

# Progression in K-9 CS education

NordNICE workshop, Uppsala

October 20-21, 2016

Linda Mannila

[linda.mannila@abo.fi](mailto:linda.mannila@abo.fi)

@lindamannila





# In a nutshell

- Programming included in mathematics, crafts and as an interdisciplinary trait in all subjects
- The national curriculum provides a short description of the goals at different levels and in different subjects
- Local curricula (can) provide more detail
- National board of education offers support material

	Tavoitteet ja sisällöt OPS	Tavoitteet ja sisällöt käytännössä	Laitteet, välineet ja ohjelmat, joita oppimisessa voidaan hyödyntää
Vuosiluokat 1-2	<p><b>Tieto- ja viestintäteknologinen osaaminen:</b></p> <p>- Oppilaat saavat ja jakavat keskenään kokemuksia digitaalisen median parissa työskentelystä sekä ikäkaudelle sopivasta ohjelmoinnista.</p> <p><b>Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 1–2 :</b></p> <p>- Tutustuminen ohjelmoinnin alkeisiin alkaa laatimalla vaihteittaisia toimintaohjeita, joita myös testataan.</p>	<p>Tutustutaan ohjelmoinnin perusasioihin esimerkiksi erilaisten leikkien ja pelien avulla:</p> <ul style="list-style-type: none"> <li>- ohjeiden antaminen</li> <li>- ohjeiden seuraaminen tietokonemaisesti</li> <li>- syy-seuraussuhteet</li> <li>- komentosarja-, ehto- ja toisto-käsitteet</li> <li>- toistuvien kaavojen tunnistaminen</li> </ul> <p>Harjoitellaan toimintaohjeiden täsmällistä antamista, testaamista ja korjaamista.</p>	<p>Opetusta voidaan toteuttaa myös ilman tietoteknistä laitteistoa.</p> <p>Hyödynnetään erilaisia pelejä ja leikkejä.</p> <p>Esimerkiksi:</p> <ul style="list-style-type: none"> <li>- appsit (esimerkiksi Bee-Bot, Lightbot, Kodable, The Foes, Fix the Factory)</li> <li>- visuaaliset ohjelmointikielet (Scratch Jr., Snap)</li> <li>- robotit (bee-bot, blue-bot)</li> <li>- materiaalikannat (CodeStudio, Koodaustunti, Koodi2016, Koodiaapinen, code.org)</li> </ul>
Vuosiluokat 3-6	<p><b>Tieto- ja viestintäteknologinen osaaminen:</b></p> <p>- Ohjelmointia kokeillessaan oppilaat saavat kokemuksia siitä, miten teknologian toiminta riippuu ihmisen tekemistä ratkaisuista.</p> <p><b>Matematiikan opetuksen tavoitteet vuosiluokilla 3–6:</b></p> <p>- Opetuksen tavoitteet: T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoina graafisessa ohjelmointiympäristössä.</p> <p><b>Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 3–6 (S1 Ajattelun taidot):</b></p> <p>- Suunnitellaan ja toteutetaan ohjelmia graafisessa ohjelmointiympäristössä.</p> <p><b>Matematiikan arviointikriteerit 6. vuosiluokan päätteeksi hyvää osaamista kuvaavaa sanallista arviota / arvosanaa kahdeksan varten:</b></p> <p>- Opetuksen tavoitteet: T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoina graafisessa ohjelmointiympäristössä.</p> <p>- Arvioinnin kohteet oppiaineessa: ohjelmointi graafisessa ohjelmointiympäristössä.</p> <p>- Hyvä/arvosanan kahdeksan osaaminen: oppilas osaa ohjelmoida toimivan ohjelman graafisessa ohjelmointiympäristössä.</p> <p><b>Käsityön tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 3–6 (S3 Kokeilu):</b></p> <p>- Harjoitellaan ohjelmoimalla aikaan saatuja toimintoja, joista esimerkkinä robotiikka ja automaatio.</p>	<p>Tutustutaan ohjelmoinnin perusasioihin esimerkiksi erilaisten leikkien ja pelien avulla:</p> <ul style="list-style-type: none"> <li>- ohjeiden antaminen</li> <li>- ohjeiden seuraaminen tietokonemaisesti</li> <li>- syy-seuraussuhteet</li> <li>- käsitteisiin tutustuminen sekä niiden havainnollistaminen</li> <li>- toistuvien kaavojen tunnistaminen</li> </ul> <p>Opetellaan ohjelmoinnin perusteita graafisissa ohjelmointiympäristöissä</p> <ul style="list-style-type: none"> <li>- kehitetään taitoa antaa täsmällisiä ohjeita</li> <li>- käytetään lauseita, ehto- ja toistorakenteita</li> </ul> <p>Tarkastellaan ohjelmoinnin läsnäoloa ympäröivässä yhteiskunnassa.</p> <p>Ohjelmointia käytetään tuottamisen ja luovan ilmaisun välineenä.</p>	<p>Käytetään monipuolisesti erilaisia ohjelmointikieliä ja -ympäristöjä.</p> <p>Hyödynnetään erilaisia pelejä ja leikkejä.</p> <p>Esimerkiksi:</p> <ul style="list-style-type: none"> <li>- appsit (esimerkiksi Bee-Bot, Lightbot, Kodable, The Foes, Fix the Factory)</li> <li>- visuaaliset ohjelmointikielet (Scratch, Snap, HopScotch, Tynker, Pyonkee, Kodu, ScratchJr)</li> <li>- robotit (Bee-Bot, Blue-Bot, LEGO Mindstorms)</li> <li>- materiaalikannat (CodeStudio, Koodaustunti, Koodi2016, Koodiaapinen, code.org)</li> <li>- mikrokontrollerit (esimerkiksi Arduino, Adafruit Gemma, LilyPad Arduino)</li> </ul>
Vuosiluokat 7-9	<p><b>Tieto- ja viestintäteknologinen osaaminen:</b></p> <p>- Ohjelmointia harjoitellaan osana eri oppiaineiden opintoja.</p> <p><b>Matematiikan opetuksen tavoitteet vuosiluokilla 7–9:</b></p> <p>- Opetuksen tavoitteet: T20 ohjata oppilasta kehittämään algoritmista ajatteluaan sekä taitojaan soveltaa matematiikkaa ja ohjelmointia ongelmien ratkaisemiseen.</p> <p><b>Matematiikan tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 7–9 (S1 Ajattelun taidot ja menetelmät)</b></p> <p>- Ohjelmoidaan ja samalla harjoitellaan hyviä ohjelmointikäytäntöjä. Sovelletaan itse tehtyjä tai valmiita tietokoneohjelmia osana matematiikan opiskelua.</p> <p><b>Matematiikan päättöarvioinnin kriteerit hyvälle osaamiselle (arvosanalle 8) oppimäärän päätyttyä</b></p> <p>- Opetuksen tavoitteet: T20 ohjata oppilasta kehittämään algoritmista ajatteluaan sekä taitojaan soveltaa matematiikkaa ja ohjelmointia ongelmien ratkaisemiseen.</p> <p>- Arvioinnin kohteet oppiaineessa: algoritmien ajattelu ja ohjelmointitaidot.</p> <p>- Arvosanan kahdeksan osaaminen: oppilas osaa soveltaa algoritmisen ajattelun periaatteita ja osaa ohjelmoida yksinkertaisia ohjelmia.</p> <p><b>Käsityön tavoitteisiin liittyvät keskeiset sisältöalueet vuosiluokilla 7–9 (S3 Kokeilu)</b></p> <p>- Käytetään sulautettuja järjestelmiä käsityöhön eli sovelletaan ohjelmointia suunnitelmiin ja valmistettaviin tuotteisiin.</p>	<p>Saadaan aikaan monipuolisesti positiivisia kokemuksia ohjelmointiin liittyen.</p> <p>Harjoitellaan ohjelmoinnin perusrakenteiden käyttöä:</p> <ul style="list-style-type: none"> <li>- muuttujat, ehtorakenne, toistorakenne sekä aliohjelmat ja funktiot</li> </ul> <p>Nivelletään ohjelmointia osaksi eri oppiaineiden opetusta.</p> <p>Osataan ratkaista ongelmia ja toteuttaa omia ideoita käyttämällä visuaalisia tai tekstipohjaisia ohjelmointikieliä.</p> <p>Kehitetään ohjelmointiin liittyviä ajattelu- ja suunnittelutaitoja:</p> <ul style="list-style-type: none"> <li>- vertailu</li> <li>- luokittelu</li> <li>- ongelman analysointi</li> <li>- päättely</li> <li>- mahdollisten ratkaisujen arviointi</li> <li>- virheenkorjaus</li> </ul> <p>Tarkastellaan ohjelmoinnin läsnäoloa ympäröivässä yhteiskunnassa.</p> <p>Ohjelmointia käytetään tuottamisen ja luovan ilmaisun välineenä.</p>	<p>Käytetään monipuolisesti erilaisia ohjelmointikieliä ja -ympäristöjä.</p> <p>Esimerkiksi:</p> <ul style="list-style-type: none"> <li>- appsit (esimerkiksi Lightbot, Fix the Factor)</li> <li>- visuaaliset ohjelmointikielet (Scratch, Snap, HopScotch, Tynker, Pyonkee, Kodu)</li> <li>- ohjelmointikielet (Python, Ruby, JavaScript, C++, Java, Processing, Racket)</li> <li>- robotit (Edison, LEGO Mindstorms, mBot)</li> <li>- materiaalikannat (CodeStudio, Koodaustunti, Koodi2016, Koodiaapinen, code.org)</li> <li>- mikrokontrollerit (esimerkiksi Arduino, Adafruit Gemma, LilyPad Arduino)</li> </ul>

# Challenge #1

Creating a progression without knowing more

- What is difficult for children and youth?
- When can concepts be introduced?

Much research has been done on novice programming at secondary and tertiary level, but barely none at K-9 level

- How can we involve teachers in research?
- What can their role be?

## Challenge #2

Building a basis in later grades during the transition period from the old to the new curriculum

- Teachers cannot assume that the pupils have any background knowledge

How can we package the basics for use in later grades?



# Challenge #3

## Defining goals based on grade

- Pupils have differing backgrounds, attitudes and motivation levels
- We cannot “hold back” on some concepts just because “pupils need something to learn later on as well”

How can we create a progression, which is not based on the grades? Can it be done?

# Challenge #3

## Defining progress within a subject

- For some subjects it is rather “easy” (e.g., music, crafts)
- For others, quite challenging (e.g. digital storytelling in languages or history)

## Do we need intra-subject progression?

- Or is it enough to have a progression in the curriculum as a whole?





# Open questions

What is the main objective for teaching programming at K-9 level?

To what extent can informal learning activities (camps, clubs, etc) be utilized to support teachers?

