkaisua ja loogista päättelykykyä vaativia ja kehittäviä tehtäviä löytyy oppimateriaaleista runsaasti. Wingin (2006, 33) mukaan algoritmisen

ajattelu sisältää ongelmanratkaisua. Liukkaan ja Mykkäsen (2014, 57) mukaan ohjelmoinnin avulla oppii yleisesti hyödyllisiä kognitiivisia taitoja, kuten loogista ja luovaa ajattelua sekä ongelman hahmottamis- ja ratkaisukykyä. Kääntäen voidaan ajatella, että juuri ongelmanratkaisua ja loogista päättelyä kehittävät tehtävät tukevat ohjelmointia ja algoritmista ajattelutaitoa. Perusopetuksen opetussuunnitelmakin korostaa matematiikan oppiaineen tavoitteissa ongelmanratkaisutaitoa (Opetushallitus, 2014, 128–129, 235–236).

Luokanopettajien kokemuksen mukaan ongelmanratkaisutaidot ovat yksi tärkeimmistä, kivoimmista ja motivoivimmista tehtävätyypeistä. Ongelmanratkaisuksi katsotaan muun muassa tehtävät, jotka sisältävät haasteita, avoimuutta ja useita erilaisia ratkaisuja.

Ongelmanratkaisutaidot on siis mun mielestä kaikista kivoimpia. (H1)

Ajatteluntaito tulee kyllä hyvin vahvasti ja se ongelmaratkaisu, että miten meidän pitää antaa ohjeita, että me saadaan se haluttu asia ilmi välineillä. (H2)

Tehtävät, jossa on niinku avoimuutta eli olisi joku haaste tai joku sellainen. Se voi olla joku opettajalta annettu tai sitten voi se ongelma olla tietysti lähtöisin siltä oppilaalta itseltään ja sitten ratkaisuja on niinku monenlaisia vaihtoehtoja, miten sen voi suorittaa. (H2)

Oppimateriaaleissa ongelmanratkaisua ja loogista päättelyä vaativat tehtävät voidaan nähdä esimerkiksi erilaisiksi tuntemattoman ratkaisemistehtäviksi ja hahmottamistehtäviksi. Oppimateriaaleissa ohjeistetaan ratkaisemaan ongelmanratkaisuja esimerkiksi piirtämällä, listaamalla järjestelmällisesti, kokeilemalla ja tarkistamalla sekä keksimällä sääntö (Milli 3A, opettajan opas, 17). Esimerkkejä ongelmanratkaisutehtävistä ja loogisen päättelykyvyn harjoittelemisesta on esitelty alla (kuvat 21, 22, 23, 24). Tehtävissä oppilaan tulee käyttää loogista päättelykykyään ratkaistakseen tehtävän tai harjoituksen.



### Kuva 21

Loogista päättelyä kehittävä tehtävä (Milli 2A, 25)

**8** Ke-nen pai-ta? Päät-te-le.



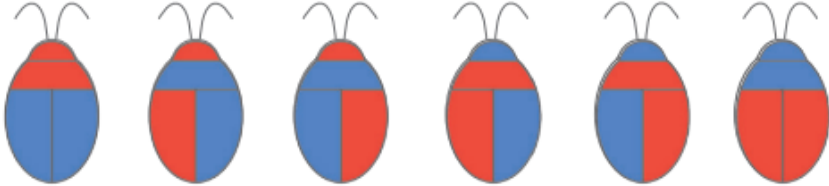
Jalo      Milli      Eena      Olli

- Ee-nan pai-dan lu-vus-sa ei ole kym-me-ni-ä.
- Ol-lin ja Ja-lon pai-to-jen lu-vut o-vat suu-rem-pi-a kuin Mil-lin pai-dan lu-ku.
- Ja-lon pai-dan lu-vus-sa kym-me-ni-ä ja yk-kö-si-ä on yh-tä mon-ta.
- Ol-lin pai-dan lu-ku on pa-ri-ton.

### Kuva 22

Loogista päättelyä kehittävä tehtävä (Milli 2A, 33)

**6** Vä-ri-tä jo-kai-nen ö-tök-kä e-ri ta-val-la. Vä-ri-tä jo-kai-ses-ta 2 o-saa ja 2 o-saa .



### Kuva 23

Päättelytehtävä (Milli 4B, 153)

**8** Täydennä.

$\text{Anchor} + \text{Anchor} + \text{Anchor} = 1,8$	$\text{Anchor} + \text{Lifebuoy} + \text{Lifebuoy} = 1,6$
$\text{Anchor} + \text{Anchor} = 1,2$	$\text{Anchor} + \text{Anchor} + \text{Lifebuoy} = 1,7$
$\text{Lifebuoy} + \text{Lifebuoy} + \text{Lifebuoy} = 1,5$	$5,0 - \text{Anchor} + \text{Lifebuoy} = 4,9$
$\text{Lifebuoy} + \text{Lifebuoy} + \text{Lifebuoy} + \text{Lifebuoy} = 2,0$	$5,0 - \text{Lifebuoy} + \text{Anchor} + \text{Anchor} = 5,7$

## Kuva 24

Ongelmanratkaisutehtävä (Milli 3A, opettajan opas, 51)



**Ongelmanratkaisua**

Harjoitellaan ongelmanratkaisua piirtämällä tehtävästä yksinkertainen kuva tai kaavio, jonka avulla tehtävä on helpompi ratkaista.

Matikkatalon pihalla on erivärisiä autoja. Punainen auto on talon vasemmalla puolella. Sininen auto on punaisen auton oikealla puolella ja valkoisen auton vasemmalla puolella. Minkä värinen auto on lähimpänä Matikkataloa?

- Kerro omin sanoin, mitä tehtävässä pitää tehdä (selvittää, mikä autoista on lähinnä taloa)
- Minkä värisiä autot ovat? Miten värit voi korvata kirjaimin? (P, S ja V)
- Piirretään talon kaavakuva.
- Mistä aloitetaan? (merkitään P talon vasemmalle puolelle, mutta jätetään riittävästi tilaa muille värimerkinnöille eli edetään tehtävän tietojen mukaan)
- Mitä tehdään seuraavaksi?

Annetaan oppilaiden ratkaista piirtämällä ja merkitsemällä kuvaan autojen värien mukaiset alkukirjaimet.

P  P S V 

Valkoinen auto on lähimpänä Matikkataloa.

Ongelmanratkaisutehtävässä (kuva 24) oppilas harjoituttaa loogisen päättelyn lisäksi muitakin algoritmista ajattelua kehittäviä taitoja, esimerkiksi kielentämistä ja toimintaohjeiden mukaan toimimista.

### *Toimintaohjeen noudattaminen*

Toimintaohjeiden mukaan toimiminen on oleellinen osa algoritmeja ja algoritmista ajattelua. Vaiheittaisten toimintaohjeiden laatiminen, ymmärtäminen ja niiden mukaan toimiminen on Fantin (1993, 4) algoritmin määritelmän mukaista, jossa algoritmin täytyy olla askel askeleelta etenevä sekvenssi toimintoja. Larssonin kumppaneineen (2021, 17) mukaan algoritminen

ajattelu on juuri ongelman ratkaisemiseen tai tehtävän suorittamiseen tarvittavien toimenpiteiden ja niiden suoritusjärjestyksen määrittämistä. Toimintaohjeiden vaiheittaisella noudattamisella ja järjestyksellä on oleellinen rooli ohjelmoinnissa. Myös perusopetuksen opetussuunnitelman matematiikan oppiaineen tavoitteissa harjaannutetaan ja innostetaan oppilaita laatimaan toimintaohjeita sekä toimimaan ohjeiden mukaan (Opetushallitus, 2014, 128–129, 235–236).

Toimintaohjeen noudattamista vaativia ja kehittäviä tehtäviä on oppimateriaaleissa runsaasti, ja niihin lukeutuvat usein erilaiset pelit ja leikit ja tehtävät, jotka vaativat ratkaistakseen vaiheittaisten toimintaohjeiden seuraamista. Lisäksi moni arkinen tapahtumasarja koulumaailmassa edellyttää oppilailta toimintaohjeiden noudattamista, esimerkiksi liikuntatuntien peliohjeistukset ja käsitöiden valmistusohjeistukset. Matematiikassa erilaiset laskualgoritmit, kuten allekkain vähennyslasku, tulevat jo alaluokilla oppilaille tutuiksi.

Luokanopettajat tuovat esille eräänlaisia koodin tarkastelutehtäviä, joissa oppilaiden tehtävänä on selvittää, missä toimintaohjeiden noudattamisessa on tehty virhe ja toisaalta tehtäviä, jotka ovat suoritettavissa yhdellä ainoalla napin painalluksella, kunhan koodi on saatu ensin tuotettua.

Voisi olla ihan hyvä, että sulla on joku "korjaa koodi" tai joku semmoinen, että pitää tarkasti lukea jotain, että miksi tämä ei nyt toimi. (H2)

Kun saadaan yhdellä käskyllä menemään pitkä rimpsu asioita läpi, yhdellä ainoalla napin painalluksella. (H1)

Toimintaohjeiden noudattamista vaativia tehtäviä ja harjoituksia on esitelty alla olevissa kuvissa (kuvat 25, 26 ja 27). Esimerkiksi tehtävässä Oppilaiden ohjelmoiminen (kuva 25) ohjeistetaan oppilaita kouluarjen työskentelyyn. Ohjelmointitehtävä (kuva 26) on oppilaan oppikirjoissa yksi tyypillisimmistä tehtävätyypeistä. Tehtävässä oppilaan tulee toimia vaiheittaisen toimintaohjeen (koodin) mukaan ratkaistakseen tehtävän ja toisaalta selvittää reitistä koodin arvot ja suunnat eli parametrit.

## Kuva 25

*Toimintaohjeiden noudattaminen toistoin ja ehdoin (Milli 4B, opettajan opas, 137)*

**Oppilaiden ohjelmoiminen**

Ennen välitunnin alkamista tai ruokailuun lähtemistä opettaja heijastaa tai kirjoittaa taululle oppilaille skriptin, jonka mukaan oppilaiden on toimittava. Käskyt voivat olla lohkoissa, silmukan sisällä tai ehtolauseessa.

**Esimerkki**

```

Toista kunnes olet jonossa
Tee matematiikan tehtäviä
Jos Kuulet nimesi
Tee Pese kätesi ja mene ruokajonoon
  
```

## Kuva 26

*Koodin mukaan piirtäminen ja koodin täydentämistä (Kymppi 3K, 198)*

**4. Piirrä punaisen leppiksen reitti koodin mukaan.**

**5. Tutki sinisen leppiksen reittiä. Täydennä koodiin oikeat luvut ja suunnat.**

**4. Koodin osat:**

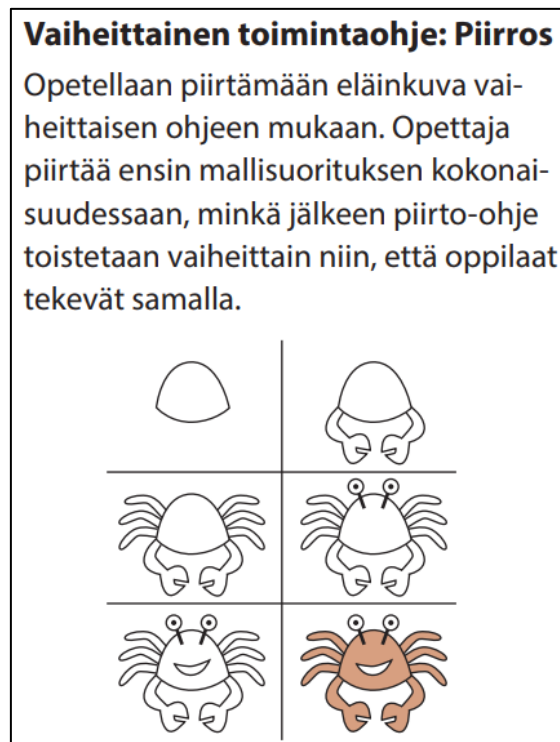
- liiku eteenpäin 2
- käänny vasemmalle
- liiku eteenpäin 3
- käänny oikealle
- liiku eteenpäin 3
- käänny oikealle
- liiku eteenpäin 4
- käänny oikealle
- liiku eteenpäin 1

**5. Täydennettävä koodi:**

- liiku eteenpäin
- käänny
- liiku eteenpäin
- käänny
- liiku eteenpäin
- käänny
- liiku eteenpäin
- käänny
- liiku eteenpäin

**Kuva 27**

*Toimintaohjeen mukaan piirtäminen (Milli 2B, 131)*



Toimintaohjeiden mukaista toimintaa voidaan harjoitella esimerkiksi piirtämällä kuten yllä olevassa tehtävässä (kuva 27). Samoin toimintaohjeiden noudattamisen harjoittelua esiintyy oppimateriaalien erilaisissa leikeissä ja peleissä (kuva 28 ja 29). Usein peleissä esiintyy monia algoritmista ajattelua kehittäviä ja tukevia toimintoja, kuten juuri toimintaohjeiden noudattamista, ongelmanratkaisua, loogista päättelyä ja suunta- ja sijaintikäsitteiden harjoittelua. Matematiikan oppiaineen oppimisympäristöissä ja työtavoissa korostuvat muun muassa pedagogisesti ohjatut leikit ja pelit (Opetushallitus, 2014, 130, 236). Liukkaan ja Mykkäsen (2014, 48, 86) mukaan ohjelmoinnillisia taitoja harjoitellaan kahdella ensimmäisellä luokalla leikkien kautta, joissa opetellaan antamaan yksikäsitteisiä komentoja. Leikeissä pyritään suorittamaan tietyt toimet tietyssä järjestyksessä tehtävän ratkaisemiseksi.

## Kuva 28

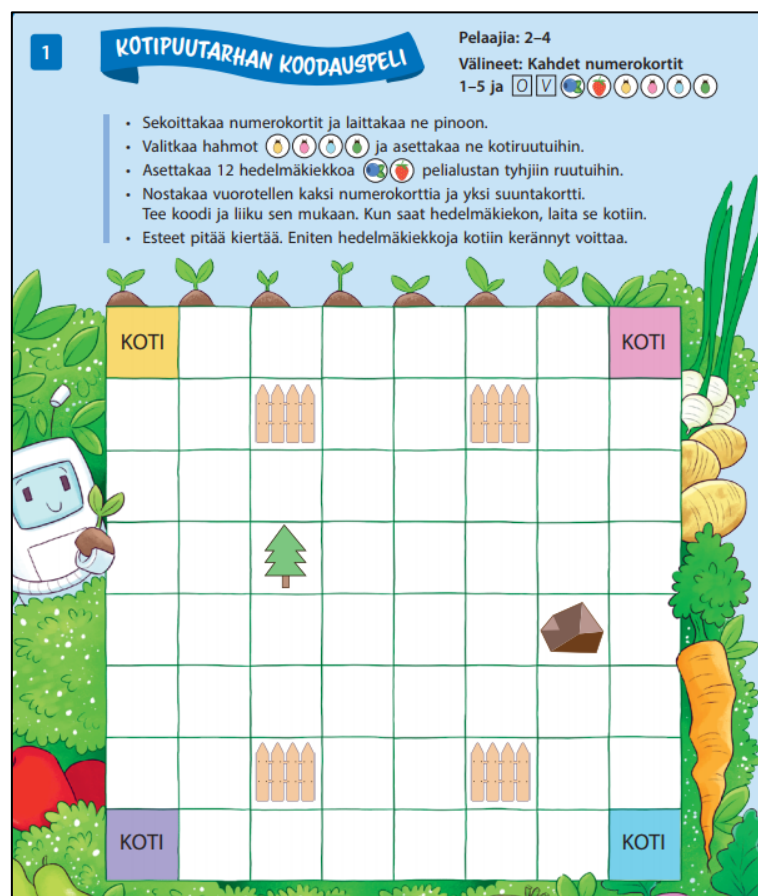
Arvaa aikani -leikki (Kymppi 4K, opettajan opas, 101)

### Arvaa aikani -leikki

- Opettaja sanoo ajattelevansa aikaa pistemerkintänä. Aika on tasan viisi minuutteja.
- Oppilaat yrittävät saada ajan selville kysymyksillä, joihin opettaja vastaa vain kyllä tai ei.
- Sopivia kysymyksiä ovat esim. Onko se ennen puoltapäivää? Onko tunteja enemmän kuin 5? Onko minuutteja enemmän kuin 30? jne.
- Oppilas, joka arvaa oikean ajan, saa olla seuraava ajan valitsija.
- Leikki sopii pariharjoitukseksi, jolloin voidaan kisata siitä, kumpi saa vähemmällä kysymyksillä selvitettyä toisen ajan.

## Kuva 29

Koodauspeli (Milli 2B, 204)



### Suunta- ja sijaintikäsite

Oppimateriaalit sisältävät suunta- ja sijaintikäsitettä harjoituttavia tehtäviä. Ohjelmointiluvuista löytyy paljon sitä kehittäviä tehtäviä. Tehtävissä annetaan tyypillisesti ohjeita, joissa oppilaan pitää kulkea reitti läpi annettujen ohjeiden mukaisesti, kuten esimerkiksi alla olevassa tehtävässä (kuva 30).

### Kuva 30

Reitin piirtäminen ohjeen eli koodin mukaan (Kymppi 4K, 202)

**3. Piirrä leppisten reitit koodin mukaan. Väritä kukka, johon leppis päätyy.**

The image shows two programming tasks for a robot on a 5x5 grid. Each task has a list of commands and a grid with obstacles (rocks) and a target (flower).

**Task 1 (Left):**

- liiku eteenpäin 2
- jos edessä kivi
- käänny oikealle
- liiku eteenpäin 2
- muuten
- liiku eteenpäin 3

**Task 2 (Right):**

- liiku eteenpäin 2
- jos edessä kivi
- käänny vasemmalle
- liiku eteenpäin 1
- muuten
- liiku eteenpäin 2

Tehtävissä avaruudellinen hahmottaminen on tärkeää, jotta oppilas osaa hahmottaa sijainnin ja suunnan oikein oikeassa paikassa. Opettajan oppaassa (esimerkiksi Milli 3A opettajan opas) tähän ohjeistetaan näin: ”Suunnan hahmottamisessa auttaa oppikirjan kääntäminen niin, että oppilas katsoo reittiä samasta suunnasta kuin hahmo”

Vastaavanlaisia kokemuksia suunnan hahmottamisesta löytyi luokanopettajalta (H1).

Hyvä esimerkki, kun kahden tytön ryhmä oli tosi inessä ja ne mietti, että mikä on tässä tämä kuvio, mäki menin viereen katsoon ja näin, että ne oli tehnyt tosi paljon oikein ja sitten älysin sen että se ei menny niin ku piti, niin se oli tämä suunta ja sijainti, kun sen (robotin) nokka oli jo toisessa suunnassa. Kun se aukesi niin, olisittepa nähneet, keuhkot täyteen ja kiljuntaa, että nyt he tajuaa tämän. Kun on hyvin pitkä rimpsu tehty siihen

ja sitä yhdessä pohdittiin. Se oli hieno kokemus ja minusta tämä suunta ja sijainti on nimenomaan se että silloin, kun sä näet, että robotti menee siellä ja sä näet, mitä siellä tapahtuu, sä muistat sun kirjoittaman koodin ja voit seurata sitä robottia, niin nämä on niitä elämyksiä joita ei varmasti voi oppikirja tarjota. (H1)

Luokanopettajan (H1) kokemuksesta huomaa, kuinka suunnan ja sijainnin hahmottaminen voi olla vaikeaa ja kuinka innostuneita oppilaat olivat onnistumisesta. Samoin luokanopettajan kokemuksesta huomaa, että kuinka tärkeässä roolissa robotit voivat olla havainnollistamaan ohjelmointia verrattuna oppikirjojen tehtäviin. Yksi Futschekin (2006, 160) määrittelemistä algoritmisen ajattelun kyvyistä, on kyky ymmärtää ongelmaa erilaisissa tilanteissa. Tässä tapauksessa ongelma on hahmon tai robotin sijainti ja kulkusuunta suhteessa annettuihin komentoihin. Myös perusopetuksen opetussuunnitelman matematiikan oppiaineen sisällöissä mainitaan, että oppilas harjoittelee suunta- ja sijaintikäsitteiden käyttöä (Opetushallitus, 2014, 128–129).

#### *Kielentämistehtävät*

Kielentäessään oppilas tuo esiin matemaattista ajatteluaan puhekielen, kirjoitetun kielen, matematiikan symbolikielen keinoin sekä graafisin esityksin. Kielentäminen auttaa oppilasta jäsentämään ajatteluaan ja välittämään sitä myös muille. Oppilaat voivat usein kokea kielentämisen vaikeaksi, mutta sitäkin voidaan harjoituttaa. Joutsenlahti ja Tossavainen (2018, 414, 422) tuovat artikkelissaan esille, että kielentämistä tulisi harjoitella systemaattisesti, jotta sitä voidaan onnistuneesti käyttää oppimisessa. Matemaattista kielentämistä voidaan toteuttaa kuviokielen, luonnollisen kielen, taktiilisen toiminnan kielen sekä matematiikan symbolikielen keinoin. Samoin voidaan ajatella algoritmisen ajattelun kielentämisessä. Kielentämällä algoritmista ajatteluaan oppilas voi tuoda esille ymmärrystään ohjelmoinnista, esimerkiksi siitä mikä on ulkoa opettelu tulosta ja mikä todellisen ymmärtämisen tulosta. Perusopetuksen opetussuunnitelman matematiikan oppiaineen tavoitteissakin kannustetaan oppilasta esittämään päättelyään ja ratkaisujaan muille konkreettisilla välineillä, piirroksilla, suullisesti ja kirjallisesti (Opetushallitus, 2014, 128–129, 235–236).



Luokanopettajan (H1) mukaan ohjelmointiharjoitukset, joissa on mukana jokin tarina, kiehtoo oppilaita ja siten motivoi harjoitukseen.

Ennen kaikkea, että näistä saa toiminnallisia kokonaisuuksia, joihin liittyy joku tarina, joka sitä kautta kiehtoo. (H1)

Kielentäminen on myös hyvä keino luokanopettajalle havaita algoritmisen ajattelun taitojen kehittymistä. Luokanopettajat kokivat myös kielentämisen konkreettisina välineinä, esimerkiksi robotein, hyödylliseksi.

Toi tommonen kielentäminen ja just vuorovaikutuksessa siinä niin sä havaitset, että mitenkä se on tajunnut sen. (H2)

Tuo robotiikka on keino visualisoida sitä osaamisen tasoa, niin kyllä siinä opettajalla on mahdollisuus seurata vierestä ja tehdä havaintoja siitä että kenellä tämä kehittyy ja kenellä ei, ja kenellä on niitä haasteita, tavallaan robotin avulla sä pystyt sen sitten niin kuin visuaalisesti todentamaan toiselle että näin mä sen ajattelin (H1)

Opettaja voi saada esimerkiksi kuvan siitä, onko oppilas ymmärtänyt algoritmisen ajattelun ohjelmointitehtävän taustalla, vaikei teknisesti ole sitä saanut suoritettua. Oppimateriaaleissa kielentämistä ohjeistetaan esimerkiksi Milli-sarjassa näin:

Tavoitteena on oppiminen ymmärryksen kautta. Ymmärrystä pyritään tuomaan tietoisemmalle tasolle kielentämisen avulla. Selitä parille -tehtävät ohjaavat oppilasta tiedostamaan oppimiaan lasku- ja ajattelustrategioita ja kertomaan niistä parille.

Alla oleva tehtävä (kuva 31) on esimerkki kielentämistehtävästä oppilaan oppikirjassa.

**Kuva 31**

*Kielentäminen. Selitä parille. (Milli 2B, 96)*

**7** Tutki kuvia ja laskuja. Korjaa väärin tehdyt laskut oikeaksi.

$4 \cdot 5 = 20$   
 $3 \cdot 5 + 4 = 19$

$4 \cdot 3 = 12$   
 $4 \cdot 3 + 2 = 14$

Miksi lasku oli väärin?  
Selitä parille.

Oppimateriaaleissa kielentämistehtäviä löytyy jonkin verran oppilaan oppikirjan ja enemmän opettajan oppaan puolelta, joissa oppilas esimerkiksi harjoittelee muodostamaan tarinasta lausekkeen tai lausekkeesta tarinan sekä harjoituksia, joissa oppilas verbaalisesti selittää muille, kuinka tehtävä (kuva 32) tai laskualgoritmi (kuva 33) rakentuu.

**Kuva 32**

*Kielentämistehtävä (Milli 3A, opettajan opas, 27)*

**Pariharjoitus**

Välineet: kymmenjärjestelmäalusta ja numerokortit

Toinen parista sanoo jonkin kolmi-numeroisen luvun ja toinen muodostaa sen numerokorteilla omalle alustalleen. Vaihdetaan rooleja.

### Kuva 33

*Laskualgoritmin kielentäminen (Kymppi 3K, opettajan opas, 97)*

**Oppilas kertoo**

- Merkitään taululle allekkain lasku, esim.  $7 \cdot 316$ . Oppilas selostaa vaiheet. Opettaja voi toimia kirjurina, jolloin hän voi pyytää selvennystä, jos ohje ei ole toimiva.
- Korostetaan vaiheita. Ensin kerrotaan ykköset,  $7 \cdot 6 = 42$ . Neljä kymmentä merkitään muistiin kymmenen päälle. Kaksi merkitään alas ykkösten paikalle.
- Sitten kerrotaan kymmenet ja lisätään muistinumero,  $7 \cdot 1 + 4 = 11$ . Yliviivataan käytetty muistinumero. Merkitään 1 muistiin satojen päälle ja 1 alas kymmeniin.
- Lopuksi kerrotaan sadat ja lisätään muistinumero  $7 \cdot 3 + 1 = 22$ . Merkitään 22 tuhansien ja satojen paikalle.
- Sopivia laskuja ovat myös  $5 \cdot 424$  ja  $6 \cdot 523$

### Kuva 34

*Tarinasta lauseke (Oivaltaja 1b, opettajan opas, 98)*

**Laskujärjestys oppilailla**

Opettaja kertoo matikkatarinan. Oppilaat esittävät tarinan vaiheittain. Merkitään lopuksi tarinasta lasku näkyviin.

Esimerkkitarinat:

- Puistossa on 5 lasta. Puistoon tulee 7 lasta lisää. 4 lasta lähtee kotiin syömään. Kuinka monta lasta puistoon jää? ( $5 + 7 - 4 = 8$ )
- Koirapuistossa on 10 koiraa. Niistä 5 lähtee pois. Puistoon tulee 8 koiraa lisää. Kuinka monta koiraa puistossa sitten on? ( $10 - 5 + 8 = 13$ )
- Oksalla on 11 lintua. Oksalle lentää 3 lintua lisää. 7 lintua lentää pois. Kuinka monta lintua oksalle jää? ( $11 + 3 - 7 = 7$ )

Tutkitaan, missä järjestyksessä laskut laskuissa laskettiin.

Todetaan lopuksi laskujärjestyssääntö: Yhteenlaskut ja vähennyslaskut laskeaan järjestyksessä vasemmalta oikealle.

Yllä olevassa tehtävässä (kuva 34) harjoitellaan luomaan lauseke tarinasta. Samalla harjoitellaan algoritmista ajattelua, kun laskujen suoritusjärjestyksellä on merkitystä.

### 7.3 Miten toiminnallisuus on otettu huomioon ohjelmoinnin opettamisessa?

Oppilaan oppikirjojen ohjelmointitehtävistä ja algoritmista ajattelua tukevista ja kehittävästä tehtävistä 409 kappaletta oli toiminnallisia tehtäviä eli 10,8 % kokonaismäärästä (3798 kappaletta). Vastaavasti opettajan oppaiden tehtävistä lähes kaikki olivat toiminnallisia, johon viittaavat useiden osioiden nimetkin kuten "Opettavat toimintavinkit" ja "Harjoittava toimintavinkki" (Oivaltaja), "Toiminta" (Tuhattaituri), "Harjoittelu" (Milli) ja "Toiminnallisia harjoituksia". Ainoastaan erilliset pulmatehtävät olivat vailla toiminnallisuutta ja ne keskittyivät ongelmanratkaisutaitojen ja loogisen päättelykyvyn kehittämiseen.

Perusopetuksen opetussuunnitelmassa mainitaan matematiikan oppiaineen tehtävässä, että "Konkretia ja toiminnallisuus ovat keskeinen osa matematiikan opetusta ja opiskelua." (Opetushallitus, 2014, 128, 234.) Haastatteluissa toiminnallisuus ohjelmoinnin opetuksessa tulee myös hyvin esiin luokanopettajien kokemuksissa.

Mä ajattelen, että se konkretia, niinku se toiminnallisuus on se tärkeä juttu, se ei oo tärkein, mutta se on tärkeä. Toiminnallisuus on muun muassa niinku ihan suora semmoinen havainnollinen osoitus siitä konkretiasta, että sulla on jokin jolla sä näet, että mitä tapahtuu käytännössä, kun annetaan käsky, niin siitä seuraa jotakin. Ja että sä oot toiminnassa mukana. (H1)

Toiminnallisuus voi luokanopettajan mukaan olla välillä hankala määritellä. Koskenkarin (2012) mukaan toiminnallisuudella tarkoitetaan toiminnan aktiivisuutta, johon yhdistyy ajatuksellisuus, osallistuminen, kokemuksellisuus ja vuorovaikutus. Norrena (2016, 14) kuvailee toiminnallista oppimista oppilaan aktiivisena toimimisena ja ajattelemisena oppimisprosessin aikana. Niin ikään perusopetuksen opetussuunnitelmassa korostetaan oppilaan aktiivisuutta ja osallisuutta oppimisprosessissa. Siinä oppilas on aktiivinen toimija ja toimii vuorovaikutuksessa yhdessä toisten kanssa. (Opetushallitus, 2014, 17.) Luokanopettajien kokemuksissa esiintyy paljon samoja määritteitä.

En mä tiedä, miten mä osaan sen tarkemmin määritellä, mutta tämmöisen niinku tekemisen kautta, sä oot itse siinä aktiivinen oppija. (H2)

Mä en itse näe, että se oppikirjan täyttäminen, vaikka kynällä piirretään viiva ja näin, se sisältää jonkinlaista toimintaa, mutta se että siihen harvoin liittyy semmoista yhdessä toimimista, joka on minulle aina tärkein. (H1)

Luokanopettajat korostavat toiminnallisuudessa yhdessä tekemistä, aktiivisuutta, tekemällä oppimista ja konkreettista toimintaa. Vastakohtana toiminnallisuudelle he pitävät oppikirjantehtäviä ja niiden täyttämistä.

Ja että käytännössä käsien kautta tapahtuis, että omilla käsillä tehdään yhdessä joku juttu ja siitä seuraa jotakin toimintaa tai robotin liikkeitä. Mä jotenkin koen, että toiminnallisuus on sitten oikeasti toimintaa, konkreettista toimintaa, joka tapahtuu. (H1)

Jotenkin mä käsitan tuon toiminnallisuuden, onko se sitten ohjelmointia tai matematiikan muuta opetusta, niin tekemällä opitaan eli päästään kokeilemaan erilaisia ratkaisuvaihtoehtoja ja etsimään niitä. Ja se kirjatehtävä on musta se vastakohta sille, että lykätään joku, että tee tämä ja siihen on ehkä yksi oikea ratkaisu. (H2)

Ilman tietokonetta toteutettavassa opetuksessa vältetään ruudun ääressä istumista ja itsenäistä työskentelyä, jolloin voidaan hyödyntää monia muita tapoja oppia yhdessä muiden kanssa, kuten toiminnallisuus (Caldwell & Smith, 2017, 4). Tyypillinen ilman tietoteknisiä koneita tehtävä toiminnallinen ohjelmointitehtävä on ihmisrobotin ohjaus. Ihmisrobotitehtävässä oppilas ohjaa opettajaa tai toista oppilasta toimimaan vaiheittaisten toimintaohjeiden mukaan. Oppimateriaalien opettajien oppaissa on ihmisrobotitehtäviä ohjelmointijaksojen yhteydessä. Tyypillinen ihmisrobotin ohjaamistehtävä on esitelty alla kuvissa 35 ja 36. Tehtävissä korostetaan ihmisrobotille annettavien komentojen tarkkuutta ja yksiselitteisyyttä. Yksi Knuthin (1997, 4–6) algoritmin viidestä ominaisuudesta on juuri definiittisyys. Jokaisen algoritmin askeleen tulee olla täsmällisesti ja yksiselitteisesti määritelty.

## Kuva 35

*Ihmisrobotin ohjaaminen (Milli 3B, opettajan opas, 137)*

<b>Oppilaan ohjelmoiminen</b>	
<p>Yksi oppilaista menee hetkeksi kuuloetäisyyden päähän, esimerkiksi luokan ulkopuolelle. Muut oppilaat sopivat yhdessä opettajan avustuksella tehtävästä, joka luokan ulkopuolelle menneen oppilaan tulee luokkaan palattuaan suorittaa. Ohjelma eli käskyt tehtävän suorittamiseen kirjoitetaan oppilaalle luokan taululle tai paperille niin, että ne ovat koko ajan oppilaan nähtävissä. Käskyjen tulee olla tarkkoja ja yksiselit-</p>	<p>teisiä niin, että ne tarkasti suorittamalla luokan asettama tavoite toteutuu.</p> <p>Oppilas aloittaa käskyjen suorittamisen sovitusta paikasta, esim. ovensuusta. Liikkumista voi ohjeistaa askelten määrällä tai hyödyntäen luokkatilan lattian kuviointia. Sopiva tavoite voi olla tiettyyn paikkaan pääseminen, yksinkertaisen toiminnon suorittaminen (esim. valon sammuttaminen luokasta) tai esineen löytäminen.</p>

## Kuva 36

*Toimintaohjeiden antaminen ihmisrobotille (Oivaltaja 3a, opettajan opas, 62)*

<b>PROJEKTITEHTÄVÄ</b>
<p><b>Ohjeet robotille</b></p> <p><b>Tarvikkeet:</b> kynä ja paperi</p> <ol style="list-style-type: none"> <li>1. Valitse parin kanssa arkinen toiminto, josta kirjoitatte ohjeet robotille.</li> <li>2. Kirjoittakaa ohjeet mahdollisimman tarkasti. Ohjeissa ei saa olla ylimääräisiä sanoja tai epätarkkoja ilmauksia. Käyttäkää ohjeissa käskymuotoa (liiku, käänny, sano...).</li> <li>3. Testatkaa ohjeita kanssa niin, että toinen toimii ohjeiden antajana ja toinen robottina.</li> </ol> <p>Esimerkki: Sukan pukeminen</p> <ul style="list-style-type: none"> <li>• Ota yksi sukka käteen.</li> <li>• Avaa sukan suuaukko.</li> <li>• Laita varpaat sukan sisään.</li> <li>• Vedä sukka kokonaan jalkaan.</li> </ul>

Yllä olevassa tehtävässä (kuva 36) korostetaan myös sitä, ettei ohjelmassa voi olla ylimääräisiä sanoja tai epätarkkoja ilmauksia, lisäksi kehoitetaan käyttämään käskymuotoa komennoissa. Myös luokanopettajalla (H1) oli kokemuksia ihmisrobotiharjoituksista ja sen yhteydessä käskyjen tarkkuudesta ja täsmällisyydestä.

Kuvitellaan robottia, että joku oppilaista tai sitten opettaja on ohjelmoitava ja oppilaat sitten ohjelmoi toisiaan, ja sitten siinä on tarkkaa sekä ohjelmoijan että ohjelmoitavan puolesta, että ihan tarkasti siinä noudattaa niitä, että ne käskyt on tarkkoja ja yksinkertaisia. (H1)

Ihmisrobotiikan hyödyntäminen ohjelmoinnin opetuksessa koettiin hyödylliseksi havainnollistamaan algoritmista ajattelua ja sitä, kuinka tärkeää on komentojen yksiselitteisyys ohjelman toimivuuden kannalta. Seuraavassa luvussa esittelemme lisää luokanopettajien kokemuksia ohjelmoitavan robotiikan avulla, jotka myös laskemme toiminnalliseksi oppimiseksi.

#### **7.4 Minkälaisia kokemuksia luokanopettajilla on ohjelmoinnin opettamisesta?**

Haasteltavilla luokanopettajilla oli molemmilla runsaasti kokemusta ohjelmoinnin opettamisesta alakoulussa. Ohjelmointiosaamisen he olivat saaneet erilaisista koulutuksista ja itsenäisesti perehtymällä. Luokanopettajien kokemuksista selviää heidän työtapansa ja minkälaisia resursseja ohjelmoinnin opettaminen luokanopettajalta vaatii heidän mielestään. Molemmat luokanopettajat opettivat ohjelmointia lähes täysin ilman matematiikan oppikirjoja. Oppikirjojen tehtäviä käytettiin ainoastaan jonkinlaisina lisätehtävinä. Muita ohjelmointiin liittyvää kirjallisuutta saatetaan käyttää tarjoamaan vinkkejä ja ideoita opetukseen.

Ehkä se olisi jotenkin täyteohjelmaa, mutta mielestäni se oppimisen kokemus, niin sitä ei voi saavuttaa niitä kautta. (H1)

Emme ole tehneet matematiikan kirjan takaa niitä ohjelmointitaitoja, oon sanonut oppilaille suoraan, että meidän ei tarvitsee näitä tehdä, vaan me tehdään ohjelmointia muuten. (H2)

Koululaisen peliohjelmointikirjaa ja Edukustannuksen tekemää Raapaisu-kirjaa, niin käytän ehkä semmoisena ideointina tai niinku, jos joku oppilas on jumissa, että mitä mä haluaisin tehdä, niin sitten voi katsoa vähän jotain vinkkejä. Linda Liukkaan kirjaa olen itse lukenut. (H2)

Luokanopettajan (H1) mielestä ohjelmoinnin opettaminen jää hyvin suppeaksi, mikäli sitä toteutetaan pelkästään oppikirjojen varassa. Opettamiseen on otettava mukaan toiminnallisuutta, esimerkiksi erilaisin välinein (robotiikka), havainnollistamaan oppimista.

Minusta kirja on kaksiulotteinen ja välillä jopa yksiulotteinen, jos ajattelee tämmöstä aihetta kuin ohjelmointi. Ei varmaan ole vastaava kokemus kuin se että sä otat iPadilla yhteyden johonki bluetooth-robottiin ja kun sä teet koodia sinne, niin se robotti seuraa ja toteuttaa ne käskyt. (H1)

Luokanopettaja (H1) opettaa ohjelmointia sisällyttäen sitä moneen hetkeen ja aiheeseen. Etenkin ongelmanratkaisutaitoihin sisällyttäen sitä voidaan kehittää useissa erilaisissa tilanteissa.

Sitä on mukana tasaisesti ja säännöllisesti kehitetään sitä hommaa. Se ei ole vain yksi jakso, joka toteutetaan tossa noin, vaan sitä tulee sujuvasti siinä melkein kaikessa toiminnassa. Sitä pystyy yhä enemmän ja enemmän sijoittamaan vaikka näihin ongelmanratkaisutehtäviin. (H1)

Luokanopettajien mukaan ohjelmoinnin opettaminen vaatii erilaisia resursseja. Opettaminen vaatii ensinnäkin opettajan omaa henkistä resurssia, jolla luokanopettaja tarkoittaa opettajan oman osaamisen ja tietotaidon kehittämistä.

Että sä olet itse sinut sen ohjelmointiasian kanssa, ehkä sun täytyy opettaa sitä ja sinä olet kerännyt itsellesi osaamisen resursseja siihen, että sä tiedät sen oman lähestymistavan siihen ohjelmointiaiheeseen. Käyt sieltä (koulutuksista) omaksumassa niitä, että miten sinä voisit välittää viestiä sinuna niille oppilaillesi, niin se on tärkein resurssi, että sä olet itse niinku sujut sen kanssa. (H1)

Eihän se opettajalta vaadi juuri mitään, muuta kuin sitä rohkeutta, uskallusta ja kokeilunhalua. Mäkin oon lähtenyt nollasta liikenteeseen, että ei mulle ole OKL:ssä opetettu ohjelmointia, ja että kyllä se oma aika on ehkä se, jos resursseja oikein mietitään, että mistä opettaja saa sen ajan, että hän voi itse tutustua asioihin. (H2)



Toiseksi nousee esille koulun aineellinen resurssi ja kuinka tärkeänä luokanopettaja sitä pitävät. Luokanopettajien mielestä kouluilta on hyvä löytyä edes jonkinlaisia tietoteknisiä valmiuksia havainnollistamaan ohjelmointia.

Voidaan ajatella tällöinen aineellinen resurssi, että kun oppikirjat laitetaan meille niin siinähan tulee sitten koodaamiseen ja ohjelmoinnin opetukseen resurssia, mutta kannattaa nyt miettiä sitten sen vaikuttavuutta ensinnäkin vaikka siitä omasta lähtökohdasta, elikkä arvostatko sinä itse, olisiko sinusta mukava oppia ohjelmointia piirtämällä viivoja kirjaan, vai olisiko sinusta hassumpaa ja mukavampaa, että olisi edes yksi robotti. (H1)

Jos meinaa jotain edes toiminnallista vähän enemmän tehdä kuin sitä robottileikkiä tai jotain muuta tällöistä ihmiseltä ihmiselle tapahtuvaa juttua, niin kyllä se vaatii ainakin ne tietotekniset laitteet eli sen pädin esimerkiksi millä voi vaikka jollakin Scratch Juniorilla tehdä. (H2)

Kolmanneksi esiin nousee taloudellinen resurssi, joihin voi löytyä apua myös ulkopuolisilta rahoittajilta. Yhtenä esimerkkinä luokanopettaja (H1) mainitsee erityisen välinelainaamon helpottamaan koulujen taloudellisia resursseja.

Taloudellinen resurssi ja sitä tulee tietysti, kun koulutuksia, tietysti on sitten paikalliset kouluttajat, joka vähentää resursseja jo ostettuina erilaisten hankkeitten myötä, mutta kyllä sitten jossain vaiheessa tulee jotain laitehankintoja. Firmoja ja muita, jotka on satsannut siihen, että on hoitanut valtavan varaston erilaisia robotiikkavälineitä ja koulut voi lainata niitä eli kouluun ei siis tarvitse ainakaan heti ostaa omia. (H1)

Luokanopettajien kokemuksista nousi esille ohjelmoinnin opettamisen työtapoja, joista yhteistoiminnallinen työtapo osoittautui selkeästi käytetyimmäksi verrattuna itsenäiseen työskentelyyn. Alla olevaan taulukkoon (taulukko 5) on koottu seikat, jotka ilmenevät luokanopettajien työtavoissa.

## Taulukko 5

### *Työtavat ohjelmoinnin opettamisessa*

Työtavat ohjelmoinnin opettamisessa		
Yhteistoiminnalliset työtavat		Itsenäinen työskentely
robotiikka	graafinen ohjelmointi	oppilaan oppikirjan tehtävät
ihmisrobotit (opettaja/oppilas) lattiarobotit (mm. Bee-Bot, Blue-Bot, Pro-Bot, Sphero BOLT) lennokit (Tello-drone)	code.org, LightBot, Scratch (Jr.), LEGO-WeeDoo/ MindStorms EV3 Swift Playgrounds	kynä-paperi-tehtävät

Yhteistoiminnallisissa työtavoissa nousi esiin kaksi muotoa: robotiikka ja graafinen ohjelmointi. Nämä ovat kuitenkin hyvin sidoksissa myös toisiinsa. Seuraavaksi käymme läpi luokanopettajien käyttämät työtavat erikseen.

#### *Yhteistoiminnallinen työskentely*

Norrena (2016, 14) mainitsee toiminnallisiksi työtavoiksi tutkimukset, projektityöt, ryhmätyöt, leikit ja yhteistoiminnallisen oppimisen. Sosiokonstruktivismissa olennaista on se, että tietoa rakennetaan sosiaalisesti ja kognitiivisesti. Opiskelijan rooli on olla vuorovaikutuksessa muiden kanssa ja osallistua yhteiseen toimintaan. (Kauppila, 2007, 35.) Sosiokonstruktivistinen oppimiskäsitys korostuu myös perusopetuksen opetussuunnitelmassa (Opetushallitus, 2014, 17). Luokanopettaja (H1) korostaa erityisesti yhdessä tekemistä myös ohjelmoinnin opetuksessaan, jolloin oppiminen on rikkaampaa.

Tärkeintä on se, että tehdään yhdessä, se on mun mielestä niinkun aivan elinehto tässä kehitystyössä, tässä ohjelmoinnin opettamisessa on se, että tehdään yhdessä ne asiat, pienissä ryhmissä tai pareittain, mutta semmonen yksinäinen tuhartaminen se saattaa kuulua sitten myöhemmin ehkä yläluokille. Yhdessä tekemisessä on hyviä juttuja, kun se oppiminen

laajenee ja vahvistuu eli on paljon rikkaampaa, jos yhdessä saadaan käydä sitä keskustelua, yhdessä ratkaistaan ongelmaa. (H1)

Luokanopettajat käyttivät ohjelmoinnin opetuksessaan paljon robotiikkaa ja graafisia ohjelmointikieliä. Ohjelmoinnissa apuna voidaan käyttää erilaisia ohjelmoitavia robotteja konkretisoimaan ja havainnoimaan algoritmista ajattelua. Ohjelmoitavat robotit edistivät algoritmisen ajattelun kehittymistä positiivisesti (Muñoz-Repiso & Caballero-González, 2019, 69). Robotiikan puolelta mainittiin erityisesti ohjelmoitavat TELLO-drone-lennokit, Sphero-, Pro-Bot-robotit. Drone-lennokkeja toinen luokanopettajista (H1) kertoi käyttävänsä erilaisissa projekteissa, joihin yhdistyivät jokin tarina ja ongelmanratkaisua.

Tykkään erityisesti, että jos meillä on tällaisia droneja, tällaisia robotteja, että on semmoinen tarina, että meillä on tuolla vuorella semmoinen myrkyllinen alue ja sinne ei voi lentää, että sieltä pitää käydä pelastamassa äkkiä tyypit ja sitten ohjelmoidaan täältä drone, joka menee ja laskeutuu, ja sit se nousee ja kääntyy, ottaa kuvan ja lentää tänne. Ja se pitäisi kaikki tapahtua ja toteutua yhdellä napinpainalluksella. (H1)

Luokanopettajan (H1) mukaan on tärkeää, että ohjelmointia tuodaan esille konkreettisin välinein, esimerkiksi erilaisilla roboteilla, jolloin syy-seuraus-suhde on havaittavissa paremmin. Matematiikan oppiaineen sisältöalueissa korostetaan syy-seuraus-suhteen havainnointia, ja että ohjelmoinnin alkeisiin tutustutaan laatimalla vaiheittaisia toimintaohjeita, joita myös testataan (Opetushallitus, 2014, 128–129, 235–236).

Aika pian pääsisi tekemään niinku robottien pariin mä jotenkin uskon siihen, että siinä se syy-seuraus, tavallaan semmoinen tekojen ja käskyjen seuraukset näkyy käytännössä. (H1)

Matematiikan oppiaineen sisältöalueissa mainitaan, että oppilas suunnittelee ja toteuttaa ohjelmia graafisessa ohjelmointiympäristössä (Opetushallitus, 2014, 128–129, 235–236). Luokanopettajien ohjelmoinnin opetuksessa graafiset ohjelmointiympäristöt olivat keskeisessä roolissa. Graafisia

ohjelmointiympäristöjä käytetään myös ohjelmoitavien robottien yhteydessä, jolloin ohjelmointi toteutetaan graafisen liittymän kautta ja suoritettava ohjelma välitetään robotille bluetooth-yhteyden avulla. Luokanopettaja (H2) haluaa haastaa oppilaita itse hoksaamaan ratkaisuja antamatta liikaa vinkkejä ja kannustaa löytämään erilaisia ratkaisuvaihtoehtoja.

Scratchia oon paljon käyttänyt niin siellä että mitä me halutaan siellä toteuttaa ja että miten me voitaisiin saada se toimimaan, mutta semmoisen tutkimuksen kautta. Mun tapa on toimia silleen, että mä esittelen jonkun vähän alkuun, että tässä on mikäkin juttu ja vähäsen semmoisia vihjeitä antaa, ettei ne ole ihan sormi suussa, mutta ei liikaa kerro, että pitää antaa joku haaste, et ratkaise tämä ongelma ja sitten ne joutuu itse miettimään ja sitten hoksata, että ratkaisuja on monia päästä samaan lopputulokseen. (H2)

Jonkun verran oon käyttänyt myös sitä code.orgia, että siellähän on valmiita oppitunteja tai siis semmoisia kursseja, että sitten pystyy tekemään ja siellähän sitten on suuntaan ja sijaintiin liittyvät taidot, niin nehän on ihan kivoja siihen. (H2)

Luokanopettajan (H1) mukaan ohjelmoinnin opettamisessa on tärkeää eriyttäminen ja oppilaiden välinen vertaistuki. Matematiikan kumulatiivisen luonteen vuoksi on tärkeää, että opettajalla on tietoa siitä, mitä oppija osaa jo aiemmin, jotta voidaan rakentaa uutta tietoa.

Eriyttäminen on isossa roolissa, osalla kokemusta enemmän ja osalla vähemmän näistä digitaalisista laitteista ja ylipäättään tämmösestä ajattelusta, joka on kehittynyt siinä ja sitten tulee lapsia, joilla ei ole ollenkaan tämmöstä, niin kaikille pitää avata väylä ja antaa mahdollisuus kehittyä siinä just niin hyväksi kuin pystyy. (H1)

Algoritmisen ajattelun kehittymisen havainnointiin luokanopettajat mainitsevat kielentämisen, robotiikan ja oppimisanalytiikan hyödyntämisen.

Tulee ainakin mieleen tämä oppimisanalytiikka, mä pidän sitä ainaki sellasena tulevaisuuden taitona, se on kuitenkin varmaan helposti mitattavissa oleva asia ja tulkittavissa ja sieltä saa niinku tämmöisen

oppimisanalytiikan kautta, voi rakentaa tehtäviä, missä tarvitaan tämmöistä algoritmista ajattelua. (H1)

Tuo robotiikka on keino visualisoida sitä osaamisen tasoa eli kyllä minusta tuota tämmöinen, kun ryhmissä tehdään ja siinä on se vertaistuki mukana ja oppilaat touhuu vaikka kolmen ryhmissä, niin kyllä siinä opettajalla on mahdollisuus seurata vierestä ja tehdä havaintoja siitä, että kenellä tämä kehittyy ja kenellä ei ja kenellä on niitä haasteita ja ryhmiä muotoillaan sitten sen mukaan, että vertaistuki saadaan niinku kohtaamaan oikealla tavalla. Vertaistuki on tavallaan se minkä se korvaa, kun on hyvä toiminnallinen oppimisympäristö ja lapset on yhdessä tutkimassa, touhuamassa, niin opettajalla on aikaa seurata ja arvioida sitä menestystä ja tuommonen robotiikka tekee sen näkyväksi, sen osaamisen tason. (H1)

Siis kyllä se näkyy noissa projekteissa eli että minkälaisia ja kuinka hienoja projekteja pystyy tuottaa. Mä täällä kuljen ja käyn tsekkailemassa ja keskustelen niiden kanssa, jos vaikka ne kysyy johonkin ongelmaan apua, että en osaa ratkaista ja että miten tässä on, että kuinka paljon esimerkiksi tarvitsee sitä johdattelua tai vinkkejä. Toi tommonen kielentäminen ja just vuorovaikutuksessa, niin siinä sä havaitset, että mitenkä se on tajunnut sen. Ja tavallaan robotin avulla sä pystyt sen sitten niin kuin visuaalisesti todentamaan toiselle, että näin mä sen ajattelin. (H2)

Kuten yllä olevissa luokanopettajan kommentteista on luettavissa, robotiikan hyödyntäminen ohjelmoinnin opettamisessa on tärkeää myös algoritmisen ajattelun kehittymisen kannalta. Kaiken kaikkiaan luokanopettajien kommentteissa painottuu vuorovaikutteisuus ja yhdessä oppiminen sekä oppilaan aktiivinen rooli oppimistilanteissa.

### *Itsenäinen työskentely*

Itsenäinen työskentely ohjelmoinnin osalta keskittyy ohjelmointitehtävien tekemiseen yksin. Oppimateriaalien oppilaan oppikirjojen ohjelmointitehtävät ovat lähes kaikki yksin tehtäviä kynä-paperi-tehtäviä, joista tyyppiesimerkit löytyvät kuvista 1 ja 5. Koska haastateltavamme luokanopettajat eivät käyttäneet oppimateriaalien tarjoamia ohjelmointitehtäviä ohjelmoinnin opetuksessaan, emme voi tämän tutkimuksen perusteella kertoa luokanopettajien kokemuksia niistä muuta kuin, että joskus oppimateriaalien tehtäviä ja harjoituksia voidaan käyttää lisätehtävinä tai uusien vinkkien ja ideoiden keräämiseen.

Myös tutkimustulokset osoittavat, että pelkällä oppikirjojen täyttämällä ei saavuteta parhaita mahdollisia oppimistuloksia. Perkkilä (2002, 172–173) toteaa, että matematiikassa oppikirjan käyttö voi rajoittaa opettajan omaa ajattelua hänen opettaessa oppikirjan avulla. Myös Heinonen (2005, 31–34) mainitsee, että kirjat ja niiden tehtävät eivät edistä oppilasta konstruoimaan.

## 8 POHDINTA

### 8.1 Tulosten tarkastelu ja johtopäätökset

Tutkimuksemme numeerisista tuloksista on luettavissa, että ohjelmointia opetetaan matematiikan oppimateriaaleissa melko suppeasti, kun huomioidaan pelkästään varsinaiset ohjelmointitehtävät. Toisaalta voidaan miettiä, kuinka suuren roolin ohjelmointi, yhtenä osa-alueena matematiikan oppimateriaaleissa, tulisi saada. Olettamuksena voidaan varsinaisten ohjelmointitehtävien määrää pitää kuitenkin suhteellisen sopivana verrattuna matematiikan muihin perinteisiin osa-alueisiin, esimerkiksi geometriaan. Tämä olettaus vaatisi kuitenkin todentamispohjaa lisätutkimuksilla. Vaikka näemme varsinaisten ohjelmointitehtävien määrän vähäisenä, on kuitenkin huomioitavaa, että ohjelmointiin vahvasti liittyvä algoritmisen ajattelu on osa ohjelmoinnin opettamista. Wingin (2006) mukaan algoritmisen ajattelun taidot edustavat yleisesti sovellettavaa asenetta ja taitoa kaikille, ei vain tietotekniikan ammattilaisille. Myös Liukas ja Mykkänen (2014) korostavat, että ohjelmoinnin opetus peruskoulutasolla tulisi olla algoritmisen ajattelutaidon kehittämistä.

Algoritmista ajattelua kehittäviä ja tukevia tehtäviä on tutkituissa oppimateriaaleissa todella paljon. Erityisesti ongelmanratkaisua ja loogista päättelyä vaativia tehtäviä on oppimateriaaleissa runsaasti. Aho (2011, 2) määrittelee algoritmisen ajattelun ajattelunprosessiksi, jossa ongelman ratkaisut voidaan esittää laskennallisina vaiheina ja algoritmeina. Ohjelmointia ja siihen liittyvää ajattelutaitoa kehitetään ja tuetaan siis monella tavalla oppimateriaaleissa, usein jopa tiedostamatta sen hyötyjä ohjelmointitaitoihin. Kansainvälisen nuorten monilukutaidon tutkimuksen (ICILS, *International Computer and Information Literacy Study*) mukaan ohjelmoinnillisen ajattelun osaaminen ei viittaa ainoastaan ohjelmointitaitoihin vaan myös ymmärrykseen siitä, mitä ohjelmoimalla voidaan saada aikaiseksi, miten ohjelmoitavat laitteet toimivat sekä miten ohjelmoinnillisia menetelmiä voidaan käyttää erilaisten ongelmien ratkaisuisissa eri tilanteissa (Leino ym., 2019, 11). Opettajan oppaista

löytyy myös ohjeita ja vinkkejä siihen, kuinka esimerkiksi ongelmanratkaisutehtäviä tai suunta- ja sijaintikäsitteiden hahmottamista voisi harjoitella. Myös Leppäaho (2007, 16) toteaa, että oppilaan oppikirjoissa ei juurikaan korosteta ongelmanratkaisun opettamista, jolloin se jää opettajan käsiteltäväksi.

Osassa oppimateriaaleja varsinaiset ohjelmointitehtävät on erillisissä ohjelmointijaksoissa ja on näin ollen helposti löydettävissä, osassa ne on sijoitettu jonkin muun jakson, kuten geometrian yhteyteen. Tuhattaituri-oppimateriaalin opettajan oppaissa oli erikseen listattu myös tehtäviä, jotka oppikirjan tekijöiden mielestä tukevat ohjelmoinnillista ajattelua. Tämän tyylinen opastus oppikirjoja käyttävälle luokanopettajalle, jolla ei ole ohjelmoinnista juurikaan kokemusta, on varmasti hyödyllistä.

Algoritmisen ajattelun kehittyminen jaettiin neljään osatekijään; ongelmanratkaisutaidot ja looginen päättely, toimintaohjeen noudattaminen, suunta- ja sijaintikäsitteet sekä kielentämistehtävät. Näitä osatekijöitä peilattiin haastattelujen sisältöihin ja niistä löydettiin paljon yhtäläisyyksiä. Näin saatiin sidottua algoritmisen ajattelun kehittymiseen liittyvät osatekijät oppimateriaalien ja luokanopettajien omien kokemusten kesken yhteen. Luokanopettajien puheissa erityisesti ongelmanratkaisutaidot nousivat esiin ja niiden harjoittelua voidaan sisällyttää moniin erilaisiin aiheisiin ja hetkiin.

Toiminnallisuus varsinaisissa ohjelmointitehtävissä oppilaan oppikirjojen sisällöissä jää niukaksi, jolloin ne keskittyvät lähes kokonaan kynä-paperitehtäviin. Norrenaa (2016, 14) lainaten toiminnallinen oppiminen antaa monenlaisia kokemuksia, oivalluksia ja onnistumisen kokemuksia, mikä taas lisää motivaatiota koulunkäyntiä kohtaan. Opettajan oppaan puolella toiminnallisia ohjelmointitehtäviä on vastaavasti lähes kaikki. Algoritmista ajattelua kehittävästä ja tukevista tehtävistä myös suurin osa on toiminnallisia, kun mukaan on laskettu erilaiset pelit ja leikit. Pelien ja leikkien yhteydessä oppilaat oppivat toimimaan vaiheittaisten toimintaohjeiden mukaan ja usein myös ratkomaan erilaisia ongelmia. Myös perusopetuksen opetussuunnitelmassa korostetaan, että lapset oppivat leikin kautta, koska se on



heille luontainen tapa oppia. Oppimisympäristöt on hyvä pitää monipuolisina, jotta ne mahdollistavat leikin ja luovan toiminnan. (Opetushallitus, 2014, 23.) Haastatellut luokanopettajat korostivat ohjelmoinnin opetuksessaan toiminnallisuutta ja erityisesti erilaisten robottien ja graafisten ohjelmointiympäristöjen käyttöä. Luokanopettajat painottivat yhteistoiminnallisen oppimisen merkitystä sekä eriyttämisen ja vertaistuen arvostusta. Repo-Kaarenon (2007, 35) mukaan yhteistoiminnallisen oppimisen keinoin yritetään parantaa sosiaalisia taitoja sekä harjoittamaan ryhmätyöskentelyä. Lisäksi luokanopettajien puheissa korostuivat opetuksessa tarinallisuus ja ongelmanratkaisutaitojen kehittäminen erilaisten haasteiden muodossa.

Vaikka tutkimuksemme luokanopettajat eivät oppimateriaalien tarjoamia ohjelmointitehtäviä juurikaan käyttäneet, on niistä varmasti hyötyä opettajille, jotka eivät ole ohjelmointiin syvemmin perehtynyt. Tulosten perusteella sekä omien kokemustemme mukaan opettajankoulutus ei vastaa riittävällä tasolla ohjelmoinnin opettamisen vaatimuksiin. Opettajan tiedot ja taidot ohjelmoinnista ja sen opettamisesta jäävätkin opettajan oman aktiivisuuden ja vapaaehtoisen kouluttautumisen varaan. Opettajankoulutukseen olisikin hyvä lisätä ohjelmoinnin peruskurssi kaikille opiskelijoille, jotta jokaisella valmistuneella luokanopettajalla olisi jonkinlainen pohjakoulutus ja valmiuksia opettaa ohjelmointia. Lisäksi opettajien kannattaisi osallistua erillisiin jatkokoulutuksiin, joissa he saisivat työkaluja ohjelmoinnin mielekkääseen opetukseen. Myös robotiikkaan opettajien tulisi tarttua rohkeasti, sillä sen avulla ohjelmoinnista saadaan innostavaa ja konkreettista toimintaa. Tutkimuksien mukaan (mm. Muñoz-Repiso & Caballero-Gonzálezin 2019, 65, 69; Khanlari 2013) robotiikka edistää algoritmisen ajattelutaidon kehittymistä ja esimerkiksi ongelmanratkaisutaitoja.

## 8.2 Tutkimuksen arviointi

Tutkimuksen luotettavuutta ja toistettavuutta arvioitaessa on otettava erityisesti huomioon tutkijoiden tapa laskea algoritmista ajattelua kehittäviä ja tukevia tehtäviä ja harjoitteita oppimateriaaleissa. Algoritmista ajattelua tukeviin ja kehittäviin tehtäviin on tässä tutkimuksessa laskettu mukaan ongelmanratkaisu- ja kielentämistehtävät, suunta- ja sijaintikäsitteiden hahmottaminen sekä loogista päättelyä vaativat ja toimintaohjeen noudattamista vaativat tehtävät. Näitä tehtäviä laskettaessa on otettava huomioon tulkinnanvaraisuus siitä, kuuluuko tehtävä edellä mainittuihin ja tukeeko ja kehittääkö se näin ollen algoritmista ajattelua. Olemme pyrkineet tarkasti noudattamaan kyseisten tehtävätyyppien tunnusmerkkejä, mutta ajoittain tehtävien luokittelu on ollut hankalaa ja joku toinen tutkija olisi voinut päättää toisin. Hirsjärven, Remeksen ja Sajavaaran (2009, 232) mukaan tutkijan on huolehdittava, että tutkimuksen vaiheet on tehty ja selostettu tarkasti ja huolellisesti. Uskomme silti, että jotkin tulkintakysymykset ja lisäksi mahdolliset laskuvirheet eivät vaikuta tulosten kokonaiskuvaan merkitsevästi. Varsinaiset ohjelmointitehtävät olivat oppimateriaaleissa helposti löydettävissä ja niissä virhemarginaali on hyvin pieni.

Tutkimuksen rajoituksina mainittakoon lukujonotehtävien poissulkeminen algoritmista ajattelua kehittävästä ja tukevasta tehtävästä sekä toiminnallisuuden puolella yksin tehtävät kynä-paperi-tehtävät. Lukujonotehtävät rajattiin pois niiden runsauden takia, koska tutkimusaineistomme keskittyi paljolti alkuopetuksen oppimateriaaleihin, joissa lukujonotehtäviä on todella paljon. Vuorisen (1993, 179–180) mukaan toiminnallisuuden rajaaminen ei ole yksiselitteistä, sillä se liittyy monenlaisiin työtapoihin, joissa on mahdollisuus toiminnalliseen aktiivisuuteen. Tässä tutkimuksessa toiminnallisuuden määritelmän tuli sisältää konkreettista toimintaa esimerkiksi omaa kehoa käyttäen tai robotiikkaa hyödyntäen. Tutkimuksen kvantitatiiviseen osuuden arvot on laskettu oppilaan oppikirjoista ja opettajan oppaista. Koska digiopetusmateriaalien sisältö on hyvin samankaltainen kuin opettajan oppaissa, toiston välttämiseksi digiopetusmateriaalit rajattiin pois laskennasta.

Eräänlaisena huomionarvoisena seikkana subjektiivisuustarkastelussa voidaan pitää toisen tutkijan taustaa. Petrillä on edellisen diplomi-insinöörikoulutuksen ja työkokemuksen kautta melko paljon kokemusta ohjelmoinnista ja sen opettamisesta alakoulussa. Hirsjärven (2016, 310) mukaan tutkijan on tavoiteltava objektiivisuutta ja pitämään tulosten tulkinta puolueettomana ja tasapuolisena. Pyrimme mahdollisimman hyvin kuitenkin sivuuttamaan toisen tutkijan taustan ja huomioimaan esimerkiksi haastatteluiden sisällöt mahdollisimman autenttisina.

On tärkeää, että tutkittavien yksityisyys ja tietosuoja säilyvät tutkimusta tehdessä. Pidimme siitä huolta, että tutkittavien henkilöllisyydet eivät ole tunnistettavissa esimerkiksi sillä, että annoimme tutkittaville tunnistekoodit H1 ja H2. Haastateltavien taustatiedoista julkaistiin vain oleelliset tiedot, kuten opettajakokemus, koulutukset ja ohjelmoinnin opetuksen kokemus.

Tutkimus toteutettiin hyviä tieteellisiä käytäntöjä noudattaen. Tutkimuksen tekeminen yhdessä vahvisti tutkimuksen luotettavuutta, sillä pystyimme yhdessä jakamaan ja pohtimaan tutkimusprosessin vaiheita ja ongelmakohtia. Myös aineistoa pystyttiin tutkimaan molempien näkökulmat huomioiden, mikä loi vahvistusta tutkimuksemme luotettavuudelle.

Aineisto alkaa saavuttaa saturaation eli kylläntymisen, kun uutta tietoa ei enää synny (Eskola & Suoranta, 2014, 62). Näemme kuitenkin kahden haastateltavan olevan riittävä otanta haastatteluun, koska tutkimuksemme sisälsi myös määrällistä tutkimustietoa ja halusimme saada sille vahvistusta luokanopettajien kokemusten myötä. Haastatteluissa nousi esiin myös paljon samankaltaista tietoa, joten koimme kahden haastateltavan olevan sopiva määrä tutkimukseemme.

### **8.3 Jatkotutkimusaiheita**

Jatkotutkimusaiheina tutkimuksellemme voisi olla esimerkiksi matematiikan oppimateriaalien tutkiminen yläkoulun vuosiluokilla 7–9. Painotetaanko niissä samoja asioita kuin alakoulussa vai keskitytäänkö niissä jo enemmän jonkin

tietyin ohjelmointikielen opetteluun ja harjoitteluun? Kun ohjelmointia ja algoritmista ajattelukykyä opetetaan ja kehitetään alakoulussa riittävälle tasolle, on yläkoulun puolella viimeistään helpompi siirtyä seuraavalle tasolle. On siis tärkeää, että niin oppimateriaalit kuin opettajankoulutuskin saataisiin sille tasolle, että oppilaalla olisi mahdollista kehittyä ohjelmoinnissa eteenpäin. Tutkimuksemme perusteella molemmissa ja erityisesti opettajankoulutuksessa on kehittämisen varaa.

Toisena jatkotutkimusideana voisi olla algoritmisen ajattelun kehittymisen seuraaminen jollain peruskoulun vuosiluokalla. Tämä vaatisi pidempiaikaista seuraamista ja hyvien kehitystä arvioivien keinojen esiintuomista ja testaamista. Tämän tutkimuksen perusteella esimerkiksi kielentämistä ja robotiikkaa pidettiin hyvänä keinona havainnollistamaan oppijan algoritmisen ajattelun kehittymistä.

Tutkimuksessa havaittiin, että oppilaan oppikirjat eivät tue toiminnallisuutta ohjelmoinnin opetuksessa. Tästä saisi tehtyä mielenkiintoisen jatkotutkimuksen kartoittamalla tarkemmin luokanopettajien käyttämiä toiminnallisia keinoja ohjelmoinnin opetuksessa ympäri Suomea, sillä tämän tutkimuksen haastateltavat olivat samalta alueelta ja asiantuntijaopettajia tässä aiheessa. Olisi myös kiinnostavaa selvittää, onko resursseilla merkitystä tähän esimerkiksi laitehankintojen, kuten robottien osalta. Jatkotutkimuksia voisi tehdä eri aineistonkeruumenetelmin, näin tutkimuksista voisi nousta esiin hyvinkin erilaisia näkökulmia ja tuloksia.

## LÄHTEET

- Aho, A. V. (2011). Ubiquity symposium: Computation and Computational Thinking. *Ubiquity*, 1. <https://doi.org/10.1145/1922681.1922682>
- Bers, M. U. (2010). The TangibleK Robotics Program: Applied Computational Thinking for Young Children. *Early Childhood Research and Practice*, 12(2). <https://files.eric.ed.gov/fulltext/EJ910910.pdf>
- Bocconi, S., Chiocciariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2018). Developing computational thinking in compulsory education –Implications for policy and practice. *European Commission, Joint Research Centre*. <https://doi.org/10.2791/792158>
- Caldwell, H., & Smith, N. (2017). Teaching computing unplugged in primary schools. *SAGE Publications, Inc.*, <https://doi.org/10.4135/9781473984332>
- Creswell, J. W. & Plano Clark, V. L. (2007). *Designing and conducting mixed methods research*. Thousand Oak: Sage.
- Dewey, John (1938). *Experience and Nature*. Kappa Delta Pi.
- Dewey, John (1957). *The School and Society*. (K. Kajava, suom.). Otava. (Alkuperäisteos julkaistu 1915).
- Eskola, J. (2018). Laadullisen tutkimuksen juhannustaiat: laadullisen aineiston analyysi vaihe vaiheelta. Teoksessa R. Valli (toim.), *Ikkunoita tutkimusmetodeihin 2. Näkökulmia aloittelevalle tutkijalle tutkimuksen teoreettisiin lähtökohtiin ja analyysimenetelmiin*. (5. painos, s. 209–231). PS-kustannus
- Eskola, J. & Suoranta, J. (2014). *Johdatus laadulliseen tutkimukseen*. Vastapaino.
- Fant, K. (1993). A Critical review of the Notion of the Algorithm in Computer Science. *Proceedings of the 21st Annual Computer Science Conference*. <https://doi.org/10.1145/170791.170794>
- Futschek, G. (2006). *Algorithmic Thinking: The Key for Understanding Computer Science*. (s. 159–168). ISSEP 2006. [https://doi.org/10.1007/11915355\\_15](https://doi.org/10.1007/11915355_15)
- Heikkinen, H. (2005). *Draamakasvatus: opetusta, taidetta, tutkimesta!* Minerva.

- Heinonen, J.-P. (2005). *Opetussuunnitelmat vai oppimateriaalit: peruskoulun opettajien käsityksiä opetussuunnitelmien ja oppimateriaalien merkityksestä opetuksessa*. [väitöskirja, Helsingin yliopisto]. Tutkimuksia 257.  
<http://urn.fi/URN:ISBN:952-10-1995-6>.
- Hesse-Biber, S. (2010). *Mixed methods research: Merging theory with practice*. The Guilford Press. <https://doi.org/10.1177/1473325011433761b>
- Hiidenmaa, P. A. (2015). Oppikirjojen tutkimus. Teoksessa H. Ruuska, M. Löytönen, & A. Rutanen (toim.), *Laatua! Oppimateriaalit muuttuvassa tietoympäristössä*. (s. 27–40). Suomen tietokirjailijat.  
[https://www.suomentietokirjailijat.fi/media/laatua\\_oppimateriaalit\\_2015\\_korjattu\\_web.pdf](https://www.suomentietokirjailijat.fi/media/laatua_oppimateriaalit_2015_korjattu_web.pdf)
- Hirsjärvi, S. & Hurme, H. (2008). *Tutkimushaastattelu: teemahaastattelun teoria ja käytäntö*. Gaudeamus.
- Hirsjärvi, S., Remes, P., & Sajavaara, P. (2009). *Tutki ja kirjoita*. (15. uudistettu painos). Tammi.
- Ilomäki, L. (2012). E-oppimateriaalit oppimisen ja opettamisen tukena. Teoksessa: L., Ilomäki (toim.), *Laatua oppimateriaaleihin, E-oppimateriaalit opetuksessa ja oppimisessa*. Opetushallitus, Oppaat ja käsikirjat 2012:5. Juvenes Print – Suomen Yliopistopaino. <https://9pdf.co/document/zkw8vede-laatua-e-oppimateriaaleihin-e-oppimateriaalit-opetuksessa-ja-oppimisessa.html>
- ISTE & CSTA. (2011). *Operational Definition of Computational Thinking for K-12 Education*. International Society for Technology in Education (ISTE).  
[https://cdn.iste.org/www-root/Computational\\_Thinking\\_Operational\\_Definition\\_ISTE.pdf](https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf)
- Johnson, R. B., Onwuegbuzie, A. J., & Turner, L. A. (2007). Toward a Definition of Mixed Methods Research. *Journal of Mixed Methods Research*, 1(2). (s. 112–133).  
<https://doi.org/10.1177/1558689806298224>
- Joutsenlahti, J. (2003). Kielentäminen matematiikan opiskelussa. Teoksessa A. Virta & O. Marttila (toim.), *Opettaja, asiantuntijuus ja yhteiskunta*. (s. 188–196).
- Joutsenlahti, J. & Tossavainen, T. (2018). Matemaattisen ajattelun kielentäminen ja siihen ohjaaminen koulussa. Teoksessa J. Joutsenlahti, H. Silfverberg, & P.

Räsänen (toim.), *Matematiikan opetus ja oppiminen* (s. 410–431). Niilo Mäki Instituutti.

- Joutsenlahti, J. & Vainionpää, J. (2007). Minkälaiseen matemaattiseen osaamiseen peruskoulussa käytetty oppimateriaali ohjaa? Teoksessa K. Merenluoto, A. Virta & P. Carpelan (toim.) *Opettajankoulutuksen muuttuvat rakenteet: Ainedidaktinen symposiumi 9.2.2007*. (s. 184–191). Turun yliopiston kasvatustieteiden tiedekunnan julkaisuja B 77.
- Joutsenlahti, J. & Vainionpää, J. (2008). Oppikirja vai harjoituskirja? Perusopetuksen luokkien 1–6 matematiikan oppimateriaalin tarkastelua MOT-projektissa. Teoksessa: A. Kallioniemi (toim.) *Uudistuva ja kehittyvä ainedidaktiikka. Ainedidaktiikan symposiumi 8.2.2008*. (s. 547–558). Helsingin yliopisto. Soveltavan kasvatustieteen laitos 299.
- Joutsenlahti, J. & Vainionpää, J. (2010). Oppimateriaali matematiikan opetuksessa ja osaamisessa. Teoksessa E.K. Niemi & J. Metsämuuronen (toim.) *Miten matematiikan taidot kehittyvät? Matematiikan oppimistulokset peruskoulun viidennen vuosiluokan jälkeen vuonna 2008*. (s. 137–148). Koulutuksen seurantaraportit 2010:2. Opetushallitus.
- Ainedidaktinen symposium 7.2.2003. Turun yliopisto.
- Jyväskylän yliopisto. (2021). *Opinto-opas 2020–23*. <https://opinto-opas.jyu.fi/2021/fi/opintojakso/otpp1030/>
- Järvilehto, L. (2014). *Hauskan oppimisen vallankumous*. PS-kustannus.
- Järvinen, M. (2011). *Konstruktivistinen oppimiskäsitys opettajan pedagogisena työvälineenä alkuopetuksessa: näkökulmia muutokseen*. [Väitöskirja, Tampereen yliopisto]. <https://urn.fi/urn:isbn:978-951-44-8380-6>
- Kauppila, R. (2007). *Ihmisen tapa oppia: johdatus sosiokonstruktiviseen oppimiskäsitykseen*. PS-kustannus.
- Kekäläinen, O. (2015). Onko automatisointiajattelu paras suomennos käsitteestä "computational thinking"? Teoksessa Viteli, J. & Östman, A. (toim.) *Tuovi 13: Interaktiivinen tekniikka koulutuksessa 2015-konferenssin tutkijatapaamisen artikkelit*. (s. 27–29). Tampereen yliopisto. <https://urn.fi/URN:ISBN:978-951-44-9909-8>

- Khanlari, A. (2013). Effects of Robotics on 21st Century Skills. *European Scientific Journal, ESJ*, 9(27). <https://www.robolab.in/wp-content/uploads/2016/10/1805-5256-1-SM.pdf>
- Knuth, D. E. (1997). *The Art of Computer Programming: Volume 1, Fundamental Algorithms*. (3. painos). Addison-Wesley.  
[https://seriouscomputerist.atariverse.com/media/pdf/book/Art%20of%20Computer%20Programming%20-%20Volume%201%20\(Fundamental%20Algorithms\).pdf](https://seriouscomputerist.atariverse.com/media/pdf/book/Art%20of%20Computer%20Programming%20-%20Volume%201%20(Fundamental%20Algorithms).pdf)
- Korhonen, H. (2021). Perushahmotusta ohjelmointiin 2: opetus, luovuus ja harrastus. *Dimensiolehti* 14.5.2021. <https://dimensiolehti.fi/perushahmotusta-ohjelmointiin-2-opetus-luovuus-ja-harrastus/>
- Koskenkari, S. (2012). *Toiminnallinen oppiminen*. [Koulutusmateriaali]. Liikkuva koulu. Opetushallitus. <https://docplayer.fi/10957690-Toiminnallinen-oppiminen-sari-koskenkari.html>
- Larsson P., Kaarto, H. & Parviainen, M. (2021). *Ohjelmoinnillisen ajattelun työkalut alakoulussa*. Turun yliopiston Oppimisanalytiikan keskus.
- Leino, K., Rikala, J., Puhakka, E., Niilo-Rämä, M., Siren, M. & Fagerlund, J. (2019). *Digiloikasta digitaitoihin: kansainvälinen monilukutaidon ja ohjelmoinnillisen ajattelun tutkimus (ICILS 2018)*. Koulutuksen tutkimuslaitos.  
<https://jyx.jyu.fi/handle/123456789/66250>
- Leppäaho, H. (2007). *Matemaattisen ongelmanratkaisutaidon opettaminen peruskoulussa*. [Väitöskirja, Jyväskylän yliopisto]. <http://urn.fi/URN:ISBN:978-951-39-2732-5>
- Lerikkanen, M-K. (2015). Aapinen elää ajassa. Teoksessa: H. Ruuska, M. Löytönen & A. Rutanen. 2015. *Laatua! Oppimateriaalit muuttuvassa tietoympäristössä*. (s. 91–102). Suomen tietokirjailijat ry.  
[https://www.suomentietokirjailijat.fi/media/laatua\\_oppimateriaalit\\_2015\\_korjattu\\_web.pdf](https://www.suomentietokirjailijat.fi/media/laatua_oppimateriaalit_2015_korjattu_web.pdf)
- Liukas, L. & Mykkänen, J. (2014). *Koodi 2016: ensiapua ohjelmoinnin opettamiseen peruskouluissa*. Lönnberg Print. [https://s3-eu-west-1.amazonaws.com/koodi2016/Koodi2016\\_LR.pdf](https://s3-eu-west-1.amazonaws.com/koodi2016/Koodi2016_LR.pdf)



- Luoma-aho, E. (2010). Matematiikan peruskäsitteiden historia: 1. Algebra ja aritmetiikka. *Matematiikkalehti Solmu*. Helsingin yliopisto.
- Luostarinen, A. & Peltomaa, I-M. (2016). *Reseptit OPSin käyttöön – opettajan opas työssä onnistumiseen*. PS-kustannus.
- Luukkainen, O. (2004). *Opettajuus - ajassa elämistä vai suunnan näyttämistä?* [Väitöskirja, Tampereen yliopisto]. <https://urn.fi/urn:isbn:951-44-5885-0>
- Metsämuuronen, J. (2009). *Tutkimuksen tekemisen perusteet ihmistieteissä*. (4. painos). Gummerus.
- Muñoz-Repiso, A. G.-V. and Caballero-González, Y.-A. (2019). Robotics to Develop Computational Thinking in Early Childhood Education. (s.63–72). *Comunicar: Media Education Research Journal*, 27(59), s. 63–72.  
<https://eric.ed.gov/?id=EJ1211742>
- Norrena, J. (2016). *Ryhmä oppimaan! Toiminnallisia työtapoja ja tehtäväkehyksiä*. PS-kustannus.
- Opetushallitus. (2014). *Perusopetuksen opetussuunnitelman perusteet 2014*. Määräykset ja ohjeet 96. Opetushallitus.
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.  
<https://dl.acm.org/doi/pdf/10.5555/1095592>.
- Papert, S. (1996). An Exploration in the Space of Mathematics Educations. *International Journal of Computers for Mathematical Learning* 1, (s. 95–123).  
<https://doi.org/10.1007/BF00191473>.
- Perkkilä, P. (2002). *Opettajien matematiikkauskomukset ja matematiikan oppikirjan merkitys alkuopetuksessa*. [Väitöskirja, Jyväskylän yliopisto].  
<http://urn.fi/URN:ISBN:978-951-39-5338-6>
- Perkkilä, P., Joutsenlahti, J., & Sarenius, V-M. (2018). Peruskoulun matematiikan oppikirjat osana oppimateriaalitutkimusta. Teoksessa J. Joutsenlahti, H. Silfverberg, & P. Räsänen (toim.), *Matematiikan opetus ja oppiminen* (s. 344–367). Niilo Mäki Instituutti.

- Rainio, P. (2002). Filosofinen keskustelu ja toiminnalliset työtavat elämäntiedon opetuksessa. Teoksessa O. Saloranta (toim.) *Ensimmäiset kouluvuodet: perusopetuksen vuosiluokkien 1–2 opetus*. Opetushallitus.
- Rauste-von Wright, M.L., von Wright, J. & Soini, T. (2003). *Oppiminen ja koulutus*. WSOY.
- Repo-Kaarento, S. (2007). *Innostu ryhmästä: miten ohjata oppivaa yhteisöä*. Kansanvalistusseura.
- Rose, K., Eldridge, S. & Lyman, C. (2015). *The Internet of Things: an Overview: Understanding the Issues and Challenges of a More Connected World*. Internet Society. <https://www.internetsociety.org/wp-content/uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf>
- Saaranen-Kauppinen, A., Lindblad, K., Pekanheimo, E. & Muittari, J. (2018). *Tartu ToiMeen!: Toiminnallisten menetelmien koulutusmateriaali opettajille*. Humanistinen ammattikorkeakoulu. <https://urn.fi/URN:ISBN:978-952-456-303-1>
- Sahlberg, P. & Leppilampi, A. (1994). Yksinään vai yhteisvoimin?: yhdessäoppimisen mahdollisuuksia etsimässä. Helsingin yliopisto, Vantaan täydennyskoulutuslaitos. <https://doi.org/10.33336/aik.96973>
- Soare, R. I. (1999). The history and concept of computability. Teoksessa E. R. Griffor (toim.) *Handbook of Computability Theory*. (s. 3–36). Elsevier.
- Sura, S. (1998). Toiminnallisuus alkukasvatustieteen oppimisen edistäjänä. Teoksessa K. Laine & J. Tähtinen (toim.) *Toiminnallinen oppiminen – oppimisen ohjaaminen esi- ja alkuopetuksessa*. (s. 43–64). Rauman opettajankoulutuslaitos.
- TENK. Tutkimuseettinen neuvottelukunta. (2012). *Hyöä tieteellinen käytäntö ja sen loukausepäilyjen käsitteleminen Suomessa*. [https://www.tenk.fi/sites/tenk.fi/files/HTK\\_ohje\\_2012.pdf](https://www.tenk.fi/sites/tenk.fi/files/HTK_ohje_2012.pdf)
- Tikkanen, P. (2008). *“Helpompaa ja hausempaa kuin luulin”: matematiikka suomalaisten ja unkarilaisten perusopetuksen neljäsluokkalaisten kokemana*. [Väitöskirja, Jyväskylän yliopisto]. <https://jyx.jyu.fi/handle/123456789/18042>.
- Tuomi, J. & Sarajärvi, A. (2018). *Laadullinen tutkimus ja sisällönanalyysi*. Tammi.
- Turun yliopisto. (2021). *Opinto-opas 2020–2022*. <https://opas.peppi.utu.fi/fi/opintojakso/LUOT0073/13825>

- Uudet lukutaidot. (2020). *Uudet lukutaidot -kehittämisohjelma*. Opetus- ja kulttuuriministeriö. <https://uudetlukutaidot.fi/>
- Valli, R. (2015). *Johdatus tilastolliseen tutkimukseen*. PS-kustannus.
- Vuorinen, I. (1993). *Tuhat tapaa opettaa – Menetelmäopas opettajille, kouluttajille ja ryhmän ohjaajille*. Resurssi.
- Weintrop, D. (2019). Block-Based Programming in Computer Science Education. *Communications of the ACM*, 62(8). (s. 22–25). <https://doi.org/10.1145/3341221>
- Wing, J. M. (2006). Computational Thinking. *Communications of the ACM*, 49(3). (s. 33–35). <https://doi.org/10.1145/1118178.1118215>.
- Wing, J. M. (2010). Computational Thinking: What and Why? <https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>
- Yadav, A., Hong, H., & Stephenson, C. (2016). Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms. *TechTrends*, 60, (s. 565–568). <https://doi.org/10.1007/s11528-016-0087-7>
- Öystilä, S. (2003). Toiminnallisen opetuksen perustan rakentajia – John Dewey, Kurt Lewin, Jacob Levy Moreno, David Kolb ja Jack Mezirow. Teoksessa E., Poikela ja S., Öystilä (toim.) *Yliopistopedagogiikkaa kehittämässä – kokeiluja ja kokemuksia*. Tampere University Press. <https://urn.fi/URN:NBN:fi:uta-201309101351>

### **Tutkimusaineistolähteet**

- Forsback, M., Kalliola, A., Tikkanen, A. & Waneus, M-L. (2022). *Tuhattaituri 1a, 1b, 2a, 2b*. Verkkomateriaali. Otava.
- Häkkinen, K., Sohlman, L., Kaleva, T. & Patrikainen, S. (2022). *Milli 2A, 2B*. Verkkomateriaali. Sanoma Pro.
- Häkkinen, K., Sohlman, L., Kaleva, T. & Similä, M. (2022). *Milli 1A, 1B*. Verkkomateriaali. Sanoma Pro.
- Hänninen, L., Malinen, K., Ranta, P. & Vallo, L. (2022). *Milli 3A, 3B, 4A, 4B*. Verkkomateriaali. Sanoma Pro.
- Kalm, M., Luoma, M., Packalén, M., Perälä, P. & Tapiainen, T. (2022). *Oivaltaja 3a*. Verkkomateriaali. Otava.

Kalm, M., Luoma, M., Perälä, P. & Tapiainen, T. (2022). *Oivaltaja 3b*.

Verkkomateriaali. Otava.

Kiviluoma, P., Nyrhinen, K., Perälä, P., Rokka, P., Salminen, M. & Tapiainen, T.

(2022). *Tuhattaituri 3a, 3b*. Verkkomateriaali. Otava.

Rautio, H-K., Salminen, M., Stenberg, H. & Vehmas, H-R. (2022). *Oivaltaja 1a, 1b*.

Verkkomateriaali. Otava.

Rinne, S., Sintonen, A-M., Uus-Leponiemi, M. & Uus-Leponiemi T. (2022). *Kymppi 1*

*syksy, 1 kevät, 2 syksy, 2 kevät, 3 syksy, 3 kevät, 4 syksy, 4 kevät*. Verkkomateriaali.

Sanoma Pro.

## LIITTEET

## Liite 1. Ohjelmointi matematiikan tavoitteissa ja sisällöissä vuosiluokilla 1-6

MATEMATIIKAN OPPIAINE 1.-6. lk.		LAAJA-ALAISET TAVOITTEET			
TAVOITTEET		SISÄLLÖT		TAVOITTEET	
1.-2. lk.	3.-6. lk.	1.-2. lk.	3.-6. lk.	1.-2. lk.	3.-6. lk.
T3 kannustaa oppilasta esittämään ratkaisujaan ja päätehtävään konkreettisin välinein, piirroksin, suullisesti ja kirjallisesti myös tieto- ja viestintäteknologiaa hyödyntäen	T3 ohjasta oppilasta kehitetään taitoan esittää kysymyksiä ja tehdä perusteltuja päätelmiä havaintojensa pohjalta	S1 Ajattelun taidot Oppilaille tarjotaan mahdollisuuksia löytää yhtäläisyyksiä, eroja ja säännönmukaisuuksia. Vertailtaan, luokitellaan ja asetetaan järjestykseen sekä luokitellaan syy- ja vaikutteita. Harjoitellaan tarkastelemaan matemaattisia tilanteita eri näkökulmista. Tutustuminen alkan laatimalla vaihteellaisia toimintaohjelmia, joita myös testataan.	S1 Ajattelun taidot Kehitetään oppilaiden taitoja löytää yhtäläisyyksiä, eroja ja säännönmukaisuuksia. Syyennetään taitoa vertailla, luokitella ja asettaa järjestykseen, etsiä vaikutteita systemaattisesti, havaita syy- ja seuraussuhteita sekä yhteyksiä matemaattisissa. Suunnitellaan ja toteutetaan ohjelmia graafisessa ohjelmuuttympparistossa.	L1 Ajattelu ja oppimaan oppiminen Ihmiset, elävät, oivaltamiselle, uuden löytämiselle ja keksimiselle, mielikuvitukselle sekä oppimisen tälle on tilaa.	L1 Ajattelu ja oppimaan oppiminen Oppilaita ohjataan huomaamaan asioiden välisiä vuorovaikutussuhteita ja keskinäisiä yhteyksiä.
T4 ohjasta oppilasta kehitetään päätehtävään ja ongelmanratkaisutaitojen	T4 kannustaa oppilasta esittämään päätelyään ja konkreettisin välinein, piirroksin, suullisesti ja kirjallisesti myös tieto- ja viestintäteknologiaa hyödyntäen	Harjoitellaan seurauksia, havaita syy- ja vaikutteita tarkastelemaan matemaattisia tilanteita eri näkökulmista. Tutustuminen alkan laatimalla vaihteellaisia toimintaohjelmia, joita myös testataan.	S2 Geometria ja mittaus Kehitetään oppilaiden taitoa hahmottaa kolmiulotteista ympäristöä ja havaita sen tason geometriaa. Harjoitellaan suurta- ja sijaintikäsittelyiden käyttöä.	L2 Ajattelu ja oppimaan oppiminen Ihmisten toimintaympäristö on osa maailmaa, jota voimme muuttaa. Ihminen on osa luonnon ympäristöä, jota voimme suojella ja huolehtia. Ihminen on osa yhteiskuntaa, jota voimme kehittää ja parantaa.	L2 Ajattelu ja oppimaan oppiminen Oppilaita rohkaistaan käyttämään kurvittelevuutta luovien ratkaisujen ja olemassa olevien rajojen ylittämiseen.
T5 ohjasta ja tukea oppilasta ongelmanratkaisutaitojen kehitämisessä	T5 ohjasta ja tukea oppilasta ongelmanratkaisutaitojen kehitämisessä	Tutustuminen alkan laatimalla vaihteellaisia toimintaohjelmia, joita myös testataan.	S3 Algebra Tutkitaan lukujonon säännönmukaisuutta säännön mukaan. Tutustutaan tuntemattoman käsitteeseen. Tutkitaan yhtälöä ja etsitään yhtälön ratkaisuja käyttämällä ja kokeilemalla.	L3 Tieto- ja viestintäteknologian osaaminen Leikkiä perustuen työskentely on edelleen keskeistä. Tieto- ja viestintäteknologian käyttöä harjoitellaan nappaintaitoja sekä muita tekstin tuottamisen ja käsittelyn perustaitoja. Oppilaita saavat ja jaksavat keskittyä tekemään digitaalisen median kanssa työskentelyä sekä ikäkaudelle sopivasta ohjelmuuttympparistosta. Pöytätyötä hyödynnetään oppimisen edistämiseksi.	L3 Tieto- ja viestintäteknologian osaaminen Leikkiä perustuen työskentely on edelleen keskeistä. Tieto- ja viestintäteknologian käyttöä harjoitellaan nappaintaitoja sekä muita tekstin tuottamisen ja käsittelyn perustaitoja. Oppilaita saavat ja jaksavat keskittyä tekemään digitaalisen median kanssa työskentelyä sekä ikäkaudelle sopivasta ohjelmuuttympparistosta. Pöytätyötä hyödynnetään oppimisen edistämiseksi.
T12 harjaamuttaa oppilasta laatimaan vaihteellaisia toimintaohjelmia ja toimimaan ohjeen mukaan	T12 innostaa oppilasta laatimaan toimintaohjelmia tietokoneohjelmoinnissa graafisessa ohjelmuuttympparistossa	Tutustuminen alkan laatimalla vaihteellaisia toimintaohjelmia, joita myös testataan.	S4 Geometria ja mittaus Kehitetään oppilaiden taitoa hahmottaa kolmiulotteista ympäristöä ja havaita sen tason geometriaa. Harjoitellaan suurta- ja sijaintikäsittelyiden käyttöä.	L4 Tieto- ja viestintäteknologian osaaminen Leikkiä perustuen työskentely on edelleen keskeistä. Tieto- ja viestintäteknologian käyttöä harjoitellaan nappaintaitoja sekä muita tekstin tuottamisen ja käsittelyn perustaitoja. Oppilaita saavat ja jaksavat keskittyä tekemään digitaalisen median kanssa työskentelyä sekä ikäkaudelle sopivasta ohjelmuuttympparistosta. Pöytätyötä hyödynnetään oppimisen edistämiseksi.	L4 Tieto- ja viestintäteknologian osaaminen Leikkiä perustuen työskentely on edelleen keskeistä. Tieto- ja viestintäteknologian käyttöä harjoitellaan nappaintaitoja sekä muita tekstin tuottamisen ja käsittelyn perustaitoja. Oppilaita saavat ja jaksavat keskittyä tekemään digitaalisen median kanssa työskentelyä sekä ikäkaudelle sopivasta ohjelmuuttympparistosta. Pöytätyötä hyödynnetään oppimisen edistämiseksi.

## Liite 2. Ohjelmointitehtävät ja algoritmista ajattelua tukevat tehtävät koottuna kaikista oppimateriaaleista

Oppimateriaali	tehtävät yhteensä	ohjelmointitehtävät	oppikirjan ohjelmointia tukevat	toiminnalliset ohjelmointitehtävät	opeopas yhteensä	opeoppaan ohjelmointia tukevat
Milli 1A	552	33	174	16	294	105
Milli 1B	481	23	160	15	245	126
Milli 2A	484	24	160	19	238	128
Milli 2B	488	20	97	18	182	129
Milli 3A	585	13	184	32	280	151
Milli 3B	602	13	165	43	261	173
Milli 4A	553	13	153	35	232	174
Milli 4B	597	12	149	41	248	169
Kymppi 1 S	369	13	70	8	266	128
Kymppi 1 K	419	10	88	10	267	141
Kymppi 2 S	462	14	111	8	248	158
Kymppi 2 K	465	13	122	6	246	177
Kymppi 3 S	474	12	118	9	225	149
Kymppi 3 K	461	18	102	9	238	176
Kymppi 4 S	502	16	145	7	244	197
Kymppi 4 K	581	14	124	5	243	185
Tuhattaituri 1A	408	0	96	6	320	176
Tuhattaituri 1B	444	7	130	12	379	212
Tuhattaituri 2A	465	11	153	12	312	165
Tuhattaituri 2B	492	16	136	19	310	166
Tuhattaituri 3A	559	2	143	13	228	125
Tuhattaituri 3B	574	1	137	13	200	131
Oivaltaja 1A	485	11	116	19	165	69
Oivaltaja 1B	469	0	147	20	122	58
Oivaltaja 3A	587	7	147	6	110	56
Oivaltaja 3B	546	10	145	8	106	65
<b>YHTEENSÄ:</b>	<b>13104</b>	<b>326</b>	<b>3472</b>	<b>409</b>	<b>6209</b>	<b>3689</b>
<b>suhteelliset osuudet:</b>		<b>2,5 %</b>	<b>26,5 %</b>	<b>10,8 %</b>		<b>59,4 %</b>

### Liite 3. Ohjelmointitehtävät ja algoritmista ajattelua tukevat tehtävät oppimateriaaleittain ja kappaleittain.

#### Milli-oppimateriaalit

luku	kpl	MILLI 1A				huom.	
		tehtävät	ohj. tukevat	toim.	oppaan ohj		
1	1	8	1		0	5	
	2	11	2		0	5	
	3	12	5		3	3	
	3	3	3		0	6	projekti
2	4	12	3		2	6	
	5	13	3		2	6	
	6	14	5		3	8	
	7	12	3		1	7	
	8	9	6	1	3	6	
	9	14	5		2	7	
	10	12	3		1	7	
	11	13	2		2	6	
	12	15	3		3	8	
	13	12	2		2	9	
	14	8	1		2	4	
15	11	1		5	8		
16	9	1		4	6	katsastus projekti	
	3	3	3		7	7	katsastus projekti
3	17	11	2		2	7	
	18	13	2		2	5	
	19	8	2		1	5	
	20	14	3		2	9	
	21	11	4		3	7	
	22	13	4		4	8	
	23	10	0		1	4	
	24	10	3		2	5	
	25	11	2		2	6	
	26	12	3	1	1	6	
	27	10	1		1	4	
	28	11	2		1	4	
	29	8	2		3	6	katsastus projekti
	3	3	3		7	7	katsastus projekti
4	30	14	3		1	7	
	31	11	1		2	7	
	32	14	3		1	7	
	33	11	2		2	7	
	34	11	3	1	2	11	
	35	11	8		2	5	
	36	12	3		1	5	
	37	10	3		2	5	
	38	14	4		1	8	
	39	11	2		2	6	
	40	13	4		1	10	
	41	10	2		3	5	
	42	11	2		3	5	
	43	9	4	1	3	5	
44	11	5	1	2	5		
	7	4		5	7	katsastus projekti	
	3	3	2				
OHJ	1	8	8		4	4	ohjelmointi
	2	12	12		5	5	ohjelmointi
	3	13	13		3	3	ohjelmointi
yht.	552	174	16	105	294		

luku	kpl	MILLI 1B				huom.	
		tehtävät	ohj. tukevat	toim.	oppaan ohj		
1	1	11	2		4	9	
	2	12	2		1	4	
	3	13	2		1	8	
	4	11	3		3	5	
	5	11	2		1	5	
	6	12	1		2	6	
	7	11	3	1	3	6	
	8	11	2		4	7	
	9	9	3		4	8	katsastus projekti
		4	4	2			
2	10	11	8		3	7	
	11	10	2		1	5	
	12	9	2		1	4	
	13	7	5	1	3	5	katsastus projekti
	3	3	1				
3	14	12	4		3	6	
	15	10	4		3	4	
	16	10	2	1	4	6	
	17	11	4		3	5	
	18	11	1		3	6	
	19	9	1		2	2	
	20	10	4	1	2	8	
21	8	2		5	6	katsastus projekti	
	2	2	1				
4	22	11	2		3	6	
	23	10	2		4	4	
	24	11	4		3	5	
	25	10	4	1	3	8	
	26	10	3		3	3	
	27	10	3		2	5	
	28	8	2	1	4	7	
	29	11	2		2	4	
	30	8	3		5	6	katsastus projekti
		2	2	0			
5	31	10	5		3	5	
	32	12	7		3	4	
	33	12	4		3	6	
	34	12	1		2	5	
	35	11	1		2	5	
	36	9	4		4	5	katsastus projekti
	3	3	2				
6	37	10	1		2	5	
	38	11	3		1	3	
	39	9	1		1	6	
	40	11	0		2	4	
	41	10	4		2	3	
	42	10	3	1	1	7	
	43	9	2		4	6	katsastus projekti
	3	3	2				
OHJ	1	8	8		3	3	ohjelmointi
	2	8	8		5	5	ohjelmointi
	3	7	7		3	3	ohjelmointi
yht.	481	160	15	126	245		

## Liite 3 jatkuu

MILLI 2A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	8	1		2	3	
	2	12	1		2	6	
	3	10	3		1	4	
	4	9	1		3	5	
	5	10	4	1	1	5	
	6	9	2		3	5	
	7	9	2	1	3	7	
	8	9	2		3	4	
	9	7	1		4	7	katsastus projekti
2	10	12	1		2	6	
	11	12	1		2	5	
	12	12	1		2	5	
	13	12	2		2	4	
	14	7	3	1	2	3	
	15	11	1		3	4	
	16	10	1		1	4	
	17	11	2	1	3	6	
	18	9	2		3	4	
	19	8	1		1	3	
	20	10	2		2	3	
	21	10	5	1	2	8	
	22	10	2		3	6	
	23	8	3		4	6	katsastus projekti
3	24	10	8		6	6	
	25	9	3		4	7	
	26	13	3		3	7	
	27	11	3		3	4	
	28	12	4		4	5	
	29	10	5	1	4	7	
	30	12	9		4	7	
	31	12	10		3	4	
	32	9	6		5	5	katsastus projekti
	33	9	2	4	3	6	
	34	9	1		3	5	
4	35	8	2	1	1	7	
	36	13	2		2	6	
	37	13	2		2	6	
	38	9	1	1	4	4	
	39	13	4		2	4	
	40	10	2		2	5	
	41	11	1		1	4	
	42	10	2		2	3	
	43	9	2	1	3	6	
	44	10	1		4	7	katsastus projekti
	45	3	3	2			
OHJ	1	7	7		3	3	ohjelmointi
	2	8	8		3	3	ohjelmointi
	3	9	9		3	3	ohjelmointi
yh.	484	160		19	128	238	

MILLI 2B								
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.	
1	1	10	1		2	4		
	2	11	2		3	4		
	3	10	2		2	2		
	4	10	1		2	2		
	5	10	1		1	3		
	6	9	2		3	3		
	7	9	2	1	4	5		
	8	9	3		3	4		
	9	10	2		3	3		
	10	11	3	1	3	4		
	11	9	2		5	5	katsastus projekti	
2	12	12	3		2	3		
	13	11	0		1	3		
	14	11	1		2	4		
	15	12	1		1	5		
	16	9	2		4	4	katsastus projekti	
	17	13	1		3	5		
	18	13	1		3	5		
	19	11	1		3	3		
	20	11	1		1	3	5	
	21	11	2		3	3		
	22	10	1		3	4		
3	23	10	1		4	4		
	24	9	2	1	2	5		
	25	10	1		2	4		
	26	10	4		2	4		
	27	12	1		3	4		
	28	9	0		4	6	katsastus projekti	
	29	10	1	2	2	3		
	30	10	0		1	4		
	31	11	2	1	3	6		
	32	11	0		3	4		
	33	11	0		3	4		
4	34	10	2	1	4	6		
	35	9	1		3	3	katsastus projekti	
	36	13	1		2	4		
	37	11	0		2	4		
	38	12	0		2	4		
	39	12	2		3	4		
	40	12	3		3	3		
	41	11	3	1	4	5		
	42	11	1		3	4		
	43	9	2		4	5	katsastus projekti	
	44	3	3		3			
OHJ	1	8	8		4	4	ohjelmointi	
	2	6	6		3	3	ohjelmointi	
	3	6	6		3	3	ohjelmointi	
yh.	488	97		18	129	182		

MILLI 3A								
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.	
1	1	12	4		3	6		
	2	13	2		4	6		
	3	13	2		4	9		
	4	14	3		5	6		
	5	12	3		4	5		
	6	11	2		3	6		
	7	11	2		4	6		
	8	11	2		3	4		
	9	10	3	1	4	4		
	10	11	2		2	4		
	11	11	2		3	4		
	12	11	3		6	6		
	13	9	3	1	2	5		
	14	12	3		3	3	katsastus projekti	
2	15	11	3		1	4		
	16	13	3		3	7		
	17	13	2		3	7		
	18	11	3		4	6		
	19	12	2		2	7		
	20	13	3		3	8		
	21	13	4		2	8		
	22	10	4	1	4	6		
	23	12	4		3	7		
	24	13	4		1	7		
	25	11	5	1	3	5		
	26	12	4		2	4	katsastus projekti	
	27	12	2		2	5		
	28	15	3		4	6		
3	29	13	2		3	6		
	30	12	3	1	5	5		
	31	11	2		3	6		
	32	14	3		4	4		
	33	14	3		5	5		
	34	10	4	1	5	6		
	35	8	4		3	5	katsastus projekti	
	36	13	1		3	4		
	37	12	1		3	7		
	38	12	4		1	6		
	39	11	2		2	11		
	40	9	4	1	2	5		
	41	11	8		3	7		
	42	11	2		2	7		
4	43	11	4		3	7		
	44	10	6	1	2	9		
	45	11	4		1	1	katsastus projekti	
	46	7	7		5			
	OHJ	1	7	7		2	3	ohjelmointi
		2	6	6		3	3	ohjelmointi
		1	11	4		6	8	kertaus
	yh.	585	184		32	151	280	

MILLI 3B								
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.	
1	1	11	4		2	4		
	2	12	2		4	4		
	3	13	1		3	5		
	4	13	2		3	6		
	5	11	3		3	4		
	6	11	3		5	5		
	7	10	3	1	5	6		
	8	11	3		3	4		
	9	11	4		5	5		
	10	9	3	1	3	5		
	11	12	2		5	5	katsastus projekti	
	12	14	3		3	4		
	13	12	2		4	6		
	14	10	4	1	2	4		
2	15	12	3		3	4		
	16	12	2		3	4		
	17	9	4	1	3	6		
	18	12	2		2	5		
	19	13	2		3	5		
	20	11	3	1	3	5		
	21	12	2		3	4	katsastus projekti	
	22	14	5		7	6		
	23	14	1		4	6		
	24	12	2		2	5		
	25	11	1		2	5		
	26	11	2	1	4	7		
	27	11	1		3	5		
	28	11	1		2	4		
3	29	10	2	1	1	6		
	30	14	1		2	4	katsastus projekti	
	31	13	2		2	5		
	32	13	1		4	5		
	33	12	1		5	5		
	34	11	2	1	4	5		
	35	11	1		4	5		
	36	11	3		3	5		
	37	10	3	1	4	6		
	38	12	2		3	5	katsastus projekti	
	39	11	2		4	6		
	40	11	2		4	5		
	41	11	2		4	5		
	42	11	2		3	4		
4	43	10	3	1	2	5		
	44	9	2		4	4		
	45	9	3	1	4	5		
	46	10	2		3	4		
	47	9	4	1	4	6		
	48	8	2		3	4	katsastus projekti	
	49	7	7		7			
	OHJ	1	6	6		3	4	ohjelmointi
		2	7	7		3	3	ohjelmointi
		1	11	3		5	9	kertaus
	yh.	602	165		43	173	261	



## Liite 3 jatkuu

MILLI 4A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	12	1		6	6	
	2	10	2		4	5	
	3	10	3		4	4	
	4	11	3		5	5	
	5	11	2		4	4	
	6	11	4		4	5	
	7	10	3	1	4	4	
	8	11	4		4	4	
	9	11	2		4	4	
	10	11	2		3	3	
	11	9	2	1	4	5	
	12	10	0		3	3	katsastus projekti
2	13	12	2	5	4	5	
	14	11	3		3	5	
	15	11	1		4	4	
	16	10	3	1	4	4	
	17	12	3		3	3	
	18	12	2		4	4	
	19	11	4	1	3	4	
	20	11	3		4	5	
	21	11	2		4	5	
	22	10	3	1	3	5	
	23	11	1		2	5	
	24	9	4	1	3	4	
25	10	2		3	3	katsastus projekti	
3	26	12	2	7	3	4	
	27	11	2		4	4	
	28	11	3		2	5	
	29	10	4		4	4	
	30	10	4	1	3	5	
	31	11	2		2	3	
	32	11	2		2	5	
	33	11	1		3	4	
	34	10	4	1	5	5	
	35	11	2		3	5	katsastus projekti
4	36	12	1		4	6	
	37	12	0		3	6	
	38	12	2		3	5	
	39	11	1		2	4	
	40	11	3		2	6	
	41	10	2	1	4	7	
	42	11	4		2	3	
	43	12	3		2	5	
	44	11	4	1	5	6	
	45	12	2		3	6	katsastus projekti
OHJ	1	7	7		2	3	ohjelmointi
	2	6	6		3	3	ohjelmointi
	1	12	3		9	9	kertaus
	2	12	3		6	11	kertaus
yht.	553	153	35	174	232		

MILLI 4B							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	12	1		1	5	
	2	11	2		5	6	
	3	11	2		4	4	
	4	11	1		5	5	
	5	10	2	1	3	5	
	6	13	2		3	4	
	7	11	2		3	3	
	8	10	3		4	4	
	9	12	3	1	4	5	
	10	9	1		3	4	katsastus projekti
2	11	11	2	6	2	6	
	12	11	3		4	5	
	13	11	3		3	5	
	14	9	3	1	4	7	
	15	13	2		3	5	
	16	12	3		3	5	
	17	11	4	1	2	5	
	18	10	3		3	4	katsastus projekti
3	19	12	2	6	5	6	
	20	11	1		5	5	
	21	12	1		3	5	
	22	12	1		4	6	
	23	12	2		1	3	
	24	12	3	1	3	5	
	25	11	1		2	5	
	26	10	2		2	5	
	27	10	2	1	2	5	
	28	11	1	7	2	3	katsastus projekti
4	29	13	2	7	5	5	
	30	11	3		2	4	
	31	11	2		3	4	
	32	11	3	1	3	5	
	33	13	1		3	4	
	34	12	2		4	4	
	35	11	3		1	4	
	36	11	3	1	1	5	
	37	11	1		2	5	katsastus projekti
	38	7	7	7			
5	38	13	2		2	5	
	39	11	2		3	4	
	40	12	2		3	3	
	41	10	2		3	4	
	42	10	3	1	4	5	
	43	11	2		4	5	
	44	10	3		3	4	
	45	10	2		4	5	
	46	9	3		2	3	
	47	9	3	1	3	6	
48	11	2		4	4	katsastus projekti	
OHJ	1	6	6		2	3	ohjelmointi
	2	6	6		5	5	ohjelmointi
	1	11	0		7	8	kertaus
	2	11	1		8	9	kertaus
yht.	597	149	41	169	248		

## Liite 3 jatkuu

## Kymppi-oppimateriaalit

KYMPPI 1 syksy							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	5	0		1	8	
	2	7	0		2	6	
	3	6	0		1	6	
	4	9	2		2	6	
	5	8	1		3	5	
	6	7	0		2	7	
	7	8	0		4	5	
	8	8	4		2	3	7
9	9	3			3	6	
10	7	1			2	5	
11	7	1			2	5	
12	8	2			2	6	
13	6	1			3	5	
14	7	1			1	6	
15	8	2			2	6	
16	8	2			2	4	
17	9	1			2	6	
18	8	2			3	6	
19	8	0			3	4	
20	8	5		2	4	7	Testataan ja toimitaan
21	9	0			3	6	
22	7	0			2	5	
23	6	2			3	5	
24	9	0			3	6	
25	7	2			2	5	
26	7	1			4	7	
27	9	1			4	6	
28	8	1			3	5	
29	7	0			2	6	
30	8	0			2	5	
31	9	0			2	5	
32	7	1			2	6	
33	8	5		2	3	6	Testataan ja toimitaan
34	8	0			4	4	
35	8	2			1	6	
36	8	1			3	5	
37	7	0			3	4	
38	7	1			3	4	
39	7	1			2	5	
40	8	1			2	4	
41	7	0			2	5	
42	6	1			4	5	
43	6	2			3	5	
44	9	4		2	4	7	Testataan ja toimitaan
45	7	7			7	7	vikat teht.
OHJ	6	6					
47	9	0			1	6	kerlaus
48	7	3			3	4	kerlaus
49	7	0			4	6	kerlaus
yht.	369	70		8	128	266	

KYMPPI 1 kevät							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	10	1		2	5	
	2	9	0		3	7	
	3	8	2		3	4	
	4	10	1		2	5	
	5	10	1		2	5	
	6	8	1		3	5	
	7	8	1		3	5	
	8	11	1		2	4	
	9	9	1		4	6	
	10	8	1		2	4	
	11	8	3		2	4	8
12	8	0		4	6		
13	8	1		1	4		
14	8	2		2	5		
15	7	1		1	4		
16	8	1		2	5		
17	8	2		2	7		
18	7	2		3	4		
19	9	4		2	2	6	Testataan ja toimitaan
20	8	2		3	5		
21	6	1		2	4		
22	7	1		3	5		
23	8	2		1	4		
24	10	2		1	5		
25	8	1		2	5		
26	8	2		3	5		
27	8	3		2	3	5	Testataan ja toimitaan
28	8	0		5	7		
29	8	0		5	6		
30	8	1		1	4		
31	7	1		2	4		
32	8	1		2	4		
33	9	4			1	4	
34	9	5			1	4	
35	8	5		2	4	6	Testataan ja toimitaan
36	8	1		2	4		
37	9	0		3	7		
38	8	1		2	5		
39	9	0		4	5		
40	10	2		4	8		
41	8	3		2	6		
42	6	1		1	4		
43	7	1		2	4		
44	7	1		2	5		
45	10	1		1	4		
46	6	1		3	5		
47	8	4		2	4	5	Testataan ja toimitaan
OHJ	48	5	5	6	6	6	vikat teht.
49	5	5		7	7	7	vikat teht.
KERT	50	8	1	3	6	6	
51	8	2		2	4	4	
52	7	2		4	6	6	
yht.	419	88		10	141	267	

## Liite 3 jatkuu

KYMPPI 2 syksy							
luku	kpI	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	9	2		5	5	
	2	9	1		3	4	
	3	9	1		2	5	
	4	15	1		2	4	
	5	8	3		4	6	
	6	9	1		4	5	
	7	9	1		4	6	
	8	8	1		3	4	
	9	8	1		1	4	
	10	11	1		4	4	
	11	7	1		3	4	
	12	7	2		3	4	
	13	11	1		3	4	
	14	10	5	2	4	5	testataan ja toimitaan
15	6	1		2	5		
16	7	1		3	5		
17	11	2		3	6		
18	7	1		2	5		
19	7	1		2	5		
20	8	1		2	4		
21	12	1		2	4		
22	7	1		2	5		
23	12	1		4	6		
24	10	1		3	4		
25	7	1		2	5		
26	15	1		3	6		
27	13	1		4	7		
28	10	1		2	6		
29	7	4	2	3	5	testataan ja toimitaan	
30	7	0		3	5		
31	7	0		3	6		
32	13	0		4	5		
33	12	1		4	5		
34	8	0		2	5		
35	10	6		1	4		
36	10	6		4	4		
37	8	4	2	3	4	testataan ja toimitaan	
38	7	4		1	5		
39	8	6		5	5		
40	7	5		3	4		
41	7	5		3	5		
42	7	5		4	5		
43	6	2		2	4		
44	8	3		2	4		
45	9	5	2	3	5	testataan ja toimitaan	
46	7	7		6	6	vikat teht.	
47	7	7		5	5	vikat teht.	
48	8	1		5	6		
49	13	1		3	4		
50	13	1		3	4		
51	11	1		5	6		
yh.	462	111	8	158	248		

KYMPPI 2 kevät							
luku	kpI	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	8	2		5	5	
	2	8	2		5	5	
	3	8	2		5	5	
	4	7	2		4	5	
	5	9	2		5	5	
	6	7	2		4	4	
	7	7	2		4	5	
	8	10	1		4	4	
	9	7	1		5	5	
	10	12	2		4	5	
	11	8	1		3	4	
	12	7	3		4	5	
	13	17	3		4	4	
	14	13	8	1	4	4	testataan ja toimitaan
15	11	1		4	8		
16	7	3		3	4		
17	7	2		3	5		
18	7	4		3	4		
19	11	1		2	6		
20	11	2		3	5		
21	7	1		2	4		
22	11	2		3	4		
23	11	2		3	4		
24	11	1		5	5		
25	8	2		3	4		
26	8	2		2	5		
27	11	7	1	4	4	testataan ja toimitaan	
28	9	1		4	5		
29	7	1		3	5		
30	7	1		3	4		
31	8	2		3	5		
32	12	1		3	5		
33	8	2		2	4		
34	12	2		3	5		
35	11	2		3	5		
36	12	2		4	4		
37	10	2		2	4		
38	7	2		1	4		
39	8	4	2	4	4	testataan ja toimitaan	
40	8	2		2	5		
41	8	2		1	5		
42	8	2		1	4		
43	6	1		1	5		
44	7	1		3	5		
45	10	5		3	5		
46	9	3		3	4		
47	10	4	2	4	4	testataan ja toimitaan	
48	7	7		5	5	vikat teht	
49	6	6		4	4	vikat teht	
50	8	1		3	5		
51	9	2		5	7		
52	9	1		5	7		
yh.	465	122	6	177	246		

KYMPPI 3 syksy							
luku	kpI	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	9	2		2	2	
	2	13	2		3	4	
	3	11	1		2	3	
	4	11	1		1	2	
	5	9	3		3	5	
	6	9	2		1	3	
	7	10	1		2	5	
	8	10	2		3	4	
	9	11	2		4	4	
	10	8	2		2	5	
	11	11	1		2	4	
	12	11	2		3	4	
	13	7	1		5	5	
	14	10	6	1	5	5	testataan ja toimitaan
15	8	2		2	4		
16	10	2		2	4		
17	10	2		1	6		
18	10	1		2	3		
19	13	1		3	4		
20	8	1		2	5		
21	10	2		4	7		
22	8	1		2	5		
23	12	1		4	6		
24	8	1		2	5		
25	13	1		2	5		
26	8	2		2	4		
27	10	1		5	7		
28	9	5	2	1	4	testataan ja toimitaan	
29	6	1		2	3		
30	8	1		2	3		
31	9	1		3	5		
32	10	1		2	3		
33	9	1		4	4		
34	10	2		3	4		
35	13	1		4	4		
36	6	2		4	4		
37	12	2		3	5		
38	9	1		4	6		
39	7	1		3	4		
40	8	6	4	4	4	testataan ja toimitaan	
41	7	2		2	4		
42	7	4		4	5		
43	10	1		2	4		
44	7	4		2	4		
45	8	5		1	3		
46	8	2		3	4		
47	8	6		4	5		
48	8	5	2	5	5	testataan ja toimitaan	
49	6	6		4	4	vikat teht	
50	6	6		3	3	vikat teht	
51	8	3		5	5		
52	7	2		5	6		
yh.	474	118	9	149	225		

KYMPPI 3 kevät							
luku	kpI	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	9	1		3	3	
	2	8	1		1	4	
	3	14	0		2	4	
	4	8	1		2	4	
	5	8	0		3	4	
	6	12	1		2	5	
	7	12	1		3	5	
	8	8	4	2	4	4	testataan ja toimitaan
	9	8	1		4	5	
	10	8	1		1	4	
	11	10	1		6	6	
	12	8	2		4	5	
	13	8	2		2	5	
	14	10	1		2	5	
15	11	5	1	4	5	testataan ja toimitaan	
16	8	3		3	5		
17	8	2		4	6		
18	9	1		4	5		
19	7	2		4	4		
20	6	1		4	4		
21	8	3		3	4		
22	7	2		4	4		
23	7	2		3	4		
24	10	2		4	4		
25	7	2		4	4		
26	8	2		7	7		
27	9	4	1	4	5	testataan ja toimitaan	
28	11	1		1	3		
29	11	3		3	3		
30	9	1		2	4		
31	9	1		4	5		
32	9	1		3	6		
33	7	2		3	3		
34	9	2		4	4		
35	9	2		3	4		
36	7	1		5	5		
37	8	2		5	5		
38	8	6	1	4	5	testataan ja toimitaan	
39	7	1		5	6		
40	7	0		3	5		
41	7	1		3	6		
42	8	1		4	6		
43	7	1		2	5		
44	11	1		2	4		
45	11	1		2	4		
46	15	1		3	3		
47	11	3		4	4		
48	10	6	4	3	5	testataan ja toimitaan	
49	9	5		4	4	vikat teht	
50	9	7		3	3	vikat teht	
51	8	2		7	7		
52	8	2		5	5		
yh.	461	102	9	176	238		

## Liite 3 jatkuu

KYMPPI 4 syksy							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	10	2		3	4	
	2	9	1		3	4	
	3	11	3		4	4	
	4	10	2		3	3	
	5	11	1		3	4	
	6	10	2		3	5	
	7	13	3		4	6	
	8	10	3		5	5	
	9	6	3		4	4	
	10	7	3		4	4	
	11	9	3		5	5	
	12	10	3		3	4	
	13	11	5	1	4	4	testataan ja toimitaan
2	14	9	2		2	5	
	15	10	2		3	3	
	16	10	3		3	4	
	17	10	3		5	5	
	18	12	2		3	5	
	19	12	2		5	5	
	20	11	2		4	5	
	21	7	1		2	4	
	22	9	3		4	4	
	23	9	2		4	5	
	24	10	1		4	5	
	25	10	3		4	4	
	26	8	2		4	4	
	27	10	6	3	4	4	testataan ja toimitaan
3	28	9	3		4	7	
	29	9	5		4	5	
	30	8	1		3	5	
	31	9	1		2	5	
	32	8	2		3	3	
	33	8	2		5	5	
	34	10	1		3	5	
	35	9	0		2	4	
	36	8	5		5	5	
	37	9	7		4	4	
	38	7	6		4	5	
	39	10	5	1	4	4	testataan ja toimitaan
	40	8	1		3	5	
	41	7	1		2	4	
4	42	7	4		4	4	
	43	16	1		3	5	
	44	7	1		3	4	
	45	7	2		4	5	
	46	7	1		4	4	
	47	12	1		3	5	
	48	7	3		5	5	
	49	9	5	2	3	4	testataan ja toimitaan
OHJ	50	8	5		5	5	vikat teht
	51	8	6		6	6	vikat teht
KERT	52	13	2		4	4	
	53	10	2		2	4	
	54	8	4		4	5	
yhjt.	502	145	7	197	244		

KYMPPI 4 kevät							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	12	1		3	4	
	2	15	2		3	5	
	3	11	0		2	4	
	4	12	0		3	3	
	5	12	2		3	4	
	6	8	1		3	4	
	7	12	2		2	5	
	8	12	1		4	4	
	9	8	1		2	3	
	10	15	1		2	4	
	11	11	5	1	4	5	testataan ja toimitaan
	12	13	1		2	4	
	2	13	9	2		4	4
14		7	3		3	4	
15		7	3		4	5	
16		7	2		2	4	
17		8	2		3	5	
18		9	3		4	4	
19		9	3		4	4	
20		7	3		3	5	
21		13	3		4	5	
22		10	5	1	4	4	testataan ja toimitaan
23		9	2		2	4	
3	24	15	2		4	5	
	25	13	2		2	5	
	26	11	2		3	4	
	27	12	2		4	5	
	28	6	2		4	5	
	29	13	2		3	4	
	30	11	2		5	5	
	31	11	2		2	4	
	32	11	2		3	4	
	33	14	2		3	4	
	34	11	2		4	4	
	35	10	3	1	3	4	testataan ja toimitaan
	36	10	6		4	4	
	37	16	2		4	6	
4	38	10	1		4	5	
	39	8	2		5	5	
	40	7	2		5	5	
	41	8	3		3	5	
	42	8	2		3	4	
	43	11	2		3	5	
	44	14	2		3	6	
	45	10	2		3	3	
	46	14	2		2	3	
	47	12	3		3	4	
	48	16	2		4	5	
OHJ	49	10	4	2	3	4	testataan ja toimitaan
	50	7	5		5	5	vikat teht
KERT	51	7	5		4	4	vikat teht
	52	12	2		6	7	
	53	12	2		6	7	
	54	15	2		5	6	
yhjt.	581	124	5	185	243		

## Liite 3 jatkuu

## Tuhattaituri-oppimateriaalit

TUHATTAITURI 1A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikk	huom.
1	1	7	1		1	6	
	2	6	1		3	6	
	3	10	2		2	6	
	4	10	2		4	7	
	5	10	2		5	8	
	6	6	1		4	7	
	7	7	0		3	7	
	8	8	0		4	7	
	9	11	0		3	6	tähtipyöski/talorasti
	10	9	1		3	7	
2	11	12	1		3	8	
	12	13	1		5	10	
	13	10	3		4	8	
	14	7	4	2	3	8	toimintatunti/talutunti
	15	11	1		4	10	
	16	9	3		5	10	
	17	8	2		2	9	
	18	9	1		5	8	
	19	9	1		7	7	tähtipyöski/talorasti
	20	12	2		2	10	
3	21	9	2		4	6	
	22	7	1		4	6	
	23	13	2		4	8	
	24	8	2		3	6	
	25	8	2	1	4	7	toimintatunti
	26	11	2		4	7	
	27	10	2		5	7	tähtipyöski/talorasti
	28	13	2		4	8	
	29	8	5		5	7	
	30	10	5		5	9	
4	31	8	2	1	6	7	toimintatunti
	32	11	2		4	9	
	33	7	7		6	6	
	34	9	5		5	8	
	35	12	3		5	6	tähtipyöski/talorasti
	36	9	3		4	8	
	37	7	4		4	7	
	38	8	3	1	5	7	toimintatunti
	39	8	2		5	7	
	40	8	2		5	6	
5	41	8	1		6	7	
	42	9	3		3	7	
	43	11	0		4	9	tähtipyöski/talorasti
	44	12	5	1			Tutkimustehtävät
	yht.	408	96	6	176	320	

TUHATTAITURI 1B							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikk	huom.
1	1	11	3		4	9	
	2	11	2		4	10	
	3	9	1		2	7	
	4	8	1		3	8	
	5	7	3		5	8	toimintatunti
	6	10	4		5	10	
	7	10	1		6	7	
	8	9	2		5	9	
	9	13	1		4	8	tähtipyöski/talorasti
	10	9	3		5	9	
2	11	10	2		4	8	
	12	9	2		5	9	
	13	9	3		7	10	
	14	9	2		5	9	
	15	9	3		7	11	
	16	8	2		6	9	
	17	9	2		6	9	
	18	8	4	2	4	10	toimintatunti
	19	11	3		6	10	tähtipyöski/talorasti
	20	9	4		8	9	
3	21	9	5		3	6	
	22	9	2		3	9	
	23	6	4	2	6	7	toimintatunti
	24	7	2		4	6	
	25	6	1		2	7	
	26	7	7		7	7	ohjelmointi
	27	9	5		3	8	tähtipyöski/talorasti
	28	10	2		5	8	
	29	10	3		4	7	
	30	10	3		7	9	
4	31	10	3		5	8	
	32	9	3		4	8	
	33	11	4		3	8	
	34	8	4	1	2	10	toimintatunti
	35	8	6		6	9	
	36	9	2		4	8	
	37	11	3		1	7	tähtipyöski/talorasti
	38	10	2		5	9	
	39	10	2		4	6	
	40	10	2		5	6	
5	41	7	2		4	6	
	42	8	2		4	6	
	43	7	5	2	3	6	toimintatunti
	44	9	1		4	6	
	45	9	1		2	6	
	46	10	2		6	8	
	47	11	2		5	9	tähtipyöski/talorasti
	48	16	2		3		Tutkimustehtävät
	yht.	444	130	3	212	379	

TUHATTAITURI 2A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikk	huom.
1	1	10	3		4	6	
	2	10	2		3	7	
	3	9	1		5	7	
	4	9	4	1	4	8	toimintatunti
	5	10	1		4	5	
	6	11	1		6	7	
	7	9	1		3	6	
	8	9	1		4	6	
	9	9	2		4	8	
	10	18	5		5	7	pyöski/rasti/hädet
2	11	10	2		5	8	
	12	8	2		4	6	
	13	10	1		4	7	
	14	8	5	2	4	7	toimintatunti
	15	11	5		5	7	
	16	9	1		3	6	
	17	10	4		2	6	
	18	11	2		4	7	
	19	17	9		3	7	pyöski/rasti/hädet
	20	9	1		3	9	
3	21	13	1		5	9	
	22	12	1		5	10	
	23	8	5	2	3	7	toimintatunti
	24	13	3		4	11	
	25	9	2		5	7	
	26	9	2		3	7	
	27	10	3		3	6	
	28	8	3		5	7	
	29	16	5		6	9	pyöski/rasti/hädet
	30	9	3	2	3	8	toimintatunti
4	31	9	2		4	5	
	32	8	4		4	5	
	33	8	4	2	4	6	toimintatunti
	34	8	2		3	5	
	35	13	3		2	6	
	36	19	8		2	7	
	37	10	4	1	2	8	toimintatunti
	38	9	5		4	9	
	39	9	2		1	7	
	40	8	3		3	7	
5	41	7	2		3	7	
	42	10	9		3	7	
	43	7	7		6	6	ohjelmointia
	44	14	8		3	9	pyöski/rasti/hädet
	45	12	9	2			Tutkimustehtävät
	yht.	465	153	12	165	312	

TUHATTAITURI 2B							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikk	huom.
1	1	10	2		2	6	
	2	11	3		3	8	
	3	9	1		4	7	
	4	8	0		3	7	
	5	12	3		4	6	
	6	7	3		3	7	toimintatunti
	7	9	5		6	7	ohjelmointia
	8	11	1		4	6	
	9	14	3		2	7	pyöski/rasti/hädet
	10	13	2		4	8	
2	11	11	1		3	9	
	12	8	4	2	4	8	toimintatunti
	13	12	2		3	8	
	14	12	3		5	8	
	15	11	3		4	8	
	16	9	1		4	8	
	17	11	4		3	8	
	18	18	7		3	8	pyöski/rasti/hädet
	19	9	1		5	8	
	20	9	1		6	7	
3	21	9	6		4	6	
	22	10	2		5	8	
	23	7	4	2	4	5	toimintatunti
	24	8	2		5	6	
	25	8	2		5	6	
	26	9	2		5	6	
	27	9	1		5	5	
	28	19	9		5	5	pyöski/rasti/hädet
	29	9	1		3	6	
	30	9	4		2	6	
4	31	9	2		2	7	
	32	9	2		3	6	
	33	6	4	2	3	5	toimintatunti
	34	8	1		3	6	
	35	10	2		2	6	
	36	10	1		2	6	
	37	10	2		3	6	
	38	16	4		3	5	pyöski/rasti/hädet
	39	8	1		2	6	
	40	11	2		3	6	
5	41	8	3	2	3	6	toimintatunti
	42	9	2		3	6	
	43	7	5	2	4	7	toimintatunti
	44	10	2		4	7	
	45	7	5	2	3	5	toimintatunti
	46	11	3		3	6	
	47	16	7		2	6	pyöski/rasti/hädet
	48	16	5		5		Tutkimustehtävät
	yht.	492	136	19	166	310	

## Liite 3 jatkuu

TUHATTAITURI 3A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	13	4		2	6	
	2	11	3		3	4	
	3	15	2		3	7	
	4	8	3		4	4	
	5	9	3		4	4	
	6	10	4		3	3	
	7	9	4		2	4	
	8	9	2		3	3	
	9	8	2		4	4	
	10	9	2		3	4	
	11	9	4		4	4	
	12	8	3		2	3	
	13	9	5	3	3	3	toimintatunti
	14	32	2		3	5	pyöski/rasiti/händet
2	15	14	3		2	5	
	16	17	2		3	7	
	17	17	2		2	6	
	18	15	2		2	6	
	19	14	2		2	5	
	20	10	2		2	5	
	21	14	3		3	5	
	22	14	2		1	5	
	23	15	5		3	6	
	24	11	4		2	5	
	25	12	1		4	6	
	26	7	5	2	2	6	toimintatunti
	27	26	8		1	4	pyöski/rasiti/händet
	28	9	2		4	5	
3	29	8	2		1	4	
	30	10	4		2	4	
	31	8	2		3	4	
	32	8	2		3	4	
	33	8	2		3	4	
	34	7	4	2	3	5	toimintatunti
	35	20	6		4	5	pyöski/rasiti/händet
	36	10	2		2	6	
	37	11	1		2	6	
	38	10	2		1	5	
	39	10	3		2	5	
	40	10	2		4	6	
	41	9	2		2	6	
	42	7	4	2	3	5	toimintatunti
4	43	25	3		2	6	pyöski/rasiti/händet
	44	10	4		3	3	
	45	11	3		3	6	
	46	9	2		4	4	
5	47	8	1		4	6	
	48	6	6	4			Projektitehtävät (2 ohj)
yh.	559	143	13	125	228		

TUHATTAITURI 3B							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	10	5		1	5	
	2	11	2		3	5	
	3	12	3		2	4	
	4	9	2		2	5	
	5	11	3		1	2	
	6	11	3		3	3	
	7	17	3		2	3	pyöski/rasiti/händet
	8	13	4		3	4	
	9	10	2		1	3	
	10	10	2		3	4	
	11	11	2		3	4	
	12	6	3		2	3	toimintatunti
	13	26	4		4	5	pyöski/rasiti/händet
	14	12	5		2	3	
2	15	12	1		4	5	
	16	11	2		3	4	
	17	13	4		1	3	
	18	13	1		2	3	
	19	12	4		2	4	
	20	13	3		2	4	
	21	12	2		3	5	
	22	7	5	2	1	4	toimintatunti
	23	26	3		4	4	pyöski/rasiti/händet
	24	10	3		2	5	
	25	11	2		2	4	
	26	10	3		3	4	
	27	10	3		2	6	
	3	28	9	6	3	3	6
29		10	4		3	5	
30		11	3		4	4	
31		9	1		3	4	
32		28	4		5	6	pyöski/rasiti/händet
33		8	2		2	5	
34		9	4		2	5	
35		9	3		5	5	
36		9	2		5	5	
37		9	2		4	5	
38		9	3		4	4	
39		9	6	3	4	4	toimintatunti
40		8	0		2	5	
41		12	1		4	4	
42	10	1		3	4		
4	43	10	3		2	4	
	44	23	2		3	3	pyöski/rasiti/händet
	45	15	5		1	3	
	46	13	1		2	4	
KERT	47	8	2		3	4	
	48	12	2		3	3	
	49	5	1		3		Projektitehtävät (1 ohj)
yh.	574	137	13	131	200		

## Oivaltaja-oppimateriaalit

OIVALTAJA 1A							
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	9	3		5	7	
	2	10	5		6	6	
	3	11	2	1	1	8	
	4	11	2	1	1	3	
	5	10	1		1	3	
	6	11	7		3	3	ohj
	7	13	4		1	6	
2	8	13	2		0	7	
	9	13	3		0	7	
	10	13	2		1	7	
	11	10	1		1	3	
	12	13	3	1	1	7	
	13	13	2	1	1	7	
	14	9	2	1	1	3	
3	15	8	1		1	1	kertaus
	16	9	2	2	4	4	
	17	12	3	1	2	3	
	18	10	10		3	3	
	19	11	2	1	3	3	
	20	12	1		2	3	
	21	11	4	1	3	3	
4	22	11	3		2	3	
	23	10	2	1	2	3	
	24	12	3	1	2	3	
	25	11	3	1	2	3	
	26	10	3	1	0	3	
	27	8	1		1	1	kertaus
	28	14	2		1	3	
5	29	11	4		1	3	
	30	14	2		1	3	
	31	11	3		1	3	
	32	14	2		1	3	
	33	10	2		1	3	
	34	12	2		2	3	
	35	9	2		1	1	kertaus
6	36	9	2	2	3	3	
	37	9	2	1	1	3	
	38	11	1		1	3	
	39	11	4	1	0	3	
	40	12	3	1	0	3	
	41	11	2		2	3	
	42	12	2		1	3	
7	43	12	3		1	3	
	44	9	2		1	1	kertaus
	45	6	6	4			
	yh.	485	116	19	69	165	

OIVALTAJA 1B								
luku	kpl	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.	
1	1	13	2	1	1	3		
	2	11	2		1	3		
	3	10	2		1	3		
	4	13	3	1	1	3		
	5	13	1		1	3		
	6	13	3		1	3		
	7	11	7		1	3		
	8	13	4	1	1	3		
	9	12	2		1	3		
	10	10	2		1	1	kertaus	
	2	11	11	2	1	2	3	
		12	12	2	1	2	3	
		13	10	2	1	1	3	
		14	10	3		1	3	
15		11	2		1	3		
16		11	4	1	2	3		
17		10	1		1	1	kertaus	
3	18	11	4		3	3		
	19	9	2	1	0	3		
	20	11	3	1	1	3		
	21	11	5	1	2	3		
	22	10	3		0	3		
	23	11	3	1	1	3		
	24	11	2		1	3		
4	25	10	4	1	1	3		
	26	10	4	1	2	3		
	27	9	2		2	3		
	28	12	2		1	1	kertaus	
	29	11	4	1	1	3		
	30	9	3		0	3		
	31	10	2	1	1	3		
5	32	10	3		1	3		
	33	10	4	1	0	3		
	34	10	1		1	3		
	35	9	2		2	3		
	36	10	1		1	1	kertaus	
	37	10	5		2	3		
	38	10	3	2	0	3		
6	39	11	4	1	2	3		
	40	10	7	1	3	3		
	41	11	10		2	3		
	42	10	6		2	3		
	43	10	7		3	3		
	44	9	7		1	1	kertaus	
	45	6	6	4				
yh.	469	147	20	58	122			

## Liite 3 jatkuu

OIVALTAJA 3A							
luku	kgf	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	13	5		1	3	
	2	12	4		2	3	
	3	11	2		1	3	
2	4	13	2		1	3	
	5	12	1		1	3	
	6	14	3		1	3	
	7	14	3		1	3	
	8	13	5	1	1	1	
	9	12	4		1	2	
	10	12	2		1	3	
	11	12	1		1	1	
	12	8	1		0	0	kertaus
	13	13	3		1	3	
	14	15	1		1	4	
3	15	14	2		2	3	
	16	13	2		2	3	
	17	13	2		2	3	
	18	13	3		1	1	
	19	13	1	1	2	3	
	20	13	3		2	3	
	21	13	2		1	3	
	22	14	3		1	1	
	23	8	0		0	0	kertaus
	24	7	1		2	3	
4	25	9	9		1	2	
	26	7	7		2	2	ohj
	27	6	6	2	2	2	
	28	2	1		0	0	kertaus
	29	12	6		1	3	
	30	10	4		0	3	
5	31	11	4		1	3	
	32	11	5		1	1	
	33	13	6		3	3	
	34	12	4	1	1	3	
	35	11	4		1	3	
	36	12	5	1	1	2	
	37	14	4		1	1	
	38	6	1		0	0	kertaus
	39	14	4		1	2	
	40	11	1		2	3	
6	41	11	2		1	2	
	42	14	1		1	3	
	43	14	2		1	2	
	44	13	2		1	2	
	45	15	2		1	1	
	46	13	3		1	3	
	47	13	3		1	2	
	48	16	2		1	3	
	49	13	3		1	1	
	50	9	0		0	0	kertaus
yht.	587	147	6	56	110		

OIVALTAJA 3B							
luku	kgf	tehtävät	ohj. tukevat	toim.	oppaan ohj	oppaan kaikki	huom.
1	1	14	3		1	3	
	2	11	2		2	3	
	3	11	2		2	3	
	4	11	3		2	3	
	5	10	3		3	3	
	6	10	4		1	1	
	7	11	4		2	3	
	8	12	4		2	3	
	9	11	3		1	2	
	10	11	3		3	3	
	11	11	3		1	1	
	12	8	1		0	0	kertaus
	13	12	1		2	3	
2	14	12	2		1	3	
	15	13	3		1	3	
	16	12	2		1	3	
	17	15	3	1	1	1	
	18	11	3		1	3	
	19	12	3		1	3	
	20	12	3		1	3	
	21	14	2		1	1	
	22	7	1		0	0	kertaus
	23	9	2		2	2	
3	24	10	10		2	2	
	25	10	10		2	2	ohj
	26	6	6	3	2	2	
	27	2	1		0	0	kertaus
	28	15	2		1	3	
	29	13	3		2	3	
	30	13	3		1	3	
4	31	14	3		1	3	
	32	16	2		1	1	
	33	14	4		0	3	
	34	13	3		0	3	
	35	12	4		1	3	
	36	14	4	1	1	1	
	37	9	1		0	0	kertaus
	38	13	4		2	3	
	39	13	3		3	3	
	40	13	1		3	3	
5	41	14	3		3	3	
	42	13	4		3	3	
	43	13	4		1	1	
	44	11	2	1	0	3	
	45	14	4		2	3	
	46	14	4		1	1	
	47	7	0		0	0	kertaus
	yht.	546	145	8	65	106	

#### **Liite 4. Haastattelurunko**

##### **Taustatiedot:**

Kuinka kauan olet toiminut opena ja paljonko olet opettanut ohjelmointia?

Kuinka paljon olet perehtynyt ohjelmointiin? Kuinka paljon siinä toiminnallisuuteen?

Oletko käynyt ohjelmointiin liittyvissä koulutuksissa?

##### **Ohjelmoinnin opetus:**

Mitä ohjelmoinnin opetukseen mielestäsi sisältyy?

Miten opetat ohjelmointia alakoulussa 1-4.lk?

Millaisilla työtavoilla ja -välineillä tai sovelluksilla olet opettanut ohjelmointia?

Mitkä olet kokenut hyödyllisiksi ja miksi?

Miksi ohjelmointia on tärkeä opettaa oppilaille?

Millaisia resursseja ohjelmoinnin opettaminen vaatii?

##### **Toiminnallisuus ohjelmoinnissa:**

Mitä mielestäsi on toiminnallisuus ohjelmoinnin opettamisessa?

Mitä mieltä olet matematiikan oppimateriaalien ohjelmoinnin opetuksesta?

Miten matematiikan oppimateriaalit tukevat toiminnallisen ohjelmoinnin opettamista? Käytätkö niitä? Onko ne tarpeeksi ohjelmoinnin opetusta?

Käytätkö muita materiaaleja toiminnallisen ohjelmoinnin toteuttamisessa?



**Algoritmista ajattelua tukevat ja kehittävät tehtävät:**

Minkälaiset tehtävät ja harjoitukset sinun mielestäsi tukevat

A) ongelmanratkaisuntaitoja ja loogista päättelyä

B) suunta- ja sijaintikäsitettä

C) toimintaohjeiden noudattamista

D) kielentämistä

Miten algoritmisen ajattelun kehittymistä voi mielestäsi seurata?

## Liite 5. Saatekirje

Hei!

Olemme Kokkolan yliopistokeskus Chydeniuksen luokanopettajaopiskelijat Jasmin Vilmusenaho ja Petri Järvelä. Teemme Pro gradu -tutkielmaa, jossa tarkastelemme matematiikan oppimateriaalien sisältämää ohjelmoinnin opettamista ja kuinka toiminallisuus on otettu siinä huomioon. Oppimateriaaliaineiston lisäksi haluaisimme saada ohjelmointia opettavilta luokanopettajilta kokemuksia ohjelmoinnin opettamisesta alakoulussa.

Luokanopettajien kokemuksia keräisimme haastattelemalla. Haastattelut pyritään toteuttamaan kevään 2022 aikana. Haastattelun kestoksi arvioimme noin tunnin. Haastattelut nauhoitetaan ja litteroidaan analysointia varten. Tutkimuksen valmistuttua haastatteluaineisto hävitetään tietoturvasääntöjen mukaisesti. Haastateltavien henkilötiedot salataan, vain taustatiedot: sukupuoli, opettajakokemus, ohjelmoinnin opetuskokemus ja tutkittavan mahdolliset ohjelmointikoulutukset tulevat tutkielmassa ilmi. Osallistuminen tutkimukseen on tutkittavalle vapaaehtoista ja tutkittava voi perua osallistumisensa milloin tahansa.

Yhteistyöterveisin

Petri Järvelä

[petri.k.jarvela@student.jyu.fi](mailto:petri.k.jarvela@student.jyu.fi)

Jasmin Vilmusenaho

[jalivilm@student.jyu.fi](mailto:jalivilm@student.jyu.fi)



## Liite 6. Tiedote tutkimuksesta

KOKKOLAN  
YLIOPISTOKESKUS  
CHYDENIUS

JYVÄSKYLÄN YLIOPISTO

1.4.2022

### TIEDOTE TUTKIMUKSESTA

#### Tutkimuksen nimi ja rekisterinpitäjä

*Pro gradu -tutkielma: "Ohjelmoinnin opetus 1–4 lk. matematiikan oppimateriaaleissa ja luokanopettajien kokemana – miten toiminnallisuus näkyy ohjelmoinnin opetuksessa?"*

Rekisterinpitäjä: Petri Järvelä ja Jasmin Vilmusenaho

#### Pyyntö osallistua tutkimukseen

Sinua pyydetään mukaan tutkimukseen, jonka tavoitteena on tutkia, miten ohjelmointia opetetaan matematiikan oppimateriaaleissa ja luokanopettajien kokemana sekä miten toiminnallisuus on otettu huomioon ohjelmoinnin opetuksessa. Keskeiset tutkimuskysymykset ovat:

**Miten ja kuinka paljon ohjelmointia opetetaan matematiikan oppimateriaaleissa vuosiluokilla 1–4?**

**Miten toiminnallisuus on otettu huomioon ohjelmoinnin opetuksessa matematiikan oppimateriaaleissa vuosiluokilla 1–4?**

**Minkälaisia kokemuksia luokanopettajilla on ohjelmoinnin opetuksesta?**

Tutkimuksen kohteena on luokanopettajien kokemus ohjelmoinnin opettamisesta alakoulussa. Kerättyä aineistoa tullaan käyttämään Petri Järvelän ja Jasmin Vilmusenahon pro gradu -tutkielmassa ” *Ohjelmoinnin opetus 1–4 lk. matematiikan oppimateriaaleissa ja luokanopettajien kokemana – miten toiminnallisuus näkyy ohjelmoinnin opetuksessa?*”. Sinua pyydetään tutkimukseen, koska sinun näkemyksesi ja kokemuksesi ohjelmoinnin opettamisesta alakoulussa olisi tutkimuksen kannalta arvokasta. Tämä tiedote kuvaa tutkimusta ja siihen osallistumistasi.

Tutkimukseen osallistuminen edellyttää, että Sinulla on **monipuolista kokemusta ohjelmoinnin opettamisesta alakoulussa**. Mukaan pyydetään yhteensä kaksi tutkittavaa.

**Osallistuvista henkilöistä ei kerätä tietoja muista lähteistä.**

## Vapaaehtoisuus

Tähän tutkimukseen osallistuminen on vapaaehtoista. Voit milloin tahansa kieltäytyä osallistumasta tutkimukseen tai keskeyttää osallistumisen. Kieltäytymisestä tai keskeyttämisestä ei aiheudu sinulle kielteisiä seuraamuksia.

## Tutkimuksen kulku

Haastatteluissa tutkitaan haastateltavan näkemyksiä siitä, miten he opettavat ohjelmointia alakoulussa ja miten toiminnallisuus on otettu opetuksessa huomioon. Haastattelut järjestetään keväällä 2022. Jokainen haastattelu kestää arviolta noin yhden tunnin. Haastattelut tallennetaan äänitallenteiksi, jotka tutkija litteroi kirjalliseen muotoon. Tämän jälkeen äänitallenteet tuhoataan. Haastatteluun osallistuminen ei vaadi erityistä valmistautumista.

Haastattelut toteutetaan mielellään lähitapaamisina. Etätapaamisessa käytetään Jyväskylän yliopiston Zoom palvelua tai Teams-sovellusta.

## Tutkimuksesta mahdollisesti aiheutuvat riskit, haitat ja epämukavuudet

Tutkimukseen osallistumisesta ei aiheudu sinulle mitään riskejä, haittoja tai epämukavuuksia.

## Tutkimuksen kustannukset ja rahoitus

Tutkimukseen osallistumisesta ei makseta palkkiota.

## Tutkimustuloksista tiedottaminen ja tutkimustulokset

Tutkimuksesta valmistuu yksi pro gradu -tutkielma, joka julkaistaan Jyväskylän yliopiston julkaisuarkistossa: <https://jyx.jyu.fi/>

## Tutkittavien vakuutusturva

Jyväskylän yliopiston henkilökunta ja toiminta on vakuutettu. Vakuutus sisältää potilasvakuutuksen, toiminnanvastuuvakuutuksen ja vapaaehtoisen tapaturmavakuutuksen.

Tutkimuksissa tutkittavat (koehenkilöt) on vakuutettu tutkimuksen ajan ulkoisen syyn aiheuttamien tapaturmien, vahinkojen ja vammojen varalta. Tapaturmavakuutus on voimassa mittauksissa ja niihin välittömästi liittyvillä matkoilla.

## Lisätietojen antajan yhteystiedot

Lisätietoja tutkimuksesta antaa tutkimuksentekijät: Petri Järvelä ja Jasmin Vilmusenaho, maisteriopiskelijat, kasvatustiede luokanopettaja, Jyväskylän yliopisto, Kokkolan yliopistokeskus Chydenius, [petri.k.jarvela@student.jyu.fi](mailto:petri.k.jarvela@student.jyu.fi), [jalivilm@student.jyu.fi](mailto:jalivilm@student.jyu.fi)



## Liite 7. Tietosuojailmoitus

KOKKOLAN  
YLIOPISTOKESKUS  
CHYDENIUS

## JYVÄSKYLÄN YLIOPISTO

1.4.2022

### TIETOSUOJAILMOITUS

Kuvaus henkilötietojen käsittelystä tieteellisessä tutkimuksessa (tietosuojailmoitus EU (679/2016)

***Tutkimuksessa: ” Ohjelmoinnin opetus 1–4 lk. matematiikan oppimateriaaleissa ja luokanopettajien kokemana – miten toimin-nallisuus näkyy ohjelmoinnin opetuksessa?”***

***käsiteltävät henkilötiedot***

Haastatteluiden avulla kerättyjä tietoja käytetään Petri Järvelän ja Jasmin Vilmusenahon pro gradu -tutkielmassa ” ***Ohjelmoinnin opetus 1–4 lk. matematiikan oppimateriaaleissa ja luokanopettajien kokemana – miten toimin-nallisuus näkyy ohjelmoinnin opetuksessa?”*** aineistona. Tutkielman tavoitteena on tuottaa tietoa siitä, miten ohjelmointia opetetaan matematiikan oppimateriaaleissa ja luokanopettajien kokemana sekä miten toiminnallisuus on otettu huomioon ohjelmoinnin opetuksessa.

Tutkimuksessa Sinusta kerätään seuraavia henkilötietoja: äänitallenne

Tiedote ja tietosuojailmoitus on lähetetty liitetiedostona haastattelupyynnön ja haastattelukutsun mukana.

***Henkilötietojen käsittelyn oikeudellinen peruste tutkimuksessa/arkistoinnissa***

Käsittely on tarpeen tieteellistä tutkimusta varten ja se on oikeasuhtaista, sillä tavoiteltuun yleisen edun mukaiseen tavoitteeseen nähden (tietosuojain 4 §:n 3 kohta). Tutkimustulokset ovat julkisesti saatavilla yliopiston julkaisuarkistosta (JYX).

***Henkilötietojen siirto EU/ETA ulkopuolelle***

Tutkimuksessa tietojasi ei siirretä EU/ETA-alueen ulkopuolelle.

***Henkilötietojen suojaaminen***

Henkilötietojen käsittely tässä tutkimuksessa perustuu asianmukaiseen tutkimussuunnitelmaan ja tutkimuksella on vastuuhenkilö. Tutkimuksessa toimitaan niin, etteivät Sinua koskevat tiedot paljastu ulkopuolisille. Tutkimustulosten (Pro gradu) osalta

pyritään siihen, ettei Sinua voida tunnistaa suoraan taikka välillisesti tutkimustuloksista. Osana Pro gradua voidaan käyttää suoria lainauksia haastattelusta.

Suorat tunnistetiedot poistetaan suojatoimena aineiston litterointivaiheessa eli tutkittavaa koskeva ääni tuhotaan.

#### **Tutkimuksessa käsiteltävien henkilötietojen suojaaminen**

Nauhoitettu aineisto litteroidaan ja tallennetaan tutkijan omalle tietokoneelle. Tutkijat eivät julkaise tutkittavien tietoja muille tutkittaville tai ulkopuolisille. Tutkijat vastaavat aineiston hallinnasta sen elinkaaren aikana ja tietojen hävittämisestä tutkimuksen päätyttyä.

Tutkimuksesta on tehty **erillinen tietosuojan vaikutustenarvio**/tietosuojavastaavaa on kuultu vaikutustenarvioinnista

Kyllä  Ei, koska tutkija on tarkastanut, ettei vaikutustenarviointi ole pakollinen.

#### Henkilötietojen käsittely tutkimuksen päättymisen jälkeen:

Tutkimusrekisteri hävitetään viimeistään pro gradu -tutkielman valmistuttua (arvio 07.2022).

**Rekisterinpitäjät, pro gradu- tutkielman suorittajat ja yhteyshenkilöt:** Petri Järvelä, maisteriopiskelija, petri.k.jarvela@student.jyu.fi. ja Jasmin Vilmusenaho, maisteriopiskelija, jalivilm@student.jyu.fi

**Tutkimuksen ohjaaja:** Yliopistonlehtori Päivi Perkkilä, Kokkolan yliopistokeskus Chydenius

#### **Rekisteröidyn oikeudet**

##### Oikeus saada pääsy tietoihin (tietosuoja-asetuksen 15 artikla)

Sinulla on oikeus saada tieto siitä, käsitelläänkö henkilötietojasi ja mitä henkilötietojasi käsitellään. Voit myös halutessasi pyytää jäljennöksen käsiteltävistä henkilötiedoista.

##### Oikeus tietojen oikaisemiseen (tietosuoja-asetuksen 16 artikla)

Jos käsiteltävissä henkilötiedoissasi on epätarkkuuksia tai virheitä, sinulla on oikeus pyytää niiden oikaisua tai täydennystä.

##### Oikeus tietojen poistamiseen (tietosuoja-asetuksen 17 artikla)

Sinulla on oikeus vaatia henkilötietojesi poistamista tietyissä tapauksissa. Oikeutta tietojen poistamiseen ei kuitenkaan ole, jos tietojen poistaminen estää tai vaikeuttaa suuresti käsittelyn tarkoituksen toteutumista tieteellisessä tutkimuksessa.

##### Oikeus käsittelyn rajoittamiseen (tietosuoja-asetuksen 18 artikla)

Sinulla on oikeus henkilötietojesi käsittelyn rajoittamiseen tietyissä tilanteissa kuten, jos kiistät henkilötietojesi paikkansapitävyyden.

##### Vastustamisoikeus (tietosuoja-asetuksen 21 artikla)

Sinulla on oikeus vastustaa henkilötietojesi käsittelyä, jos käsittely perustuu yleiseen etuun tai oikeutettuun etuun. Tällöin yliopisto ei voi käsitellä henkilötietojasi, paitsi jos se voi osoittaa, että käsittelyyn on olemassa huomattavan tärkeä ja perusteltu syy, joka syrjäyttää oikeutesi.

### Oikeuksista poikkeaminen

Tässä kuvatuista oikeuksista saatetaan tietyissä yksittäistapauksissa poiketa tietosuojasetuksessa ja Suomen tietosuojalaissa säädetyillä perusteilla siltä osin, kuin oikeudet estävät tieteellisen tai historiallisen tutkimustarkoituksen tai tilastollisen tarkoituksen saavuttamisen tai vaikeuttavat sitä suuresti. Tarvetta poiketa oikeuksista arvioidaan aina tapauskohtaisesti.

### Profilointi ja automatisoitu päätöksenteko

Tutkimuksessa henkilötietojasi ei käytetä automaattiseen päätöksentekoon. Tutkimuksessa henkilötietojen käsittelyn tarkoituksena ei ole henkilökohtaisten ominaisuuksiesi arviointi, ts. profilointi vaan henkilötietojasi ja ominaisuuksia arvioidaan laajemman tieteellisen tutkimuksen näkökulmasta.

### **Rekisteröidyn oikeuksien toteuttaminen**

Jos sinulla on kysyttävää rekisteröidyn oikeuksista, voit olla yhteydessä yliopiston tietosuojavastaavaan. Kaikki oikeuksien toteuttamista koskevat pyynnöt toimitetaan Jyväskylän yliopiston kirjaamoon. Kirjaamo ja arkisto, PL 35 (C), 40014 Jyväskylän yliopisto, puh. 040 805 3472, e-mail: kirjaamo(at)jyu.fi. Käyntiosoite: Seminaarinkatu 15 C-rakennus (Yliopiston päärakennus, 1. krs), huone C 140.

Sinulla on oikeus tehdä valitus erityisesti vakinaisen asuin- tai työpaikkasi sijainnin mukaiselle valvontaviranomaiselle, mikäli katsot, että henkilötietojen käsittelyssä rikotaan EU:n yleistä tietosuojasetusta (EU) 2016/679. Suomessa valvontaviranomainen on tietosuojavaltuutettu.

Tietosuojavaltuutetun toimiston ajantasaiset yhteystiedot: <https://tietosuoja.fi/etusivu>

## Liite 8. Esimerkkejä ohjelmointiroboteista ja -ympäristöistä

### ESIMERKKEJÄ OHJELMOITAVISTA ROBOTEISTA

nimi	kuvaus	verkkosivu
Bee-Bot	 Bee-Bot -lattiarobotti soveltuu ohjelmoinnin alkeiden harjoitteluun, suunta- ja sijaintikäsitteiden havainnollistamiseen.	<a href="https://www.tts-group.co.uk/bee-bot-programmable-floor-robot/1015268.html">https://www.tts-group.co.uk/bee-bot-programmable-floor-robot/1015268.html</a>
Blue-Bot	 Kuten Bee-Bot, mutta Blue-Bot -lattiarobottia voidaan ohjelmoida painonappien lisäksi myös iPadilla bluetooth-yhteyden välityksellä	<a href="https://www.tts-group.co.uk/blue-bot-bluetooth-programmable-floor-robot/1015269.html">https://www.tts-group.co.uk/blue-bot-bluetooth-programmable-floor-robot/1015269.html</a>
Pro-Bot	 Pro-Bot -lattiarobottia voidaan ohjelmoida painonappien lisäksi myös tietokoneen avulla. Robotissa on valo-, kosketus ja äänisensorit.	<a href="https://www.tts-international.com/pro-bot-rechargeable-floor-robot/1009825.html">https://www.tts-international.com/pro-bot-rechargeable-floor-robot/1009825.html</a>
Sphero BOLT	 Sphero BOLT -robottia ohjelmoidaan iPadillä graafisessa ohjelmointiympäristössä.	<a href="https://www.sphero.com">https://www.sphero.com</a>
LEGO MindStorms EV3	 EV3 -roboteissa yhdistyy LEGO technics palikoiden rakentelu ja ohjelmointi iPadin tai tietokoneen avulla graafista käyttöliittymää käyttäen. Sisältää useita sensoreita, kuten valo, kosketus ja gyro.	<a href="https://www.lego.com/fi-fi/product/lego-mindstorms-ev3-31313">https://www.lego.com/fi-fi/product/lego-mindstorms-ev3-31313</a>
Tello	 Tello -dronea ohjelmoidaan iPadillä graafisessa ohjelmointi- ympäristössä. Tellossa on sisäänrakennettu kamera.	<a href="https://www.ryzerobotics.com/tello">https://www.ryzerobotics.com/tello</a>

### ESIMERKKEJÄ GRAAFISISTA OHJELMOINTIYMPÄRISTÖISTÄ

nimi	kuvaus	verkkosivu
code.org	 code.org -ympäristössä ohjelmoidaan tuttuja animaatiohahmoja hauskojen tehtävien muodossa. Lohkokoodaus soveltuu hyvin ensimmäiseksi ohjelmointikieleksi. Sivusto tarjoaa harjoituksia kaiken tasoisille.	<a href="https://www.code.org/">https://www.code.org/</a>
Scratch (Jr.)	 Scratch -ympäristössä ohjelmoidaan hahmoa lohkokoodauksen keinoin. Ympäristössä voidaan toteuttaa hyvinkin monipuolisia ohjelmointiharjoituksia. Soveltuu hyvin ohjelmointialgoritmien harjoitteluun.	<a href="https://www.tts-group.co.uk/blue-bot-bluetooth-programmable-floor-robot/1015269.html">https://www.tts-group.co.uk/blue-bot-bluetooth-programmable-floor-robot/1015269.html</a>
Swift Playgrounds	 Applen kehittämän Swift-ohjelmointikielen opetteluun ja harjoitteluun. Sopii myös aloittaville koodaajille.	<a href="https://www.apple.com/ios/swift/playgrounds/">https://www.apple.com/ios/swift/playgrounds/</a>