

# Ohjelmointiin ja Internetiin liittyviä harjoituksia ja projekteja eri luokka-asteille



Kuva: Pete Linforth (TheDigitalArtist), Pixabay



Materiaali on laadittu osana LUMATIKKA-täydennyskoulutusohjelmaa, jonka toteutuksesta vastaa LUMA-keskus Suomi -verkosto yhteistyökumppaneineen. Ohjelman rahoittaa Opetushallitus. Tämä materiaali on julkaistu CC BY-SA 4.0 -lisenssillä.

Materiaalin ovat tehneet Tony Puustinen, Lea Kutvonen ja Virpi Sumu Helsingin yliopistolta.

Voit ladata muokattavan version materiaalista <https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka3/harjoitus-pdf-muokattava.zip>

Päivämäärä 17.6.2020

# Ohje opettajille

Tässä materiaalissa on pyritty antamaan karkeita suuntaviivoja ohjelmoinnin oppimispolulle. Ohjelmointi on kuitenkin uusi teema kouluissa, joten sen luokka-astekohtaiset sisällöt ovat vielä muotoutumisvaiheessa.

Siksi on suositeltavaa, että opettajat tutustuvat laajasti kaikkien luokka-asteiden sisältöihin ja valikoivat niistä omalle ikäryhmälleen sopivia harjoituksia. Materiaali sopii varhaiskasvatukseen, perusopetukseen, toisen asteen koulutukseen sekä aikuiskoulutukseen.

Tarvitsemme innokkaita ja kokeilunhaluisia opettajia tekemään monenlaisia opetuskokeiluja ohjelmoinnin opetuksessa sekä sen soveltamisessa työkaluna muissa oppiaineissa, ja jakamaan omia kokemuksiaan sekä parhaita käytäntöjä muille opettajille. Voisi jopa sanoa, että ohjelmoinnin opetuksen parissa opettajilla on ainutlaatuiset mahdollisuudet päästä kokeilemaan – lähtien lähes puhtaalta pöydältä – uusia tapoja lähestyä opetusta ja oppimista ilmiölähtöisesti, ongelmalähtöisesti ja projektilähtöisesti.

Lisää luettavaa harjoituksista ja tavoitteista löydät tekstistä

*Asiaa harjoituksista ja tavoitteista eri luokka-asteille*

<https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka3/asiaa-harjoituksista-ja-tavoitteista-eri-luokka-asteille.pdf>

# Muistutus tietosuojasta sekä oppilaan omien laitteiden käytöstä opetuksessa

Yleisesti ottaen kaikkien tunnuksellisten palvelujen käyttämisessä aikuiset huolehtivat lasten tietosuojasta, ottamalla huomioon asioita kuten lasten ikä, käytettävän palvelun käyttösäännöt, kunnan opetustoimen antamat ohjeistukset sekä koulun omat säännöt. Opetus- ja kulttuuriministeriöltä löytyy myös oppilaitoksille tarkoitettu [tietosuojaopas](#).

Oppilaan omaa laitetta voidaan käyttää koulussa huoltajan tai 15-vuotiaan suostumuksella. Opettaja päättää laitteiden mahdollisesta opetuskäytöstä oppitunnilla. Opetushallituksella on [ohjeistus](#) aiheeseen liittyen.

Tietosuojalain ja tietosuoja-asetuksen mukaan 13 vuotta täyttänyt voi antaa suostumuksen tietoyhteiskunnan palveluiden (kuten sosiaalisen median palvelut tai pikaviestipalvelut) käyttöön.

# Sisällysluettelo

1. Ohjelmointia ilman tietokoneita (esimerkiksi varhaiskasvatukseen ja alakouluun)
2. Visuaalista ohjelmointia (alakoulu)
3. Tekstipohjaista ohjelmointia (yläkoulu ja edistyneemmät alakoululaiset)
4. Projekteja yläkouluun ja lukioon
5. Tieto- ja viestintäteknologian turvallinen käyttö, globaalit riskit, tietosuojaan perehtyminen sekä yksityisyyden suojaaminen (perusopetus ja lukio)

Ohjelmointia ilman  
tietokoneita (esimerkiksi  
varhaiskasvatukseen ja  
alakouluun)

# Ohjelmoinnillisen ajattelun harjoitteita ilman laitteita (opettajan ohje)

1. Avaa Innokas-verkoston materiaalipankki <https://www.innokas.fi/materiaalit/> ja valitse sieltä joko
  - [Ohjelmoinnillinen ajattelu](#) ja sen sisältämä materiaali *Ohjelmoinnillisen ajattelun harjoitteita ilman laitteita* tai
  - [Lähtölaukaus koodaukseen](#) ja sen sisältämä materiaali.
2. Valitse materiaaleista jokin sinun opetukseesi sopiva aktiviteetti, kuten vaikkapa *Ope-robotin ohjelmointi*, *Viruskaappari* tai *Algoritmien muodostaminen*.
3. Toteuta aktiviteetti lasten kanssa!

# Leikkejä materiaalista 'Ohjelmoinnin ABC varhaiskasvatukseen' (opettajan ohje)

1. Mene osoitteeseen <https://helda.helsinki.fi/handle/10138/301730> ja avaa opas *Ohjelmoinnin ABC varhaiskasvatukseen*

2. Valitse oppaasta jokin seuraavista leikeistä ja toteuta se lasten kanssa:

- Kommunikaatioleikki symboleilla
- HARRI-robotti
- LEGO-LABYRINTTI
- Näkymätön Lela
- Mikä tekee koirasta koiran?
- Pelit ohjelmoinnissa
- Beebot ja Bluebot -Robottilelut
- Rakentelu ja ohjelmointi, Lego Education WeDo 2.0
- Tietokoneleikki
- Robottileikki
- Ohjelmoinnillisia välipaloja



# Visuaalista ohjelmointia (alakoulu)

# Visuaalisen ohjelmoinnin alkeiden opettelu Scratchillä

(opettajan ohje, sivu 1/2)

[Scratch-ohjelmointiympäristö](#) on yhdysvaltalaisen MIT-korkeakoulun tuotos ja se on vapaasti käytettävissä.

Scratch-yhteisössä tunnukset ovat pseudonyymejä, mikä tarkoittaa että aitoja nimiä ei tarvitse käyttää, vaan kaikilla on erillinen näkyvä nimimerkki. Tehtäviä voi toki palvelussa tehdä kirjautumattakin, mutta projekteja ei silloin voi jakaa toisten nähtäväksi eikä esittää keskusteluryhmissä kysymyksiä. Kaikki Scratch-järjestelmän keskustelupalstat ovat valvottuja. Periaatteena on nostaa nuoria edistyneitä käyttäjiä palstavahdeiksi ja antaa heille valtaa esimerkiksi nostaa esiin omia suosikkipelejäan jaettujen projektien joukosta. Alle 13-vuotiaan tunnuksen luonnissa tarvitaan vanhemman tai huoltajan sähköpostiosoite.

# Visuaalisen ohjelmoinnin alkeiden opettelua Scratchillä (opettajan ohje, sivu 2/2)

1. Mene osoitteeseen <https://scratch.mit.edu/>, vaihda kieleksi Suomi ja luo itsellesi tunnukset sivun yläreunassa olevalla painikkeella *Liity Scratchiin*
2. Mene sitten Linkin materiaalipankkiin osoitteessa <http://linkki.cs.helsinki.fi/cgi-bin/debbie-action-materiaalit?syn>
3. Valitse Linkin materiaalipankista mikä tahansa aloittelijoille tarkoitettu peli
4. Ohjelmoi valitsemasi peli opettajan ohjeen, lapsille tarkoitetun ohjeen ja mahdollisen opettajan videon avulla
5. Virheenjäljityksen apuna voit käyttää seuraavia ohjeita:  
[https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka/luokka-asteet/debuggauksen\\_abc.pdf](https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka/luokka-asteet/debuggauksen_abc.pdf)
6. Julkaise valmis peli Scratchin verkkosivulla <https://scratch.mit.edu/>
7. Järjestä tämän itsenäisen harjoittelusi jälkeen oppitunti ohjelmoinnista, jossa sinä toimit sekä valmentajan että oppijan roolissa! (Mikäli tarvitset valmiita ohjeita, niin voit hyödyntää [Pulmaario-ohjaajan opasta](#))

# Visuaalisen ohjelmoinnin alkeiden opettelua Scratchillä

(oppilaan ohje)

1. Mene osoitteeseen <https://scratch.mit.edu/>, sivun alareunasta vaihda kieleksi Suomi ja luo itsellesi tunnukset sivun yläreunassa olevalla painikkeella *Liity Scratchiin*
2. Mene Scratchin ohjelmointiympäristöön klikkaamalla sivun yläreunassa olevaa *Luo*-painiketta
3. Tee peli seuraavan ohjeen mukaan: [opettaja sijoittaa tähän linkin edellisen sivun ohjeiden mukaisesti]
4. Kun peli on valmis, voit julkaista sen Scratchin verkkosivulla <https://scratch.mit.edu/> tai sitten tallentaa projektin tietokoneellesi kohdasta *Tiedosto > Tallenna tietokoneellesi*

Tekstipohjaista ohjelmointia  
(yläkoulu ja edistyneemmät  
alakoululaiset)

# Tekstipohjaisen ohjelmoinnin alkeiden opettelua verkkokurssilla

(opettajan ohje)

1. Mene osoitteeseen <https://tie.koodariksi.fi>
2. Klikkaa *Kirjaudu sisään* -painiketta ja luo siellä annettujen ohjeiden mukaisesti itsellesi tunnukset
3. Siirry sivuston etusivulta kurssille *Ohjelmoinnin alkeet* ja suorita kurssin neljä ensimmäistä lukua
4. Tilaa tunnuksellesi opettajan oikeudet etusivulla annettujen ohjeiden mukaisesti
5. Järjestä oppitunti ohjelmoinnista, missä oppilaat pääsevät itsenäisesti etenemään kurssilla, ja jossa sinä toimit sekä valmentajan että oppijan roolissa!

# Tekstipohjaisen ohjelmoinnin alkeiden opettelua verkkokurssilla

(oppilaan ohje)

1. Mene osoitteeseen <https://tie.koodariksi.fi>
2. Klikkaa Kirjaudu sisään -painiketta ja luo siellä annettujen ohjeiden mukaisesti itsellesi tunnukset
3. Siirry sivuston etusivulta kurssille *Ohjelmoinnin alkeet* ja liity ryhmään saamallasi ryhmäavaimella
4. Ala suorittamaan verkkokurssia itsenäisesti
5. Ennen kuin kysyt luokkatoverilta tai opettajalta apua, kokeile ratkaista tehtävää sinnikkäästi itse sekä tarvittaessa etsi hakukoneen avulla videoita ja muita materiaaleja, jotka auttavat sinua ongelman ratkaisemisessa. Virheenjäljityksessä hyödynnä myös seuraavaa ohjetta:  
[https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka/luokka-asteet/debuggauksen\\_abc.pdf](https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka/luokka-asteet/debuggauksen_abc.pdf)

# Projekteja yläkouluun ja lukioon



# Tutkimustehtäviä: Matematiikan sovelluksia ohjelmoinnissa (sivu 1/2)

## Koneoppiminen

"Tekoälyn" kehittäminen perustuu [koneoppimiseen](#).

Selvitä minkälaisen matematiikan päälle koneoppiminen rakentuu. Missä tai minkä materiaalien avulla voisit oppia koneoppimista ja siihen liittyvää matematiikkaa?

## Kryptografia

Käyttäessäsi Internet-palveluita (kuten verkkosivuja tai pikaviestipalveluita), sinun laitteesi ja käyttämäsi palvelut vaihtavat keskenään monenlaista tietoa. Usein halutaan, että tämä vaihdettu tieto pysyy (ainakin osittain) luottamuksellisena laitteesi ja palvelun välillä.

Tämän luottamuksellisuuden saavuttamiseksi, laitteesi ja palvelu käyttävät tiedon vaihtamisen yhteydessä erilaisia [salakirjoitusmenetelmiä](#) sekä [digitaalisia allekirjoituksia](#). Tällöin kolmansien osapuolien on hankalampi selvittää mitä tietoa on vaihdettu tai muokata sitä matkan varrella. Tiedon salaaminen ja digitaaliset allekirjoitukset liittyvät [kryptografiaan](#).

Selvitä minkälaista matematiikkaa kryptografiassa tarvitaan. Missä tai minkä materiaalien avulla voisit oppia tätä matematiikkaa?

# Tutkimustehtäviä: Matematiikan sovelluksia ohjelmoinnissa (sivu 2/2)

## Mallintaminen ja tietokonegrafiikka

Kun halutaan [mallintaa](#) ja [simuloida](#) ajasta riippuvia systeemejä, kuten kappaleiden liikettä sisältäviä tai muita fysikaalisia systeemejä, tarvitaan vektoreita, trigonometrisia funktioita sekä [differentiaali-](#) ja [integraaliyhtälöitä](#). Tietokonegrafiikassa (2D tai 3D) tarvitaan usein myös juuri samoja käsitteitä ja niiden lisäksi vielä [matriiseja](#) sekä [kompleksilukuja](#). Populaatioiden kasvun mallintamisessa tarvitaan myös differentiaali- ja integraaliyhtälöitä.

Selvitä mitä ovat differentiaali- ja integraaliyhtälöt. Millä tavalla ne liittyvät lukiossa opetettavaan matematiikkaan?

## Formaalit menetelmät

Kun halutaan rakentaa äärimmäisen virhevapaita ohjelmia, ohjelmia joista esimerkiksi ihmishenget saattavat riippua, voidaan käyttää [formaaleja menetelmiä](#). Formaali verifiointi kuuluu näihin menetelmiin, ja siinä *matemaattisesti todistetaan*, että ohjelmakoodi toteuttaa tietyn (suunnitelman mukaisen) matemaattisen spesifikaation. Formaaleissa menetelmissä sovelletaan muun muassa [matemaattista logiikkaa](#).

Selvitä minkälaista logiikkaa lukiossa on mahdollista oppia.

# Ideoita ohjelmointiprojektien aiheiksi matematiikan käsitteiden oppimista varten

- Murtolukujen havainnollistaminen omalla ohjelmalla, suhde prosentteihin ja desimaalilukuihin
- Tarkkuus ja pyöristysvirheiden kumuloituminen laskennassa
- Tekijöihin jako, alkulukujen laskeminen
- Pythagoraan kolmio, Pascalin kolmio
- Koordinaatisto pelikenttänä havainnoiden
- Vektorilaskenta koordinaatistossa
- Trigonometrinen funktioiden perusteiden havainnoiminen ja käyttö käytännöllisissä tilanteissa
- Napakoordinaattilaskenta [ruusukäyriä](#) tai [Arkhimedeen spiraalia](#) tutkien (esimerkiksi lasten tuottamia teoksia voisi löytää [Scratch-yhteisöstä](#) hakusanoilla 'polar rose' tai 'spiraali')
- Kolmiulotteisten kappaleiden tilavuuden tai pinta-alojen havainnollistaminen yksikköneliöillä tai -kuutiolla.
- [Demojen](#) eli audiovisuaalisten taideteosten ohjelmointi, hyödyntäen esimerkiksi [Processing-ohjelmointikieltä](#)

# Ideoita projektien aiheiksi, joissa ohjelmointia käytetään opiskelu- tai havainnointimenetelmänä

- Fyysisten ilmiöiden simulointi, esimerkkinä [gravitaatio](#) ja [väliaineen aiheuttama vastus](#), joiden toteuttamiseksi Processing-ohjelmointikielellä voi saada apua esimerkiksi verkkokirjasta [The Nature of Code](#) (kirja on saatavilla ilmaiseksi, mutta siitä voi myös halutessaan maksaa)
- Usean funktion samanaikaisen vaikutuksen havainnoiminen
- Graafinen optimointi kun epäyhtälöitä on useita
- Kemiallisten aineiden rakentelu tai reaktioyhtälöiden tasapainottaminen, hyödyntäen esimerkiksi [näitä sovelluksia](#)
- Tilastoanalyysi (keskilukujen ja hajontalukujen harhaanjohtavuuksien kokeiluja)
- Todennäköisyyslaskentaa (satunnaisuus piin desimaaleissa)
- Kappaleiden [projektoiden](#) käsittely
- Heijastumisen laskeminen
- Lukusarjojen laskeminen, niiden summien likiarvot, hajaantuminen ja suppeneminen näkyviin kokeilemalla
- [L-systeemit](#) piirtämisessä ja “luonnollisten muotojen” muodostamisessa, joiden toteuttamiseksi voi saada vinkkejä tästä [blogikirjoituksesta](#) tai Scratch-ohjelmasta [Lumihiutale](#)

Tieto- ja viestintäteknologian  
turvallinen käyttö, globaalit  
riskit, tietosuojan  
perehtyminen sekä  
yksityisyyden suojaaminen  
(perusopetus ja lukio)

# Opettajille: Seuraavien harjoitusten liittyminen opetussuunnitelmiin

Perusopetuksen opetussuunnitelmassa 2016 asetetaan korkean tason tavoitteita laaja-alaiselle osaamiselle, kuten tieto- ja viestintäteknologian osaamiselle, arjen taidoille, monilukutaidolle, yrittäjävalmiuksille ja tulevaisuuden rakentamiseen liittyville valmiuksille.

Lukion opetussuunnitelmassa 2015 asetetaan tavoitteita muun muassa aihekokonaisuuksille monilukutaito ja mediat, sekä teknologia ja yhteiskunta.

Edellä mainittuihin kokonaisuuksiin liittyen perusopetuksen opetussuunnitelma asettaa tavoitteita tieto- ja viestintäteknologian käyttö- ja toimintaperiaatteiden ymmärtämisestä, turvallisesta käytöstä sekä siihen liittyvien globaalien riskien hahmottamisesta. Tietosuojan perehtyminen, tietoturvalliset toimintatavat sekä oman yksityisyyden suojaaminen ja muiden yksityisyyden kunnioittaminen nostetaan myös tavoitteiksi opetussuunnitelmassa. Samoin opetussuunnitelmassa asetetaan tavoitteita medialukutaidolle ja mediakasvatukselle.

# Koulun verkkoon tutustuminen

Tutustukaa yhdessä koulunne IT-tuen kanssa teidän koulunne verkkoon. Yrittäkää selvittää vastauksia esimerkiksi seuraaviin kysymyksiin:

- Kuinka monta tietokonetta verkkoon kuuluu?
- Miten laitteet ovat kytketty toisiinsa ja miten niiden välillä voidaan siirtää tietoa?
- Mikä palveluntarjoaja yhdistää koulun verkon Internetiin?
- Mitä sähköpostipalveluntarjoajaa koulunne käyttää?
- Mitä pilvipalveluita koulunne käyttää? Toimittaako näitä palveluita suomalaiset vai ulkomaalaiset yritykset?
- Minkä yrityksen palvelimilta koulunne verkkosivut toimitetaan?
- Minkälaisia tietoja koulunne verkkosivut keräävät käyttäjistä? Käyttävätkö koulunne sivut jonkin kolmannen osapuolen analytiikkapalveluita? (esimerkiksi selainten Web-työkaluilla voi selvittää tätä asiaa)
- Mitä ohjelmistoja ja käyttöjärjestelmiä koulussanne käytetään?
- Kuka ylläpitää verkkoa, ohjelmistoja ja koulunne laitteita? Kuinka suuret kustannukset hankinnoista, ylläpidosta ja lisensseistä aiheutuu?

# Tilastointitehtävä: Mistä suomalaiset verkkosivut tulevat? Mitä sähköpostinpalveluntarjoajaa ne käyttävät? Käyttävätkö ne analytiikkaa?

Katsokaa ensin video [Lyhyt kurkistus Internetin toimintaan ja pilvipalveluihin](#). Sen jälkeen selvittäkää ja muodostakaa tilastot:

1. Minkä yrityksen palvelimilta suosituimmat suomalaiset verkkosivut toimitetaan?
2. Mitä sähköpostipalveluntarjoajia nämä tutkianne verkkosivut käyttävät?
3. Kuinka monella sivuista on Google Analytics tai jokin muu kolmannen osapuolen analytiikkapalvelu käytössä?

Tässä alla on muistilista videolla näytetyistä työkaluista ja resursseista, joita tulette tarvitsemaan tehtävän suorittamiseksi:

- nslookup- tai dig-työkalu, esimerkki komennot:
  - nslookup -type=a VERKKOTUNNUS
  - nslookup -type=mx VERKKOTUNNUS
- IANA:n [IP-osoite rekisteri](#) ja [RIR](#):ien verkkosivut
- selaimen Web-työkalut (ohjeet: [Firefoxille](#), [Edgelle](#) ja [Chromelle](#))

*Vinkki: Ohjelmoinnin avulla on mahdollista tehdä esimerkiksi tuhannen verkkosivun otanta, mutta ohjelmoinnin soveltaminen tai näin suuren otannan ottaminen ei ole välttämätöntä.*



# Keinoja oman sekä muiden yksityisyyden ja tietoturvan lisäämiseen Internetissä (sivu 1/4)

Tietosuojan perehtyminen, tietoturvalliset toimintatavat sekä oman yksityisyyden suojaaminen ja muiden yksityisyyden kunnioittaminen nostetaan tavoitteiksi perusopetuksen opetussuunnitelmassa (2016).

Tässä kalvosarjassa on esitetty kokeiltavaksi kouluihin joitakin keinoja ja työkaluja, joiden avulla on mahdollista suojata omaa ja muiden yksityisyyttä.

Monessa harjoituksessa on tarpeellista asentaa ohjelmia tietokoneille tai muille laitteille, joten harjoitukset eivät välttämättä sovi kaikkiin ympäristöihin. Opettaja arvioi tapauskohtaisesti, voiko oppilaita pyytää asentamaan ohjelmia laitteilleen, vai tutkitaanko niitä yhdessä esimerkiksi opettajan laitteilta. Oppilaiden käyttöoikeuksia koulun laitteilla voi olla myös rajattu sellaisella tavalla, joka hankaloittaa harjoitusten toteuttamista.

## Johdanto

YK:n vuoden 2018 yleiskokouksen päätelausemassa ([The right to privacy in the digital age, 73/179](#)) sanotaan muun muassa seuraavaa

### Lainaus 1

*Emphasizing that the protection and promotion of, and respect for, the right to privacy benefit from sustained engagement, including through informal dialogues, among all stakeholders, including States, business enterprises, international organizations and civil society,*

### Lainaus 2

*Calls upon all States: (k) To promote quality education and lifelong educational opportunities for all to foster, inter alia, digital literacy and technical skills to effectively protect privacy;*

Päätelausemassa siis kehoitetaan jäsenvaltioita tarjoamaan kaikille kansalaisille mahdollisuuksia oppia teknisiä taitoja oman yksityisyyden suojaamiseksi.

# Keinoja oman sekä muiden yksityisyyden ja tietoturvan lisäämiseen Internetissä (sivu 2/4)

## Oman viestinnän suojaaminen

Suosituimpia pikaviestipalveluita on Facebookin omistama WhatsApp. Viestien vaihtaminen palvelussa on suojattu [Signal-protokollan](#) avulla. Tästä suojauksesta huolimatta, on palvelussa ainakin kolme huolenaihetta yksityisyyden suojaan liittyen.

Ensinnäkin, keskeinen osa Facebookin liiketoimintamallia on kohdennetun mainonnan toteuttaminen, minkä vuoksi Facebookilla on taloudelliset kannustimet kerätä heidän palveluidensa käyttäjistä dataa sekä muodostaa heistä erinäisiä profileja.

Toiseksi, palvelulla on tieto käyttäjän kaikista puhelimeen lisätystä kontakteista sekä mahdollisuus kerätä monenlaista aktiivisuusdataa ja sijaintidataa. Kenelle käyttäjä on lähettänyt viestejä? Mihin kellonaikaan? Kuinka useasti? Kenen profiilikuvaa käyttäjä on katsonut ja milloin? Missä käyttäjä on ollut näinä hetkinä?

Kolmanneksi, WhatsApp-sovelluksen lähdekoodi ei ole [avoin](#), mikä tekee ohjelman toiminnan analysoimisesta niin hankalaa, että joudumme käytännössä luottamaan Facebookin sanaan siitä, mitä tietoa ohjelma oikeasti lähettää. Tästä johtuen sovelluksen tekijöiden on myös teknisesti helppo piilottaa takaportteja sovellukseen sekä lisätä niitä päivitysten yhteydessä, jos vaikkapa paikallinen lainsäädäntö tai muu ulkoinen painostus siihen velvoittaisi. Sopivalla takaportilla olisi mahdollista esimerkiksi kytkeä pois päältä Signal-protokollan noudattaminen valikoitujen käyttäjien sovelluksista, tai lähettää kaikki heidän WhatsApp-sovelluksissaan tallessa olevat viestit jollekin kolmannelle osapuolelle. Kohteena olevat käyttäjät olisi mahdollista valikoida yksitellen, tai käyttää erilaisia algoritmeja ja koneellisesti valikoida tietyt kriteerit täyttävät käyttäjät.

## Tehtävänanto

- Privacy Tools -sivusto [suosittelee](#) WhatsAppin korvaajaksi avoimen lähdekoodin [Signal](#)-pikaviestisovellusta. Asentakaa ohjelma laitteillemme ja kokeilkaa sen käyttämistä.
- Pohtikaa: Miksi WhatsAppilla on niin paljon käyttäjiä? Olisiko luokassanne tai koulussanne mahdollista siirtyä käyttämään esimerkiksi Signalia? Mitkä seikat tukevat (tai eivät tue) tällaisen ratkaisun tekemistä? Mitä haasteita pikaviestisovelluksen vaihtamisessa olisi?

# Keinoja oman sekä muiden yksityisyyden ja tietoturvan lisäämiseen Internetissä (sivu 3/4)

## Yksityisyyden suojaaminen pilvitalennuspalveluja käytettäessä

Pilvipalveluja (joihin pilvitalennuspalvelut kuuluvat) käytetään monesti niiden luotettavuuden, helppokäyttöisyyden ja suhteellisen halpojen kustannusten vuoksi.

Pilvipalvelujen käyttämiseen liittyy kuitenkin varjopuolia, sillä lähtökohtaisesti pilvipalveluntarjoajilla on tekniset keinot: 1) muokata tai poistaa pilvessä toimivia palveluja 2) lukea, muokata, kopioida ja poistaa pilveen tallennettuja tietoja 3) seurata mitkä käyttäjät pilvessä toimivia palveluita käyttävät ja miten he niitä käyttävät. Pilvipalveluntarjoajat pystyvät myös antamaan haluamilleen kolmansille osapuolille pääsyn pilvessä toimiviin palveluihin ja siellä säilytettäviin tietoihin. Lainsäädäntö ja erilaiset sopimukset rajoittavat jossain määrin pilvipalveluntarjoajien toimintaa, mutta edellä mainitut tekniset keinot palveluntarjoajilla on silti lähtökohtaisesti käytössään.

## Tehtävänanto

- Selvittäkää ja kokeilkää miten voisitte säilyttää pilvitalennuspalveluissa (kuten Google Drive, OneDrive tai DropBox) ainoastaan salattuja tiedostoja [Cryptomatorin](#) avulla. Riittävän suojan saamiseksi (salauksen murtaminen tarpeeksi vaikeaa) käytettävän salasanan tulee olla todella vahva (vähintään 20 merkkiä). Salasanasta tulee olla myös useita varmuuskopioita.
- Asentakaa GnuPG (Windowsille [Gpg4win](#) ja Macille [GPG Suite](#)) tietokoneillenne ja sen jälkeen: 1) luokaa itsellenne avainpari 2) ottakaa avainparista varmuuskopiot 3) kokeilkää salata jokin tiedosto ja sen jälkeen purkaa salaus. Avainparia luodessa ohjelma salaa yksityisen avaimen käytössä olevalle tietokoneelle salasanan (passphrase) avulla. Linuxille ohjelma on jo usein valmiiksi asennettu. Valitettavasti ohjelman käyttöliittymä ei ole kovin hyvä, joten toivottavasti helpommin käytettäviä ohjelmia kehitetään lähitulevaisuudessa. GnuPG:tä voi myös käyttää viestien salaamiseen, jotka aikoo välittää jonkin muun ohjelman kautta.

# Keinoja oman sekä muiden yksityisyyden ja tietoturvan lisäämiseen Internetissä (sivu 4/4)

## Käyttöjärjestelmien ja ohjelmien vaikutus yksityisyyteen

Laitteisiin asennetuista suljetun lähdekoodin ohjelmista ja käyttöjärjestelmistä on hankala tarkalleen tietää, mitä tietoa kyseiset ohjelmat ja käyttöjärjestelmät lähettävät. Lähdekoodien ollessa suljettuja, joudumme usein luottamaan pelkästään ohjelmien tekijöiden sanaan siitä, mitä ohjelmat tarkalleen ottaen tekevät. Tämän lisäksi on tiedostettava, että käyttöjärjestelmien graafisten käyttöliittymien alla, on aina käynnissä monenlaisia eri prosesseja. Vaikka jokin tietoa lähettävä toiminto tai ohjelma näyttäisikin olevan kytkettyä pois graafisen käyttöliittymän perusteella, niin se ei välttämättä tarkoita sitä, etteikö kyseinen toiminto tai ohjelma olisi silti jollakin tavalla aktiivisena. Näistä syistä onkin aiheellista ottaa huomioon esimerkiksi tekijöiden toimintaan vaikuttavia taloudellisia kannustimia sekä tekijöihin kohdistuvia muita paineita, kun arvioimme eri ohjelmien luotettavuutta.

Esimerkkejä suljetun lähdekoodin käyttöjärjestelmistä ja ohjelmista ovat iOS, Windows, WhatsApp-sovellus, Instagram-sovellus, Snapchat-sovellus, Google Chrome -selain ja Google Play Store -sovellus. Android-käyttöjärjestelmän lähdekoodi on avoin, mutta käytännössä Android-puhelimet sisältävät useita suljetun lähdekoodin ohjelmia valmiiksi asennettuna.

Älypuhelimien käyttöjärjestelmät ja puhelimiin asennetut sovellukset lähettävät paljon erilaista tietoa niiden käyttäjistä. Windows 10 -käyttöjärjestelmä kerää myös runsaasti tietoa käyttäjän toimista, joten yksityisyyden kannalta voisi olla viisaampaa käyttää jotakin avoimen lähdekoodin Linux-käyttöjärjestelmää. Tietoturvapainotteinen [QubesOS-käyttöjärjestelmä](#) voi olla myös harkinnan arvoinen vaihtoehto.

## Tehtävänanto

- Kokeilkaa asentaa (tai ajaa asentamatta USB-tikulta) jollekin laitteellenne [Linux Mint MATE](#) -käyttöjärjestelmä. Linux Mint on käyttöliittymältään lähellä Windowsia.
- Tutustukaa Privacy Tools -sivuston [suositteluihin](#) mobiilikäyttöjärjestelmiin [GrapheneOS](#) ja [LineageOS](#). Jos teillä on käytössänne jokin vanha puhelin, johon talletettuja tietoja te ette enää tarvitse, niin voitte koettaa asentaa siihen toisen edellä mainituista käyttöjärjestelmistä.

# Muita tehtäviä Internetiin liittyen

1. Kuinka nopeasti viesti voi nopeimmillaan kulkea Suomesta Yhdysvaltoihin?
2. Mitä tarkoittaa kaksivaiheinen tunnistautuminen ja mitä hyötyä siitä on? Miten voit käyttää tätä suojakeinoa käyttämässäsi palveluissa? Selvitä miten TOTP teknisesti toimii.
3. Katso kuvaa [FUNET-verkosta](#). Kuinka montaa erilaista reittiä pitkin IP-tietoliikennepaketti voi kulkea Espoosta Joensuuhun, kun oletetaan että samasta paikasta ei saa kulkea kahdesti?
4. Selvitä kuinka monta erilaista IPv4-osoitetta on olemassa. Miksi niitä on juuri niin paljon? Perustele asia luokkakaverillesi matematiikan avulla.
5. Millä muilla tavoilla dataa voidaan siirtää kuin merenalaisilla optisilla kaapeleilla?
6. Mikä on URL?
7. Mitä kaikkea tietoa eri palvelut keräävät ja missä näitä tietoja säilytetään?
8. Tutustu Dijkstran algoritmiin ja tee siitä jokin toteutus valitsemallasi ohjelmointikielellä.
9. Miten World Wide Web ja Internet liittyvät toisiinsa? Mitä eroa niillä on?
10. Tutki traceroute/tracert -työkalun avulla mitä reittiä paketti kulkee osoitteisiin [www.facebook.com](http://www.facebook.com) ja [www.whatsapp.com](http://www.whatsapp.com). Mitä huomaat?
11. Minkälaista vahinkoa voi koitua henkilön maineelle, jos joku pahantahtoinen taho saa pääsyn hänen sosiaalisen median tunnuksiin, käyttämänsä pilvitalennuspalvelun tunnuksiin tai puhelimeensa tallennettuihin tietoihin?
12. Tutustukaa selaimien Web-työkaluihin (ohjeet: [Firefoxille](#), [Edgelle](#) ja [Chromelle](#)). Mitä asioita te voitte selvittää eri verkkosivuista näiden työkalujen avulla? Mitä varten selaimiin on rakennettu tällaiset työkalut?
13. Mitä ominaisuuksia on riittävän vahvalla salasanalla? Miten omia salasanoja tulee hallita? Miksei samaa salasanaa pidä käyttää useassa kriittisessä palvelussa, kuten rahaan, opintoasioihin tai maineeseen liittyvissä palveluissa?
14. Miten otat varmuuskopiot laitteillasi olevista tiedoista? Miten palautat varmuuskopiot? Missä (ja kuinka monessa) eri fyysisissä ja digitaalisissa paikoissa varmuuskopioita kannattaa säilyttää?
15. Mitä työkaluja käyttämällä voit salata yksittäisiä tiedostoja tai koko tiedostojärjestelmän? Mitä ovat yksityinen avain ja julkinen avain?

# Mitä on "yksityisyys" ja mitä hyötyä siitä voi olla? (sivu 1/2)

## Johdanto

[YK:n ihmisoikeuksien yleismaallisen julistuksen](#) 12. artikla kuuluu näin

*Älköön mielivaltaisesti puututtako kenenkään yksityiselämään, perheeseen, kotiin tai kirjeenvaihtoon älköönkä loukattako kenenkään kunniaa ja mainetta. Jokaisella on oikeus lain suojaan sellaista puuttumista tai loukkausta vastaan.*

## Tutkikaa

- Millä eri tavoilla käsitettä "yksityisyys" on yritetty määritellä tai kuvailla?
- Minkälaisia eri kannanottoja ja näkökulmia löydätte hakukoneiden avulla liittyen yksityisyyteen (hyödyt yksilölle tai yhteiskunnalle, haitat, teknologian vaikutus ja tulevaisuus)? Millä tavalla näitä kannanottoja ja näkökulmia on perusteltu?
- Mitkä eri lait säätelevät ihmisten oikeutta yksityisyyteen Suomessa ja Euroopassa?

# Mitä on "yksityisyys" ja mitä hyötyä siitä voi olla? (sivu 2/2)

## Pohtikaa ja keskustelkaa

- Mitä hyötyä yksityisyydestä voi olla yksilölle? Entä onko siitä yhteiskunnallista hyötyä? Entä voiko siitä olla haittaa?
- Millä tavalla eri laitteet ja verkkopalvelut vaikuttavat ihmisten yksityisyyteen? Laitteista esimerkkejä ovat älypuhelimet, tietokoneet ja kamerat. Verkkopalveluista esimerkkejä ovat verkkosivut, pikaviestipalvelut, pilvitallennuspalvelut, sähköpostipalvelut, nettitreffitpalvelut (esimerkiksi Tinder), videotoistopalvelut sekä sosiaalisen median palvelut.
- Mikä on ["Ei mitään salattavaa" -perustelu](#) ja millä eri tavoin tätä perustelua on kommentoitu?
- Minkä muiden oikeuksien, velvollisuuksien tai tarpeiden kanssa yksilön oikeutta yksityisyyteen on tasapainoteltava?
- Minkälaista olisi elää maailmassa tai valtiossa, jossa ei olisi oikeutta yksityisyyteen?

**Huomautus:** Teidän ei tarvitse nyt muodostaa "yksityisyydestä" omaa mielipidettä, vaan voitte vain tutustua erilaisiin näkökulmiin ja pohtia aihetta yhdessä.

# Tietojen kerääminen ja digitaalisen jalanjäljen muodostuminen

## Ennen keskustelua

Lukekaa aluksi kirjoitus *Tietojen kerääminen ja digitaalisen jalanjäljen muodostuminen*

<https://www.cs.helsinki.fi/group/linkki/materiaali/lumatikka3/tietojen-kerääminen-digitaalinen-jalanjalki.pdf>

## Pohtikaa ja keskustelkaa

- Mitä kaikkea tietoa (tai teihin mahdollisesti yhdistettävää tietoa) teistä tallentuu digitaalisessa muodossa, kun käytätte erilaisia laitteita, älypuhelinsovelluksia, verkkosivuja, pilvipalveluita, pelejä tai muita verkkopalveluita? Entä silloin kun fyysisesti vieraillette eri paikoissa tai tapahtumissa, tai kun käytte vaikkapa ostoksilla?
- Millä kaikilla tahoilla on pääsy näihin tietoihin, joita teistä on kerätty?
- Millä tavalla teidän oma toimintanne vaikuttaa siihen, mitä tietoja muista ihmisistä kerääntyy ja minkälainen digitaalinen jalanjälki muista ihmisistä muodostuu?
- Miten te voisitte soveltaa arjessanne seuraavalta sivustolta löytyviä työkaluja suojataksenne paremmin teidän omaa, sekä samalla lähipiirinne, yksityisyyttä: <https://www.privacytools.io/software/> ?
- Minkälaisia taloudellisia kannustimia teknologiayrityksillä voi olla kerätä käyttäjistään tietoja sekä muodostaa profiileja kerättyjen tietojen perusteella?
- Voisiko teistä kerättyä tietoa käyttää joskus tulevaisuudessa teitä vastaan tai teidän etujenne ajamiseksi? Miten?



# Suosittelualgoritmien hyödyt ja niihin liittyvät riskit

Katsokaa aluksi video [Suosittelualgoritmeista sekä niiden vaikutuksista](#). Tämän jälkeen pohtikaa seuraavia kysymyksiä:

- Mitkä käyttämistänne palveluista suosittelevat teille teitä varten yksilöityä sisältöä?
- Mihin suosittelualgoritmeja tarvitaan? Entä mihin niillä pyritään?
- Mitkä ovat suosittelualgoritmien päätoimintaperiaatteet?
- Mitä tarkoitetaan käsitteellä 'filterikupla'? Minkälaisia palveluja käyttäessä sellaiseen voi joutua? Entä minkälaisia vaikutuksia sellaiseen joutumisella voi olla?
- Onko esimerkiksi Facebookin tai YouTubeen käyttäjän mahdollista tietää, minkälaiset algoritmit ja ohjelmat hänelle tuottavat suosituksia?
- Minkälaisia tahattomia vääristymiä käyttäjien maailmankuvissa suosittelualgoritmit voisivat aiheuttaa? Olisiko niiden avulla mahdollista myös tahallisesti aiheuttaa vääristymiä?

# Avoimen lähdekoodin ohjelmat

## Selvittäkää

- Mitä on avoimen lähdekoodin ohjelmat?
- Mitä avoimen lähdekoodin ohjelmia te käytätte (ehkä tietämättänne)?
- Millä eri tavoin ja missä eri rooleissa avoimen lähdekoodin ohjelmien kehittämiseen voi osallistua?

## Pohtikaa ja keskustelkaa

- Onko joku teistä julkaissut omien projektiansa lähdekoodeja? Millä lisenssillä lähdekoodit on julkaistu? Missä palvelussa lähdekoodit on julkaistu?
- Mitä hyötyä voisi olla yksilölle käyttää avoimen lähdekoodin ohjelmia?
- Mitä yhteiskunnallista hyötyä voisi olla siitä, että suurempi osuus väestöstä olisi tietoisia avoimen lähdekoodin ohjelmista sekä käyttäisi niitä enemmän?
- Mitä hyötyä voisi olla yritykselle tai jollekin oppilaitokselle siitä, että he käyttäisivät enemmän avoimen lähdekoodin ohjelmia?
- Mitkä asiat voivat vaikeuttaa avoimen lähdekoodin ohjelmien käyttöönottoa ja hyödyntämistä yrityksissä tai oppilaitoksissa?