



**TAL  
TECH**  
IT KOLLEDŽ

# Loeng Vaba tarkvara hariduses

Edmund Laugasson  
IT Kolledž  
Infotehnoloogia teaduskond  
Tallinna Tehnikaülikool

Informaatika õpetamise  
konverents 2022  
13.-15.01.2022 Tartus



Kuvastik ja lisateave

# Juttu tuleb

- vaba tarkvaraga seotud olulisus, väärtused
- mehhaaniline vs loogiline metoodika
- metoodika näited õpetamisel
- eesmärk ja vahend informaatika õpetamisel
- informaatika õpetamine vaba tarkvara abil

# Vaba tarkvara olulisus, väärtused

Räägime vaba tarkvaraga seotud terminitest, olulisusest, väärtustest hariduses, eeskätt informaatika õpetamisel.

## Terminid, 1

**vaba tarkvara** (ka vabavara, privara) on tarkvara, mida saab ilma piiranguteta kasutada, kopeerida, uurida, muuta ning edasi jagada (vabadusastet):  
**vabadus 0:** kasutada tarkvara mistahes eesmärgil (mitta-äriline, äriline)  
**vabadus 1:** uurida lähtekoodi, seeläbi õppida, soovii korral muuta  
**vabadus 2:** võimalus aidata naabrit; levitada koopiaid  
**vabadus 3:** võimalus panustada kokkoonde - vabadus levitada koopiaid muudetud versioonina

**omanduslik tarkvara** (ka omandavara) on tarkvara, mis on saadaval raha eest või ka tasuta (raha maksmata), kuid on omanduslik, kuna kasutajatel ei ole vabadust seda kasutada, kopeerida, uurida, muuta ja levitada välja arvatud tarkvaratooja (tooja) lubatud juhtudel.

**tootjalukustus** on sõltuvus tootja poolt pakutavale tootele või teenustele suhtes läbi tehnoloogilise sõltuvuse tüübid, firmasised ja -omased, mõnikord ka päritolulist standardid, tehnilised normid, spetsifikatsioonid

## Terminid, 2

**informaatika** on info struktuuri, loomist, harkimist, töötlemist, lõigendamist, edastamist ning esitamist kaaltev teaduse ja tehniku haru. <http://viki.tu.ee/infid/opk/gp/11.htm>

**Digipädevus** on suutlikkus kasutada uuenevat digitehnoloogiat toimetulekks kiiresti muutvas ühiskonnas nii õppimisel, kodaniku tegutsedes kui ka kokkonnades suheldes; leida ja säilitada digivahendeid abi inri ning hinnata selle asjakohasust ja saadaväärsust; osaleda digitaalses süüboones, sh tekstide, pilvide, multimeediumide loomist ja kasutamisel; kasutada probleemilahenduseks sobivaid digivahendeid ja võrde, suhelda ja teha koostööd erinevates digikeskkondades; olla teadlik digikeskkonna ohtudest ning osata kaista oma privaatsust, isikuandmeid ja digitaalselt identiteeti; järgida digikeskkonnas samu moraalil- ja väärtuspõhimõtteid nagu igapäevaelus.

DigComp 2 (07.02.2016)

Sisuliselt määratletud EL kodanike digipädevusmodeliga DigComp. <https://ec.europa.eu/eu/er/en/digcomp/digital-competence-framework>

Lisateave <https://www.rigitehaja.ee/akt/29062014018>

## Vaba tarkvara väärtused

**Vaba tarkvara on eetilise sotsiaalne süsteem, mis austab igatihe vabadust.** Inimvääruste õpetamine on oluline ka hariduses: avatus, vabadus, etika, sotsiaalsus, ausus teiste vastu, seaduslik jagamine igal tasandil, üksteise sallimine.

4 vabadusastet annavad avatus, vabadus, võimaluse suhelda, austades üksteist viiuldusi tarkvara järele ja ka autorisust; aidates üksteist ka vajaliku tarkvaraga, teadmistega läbi avatud lähtekoodi, jne.

Vaba tarkvara litsentsid (GPL, j) annavad seadusliku õiguse nimetatud vabadusteks, sh üksteisele jagamine.

Jagamine hariduses: õpetaja, õppejuhud saab õppuritele jagada seaduslikult tarkvara, õppurid saavad ka ise alla laadida ja omakorda edasi jagada.

Lisateave: <https://www.gnu.org/education/>  
 GPL - <https://www.gnu.org/licenses/gpl.html>

## Vaba tarkvara olulisus hariduses, 1

Sõltuvuse ja lisakulutuste tekkimise vähimine läbi tootjalukustuse puudumise vabatarvaras.

Haridussuutuste üldjuhul ei ole ülearu raha. Kuigi raha ei ole kaugeltki mitte peamine, siis haridussuutuste jaoks ikkagi oluline. Kuigi tarkvara annetamisega võidakse rahaline pool kõrvaldada, siis tootjalukustust ja hilisemat pealesunnitud maksmist ei kõrvalda mingi. Pealesunnitud, kuna lähtest andmed on lukustatud omandavalistestest vormingutesse.

Vaba tarkvara kasutades olatakse hiljem tõttu, et sisemises vaba pealesunnitud sõltuvusest kasutada omandavara. See aitab parandada konkurentsivõimet tööturul läbi teenuste, toodete otavama omamirna, mis on võimalik saavutada kui ettevõttes rakendatakse vaba tarkvara. Lisaks rahale säästetakse ka aega ja ressursse.

Märkida teus ka pahaarvatud talenemete kahjude süülistel vähimire kuna vabatarvara korral on see peaaegu olematu. Süülistel ei leki töötajateks ega seeläbi tulenevat kahju. Samuti ei teki muud kahjusid, näiteks lunavara kahjud, töötuspuudust, jn.

Lisateave: <https://www.gnu.org/education/>

## Vaba tarkvara olulisus hariduses, 2

Vaba tarkvara abili IT-eadmist omandamine laiendab oluliselt silmaringi.

TalTechi IT Kolledži kogemuse kinnitab, et GNU/Linuxiga üliõpilased saavad oluliselt paremini IT-ülesannetega hakkama nii õppeainetes kui ka tööturul. See on ka üks põhjus, miks IT Kolledži õppetunni hinnatakse tööturul.

Silt ka ettepanek kasutada nii informaatika kui digipädevuste õpetamiseks kõikidel haridustasemetel põhiliselt tarkvarana vaba tarkvara. Omandavara tarkused onandatakse seeläbi oluliselt kergemalt. Haridussuutused ei pea süülistel vaeva nägema omandavara õpetamisega, kuna samade teadmistega saab seda ka kasutada. Kui aga vastupidiselt ettepanek omandavara ja seeläbi vaba tarkvara, siis on informaatika kui ka digipädevuste õppimiskurvi pikem, hilisem toimetulek kehvem. Teadusuringuid siin taga ei ole, kuid kohal on see vaele väga suur.

Lisateave: <http://tiny.cc/tarkvara-veebis-aruus>  
<https://www.gnu.org/education/>

## Vaba tarkvara olulisus hariduses, 3

Vaba tarkvara võimaldab õppida programmeerimist teiste kogemuste pealt.

Taru lähtekoodi avatusel on võimalik õppida selle pealt, kuidas teised tarkvararendajad on lähtekoodi kujundanud ja seeläbi tarkvara loonud. Kokkuvõttes on see inimkonna teadmus, hindamatu väärtusega kogemus, millest on võimalik kõigi huvilistel õppida. Seeures on võimalik ka teiste tarkvararendajatega kergemini suhelda. Seega oluline ei ole mitte niivõrd lähtekoodi avatus, kui selle ümber tekkiv sotsiaalne suhtus (Eric Steven Raymond).

**Kõrvalfokt on see, et tarkvara lähtekoodi saab ka kvaliteetsem, kui seda jälgib rohkem silmapaare (vt ka Linuxe seadus).** Ühtasi aitab lähtekoodi avatus ka vigu tarkvaras kergemini avastada ja neid ka parandada.

Lisateave: [https://en.wikipedia.org/wiki/Linux%27s\\_law](https://en.wikipedia.org/wiki/Linux%27s_law)  
[https://en.wikipedia.org/wiki/Eric\\_S\\_Raymond](https://en.wikipedia.org/wiki/Eric_S_Raymond)

# Terminid, 1

- **vaba tarkvara** (ka vabavara, priivara) on tarkvara, mida saab ilma piiranguteta kasutada, kopeerida, uurida, muuta ning edasi jagada (4 vabadusastet):

**vabadus 0:** kasutada tarkvara mistahes eesmärgil (mitte-äriline, äriline)

**vabadus 1:** uurida lähtekoodi, sellest õppida, soovi korral muuta

**vabadus 2:** võimalus aidata naabrit, levitada koopiaid

**vabadus 3:** võimalus panustada kogukonda - vabadus levitada koopiaid muudetud versioonist

- **omanduslik tarkvara** (ka omandvara) on tarkvara, mis on saadaval raha eest või ka tasuta (raha maksmata), kuid on omanduslik, kuna kasutajatel ei ole vabadust seda kasutada, kopeerida, uurida, muuta ja levitada välja arvatud tarkvaratootja (looja) lubatud juhtudel.

- **tootjalukustus** on sõltuvus tootja poolt pakutavate toodete või teenuste suhtes läbi tehnoloogilise salastatuse (suletud, firmasisesed ja -omased, mõnikord ka patenteeritud standardid, tehnilised normid, spetsifikatsioonid)

# Terminid, 2

**Informaatika** on info struktuuri, loomist, hankimist, töötlemist, tõlgendamist, edastamist ning esitamist käsitlev teaduse ja tehnika haru. <http://htk.tlu.ee/infdid/opik/ptk11.html>

**Digipädevus** on suutlikkus kasutada uuenevat digitehnoloogiat toimetulekuks kiiresti muutavas ühiskonnas nii õppimisel, kodanikuna tegutsedes kui ka kogukondades suheldes; leida ja säilitada digivahendite abil infot ning hinnata selle asjakohasust ja usaldusväarsust; osaleda digitaalses sisuloomes, sh tekstide, piltide, multimeediumide loomisel ja kasutamisel; kasutada probleemilahenduseks sobivaid digivahendeid ja võtteid, suhelda ja teha koostööd erinevates digikeskkondades; olla teadlik digikeskkonna ohtudest ning osata kaitsta oma privaatsust, isikuandmeid ja digitaalset identiteeti; järgida digikeskkonnas samu moraali- ja väärtuspõhimõtteid nagu igapäevaelus.

*DigComp 2.1 (07.02.2018)*

Sisuliselt määratletud EL kodanike digipädevusmudeliga DigComp.  
<https://ec.europa.eu/jrc/en/digcomp/digital-competence-framework>

*tulekul: DigComp 2.2*

Lisateave <https://www.riigiteataja.ee/akt/129082014018>

# Vaba tarkvara väärtused

**Vaba tarkvara on eetiline sotsiaalne süsteem, mis austab igaühe vabadust.**

Inimväärtuste õpetamine on oluline ka hariduses: avatus, vabadus, eetika, sotsiaalsus, austus teiste vastu, seaduslik jagamine igal tasandil, üksteise aitamine.

4 vabadusastet annavad avatuse, vabaduse, võimaluse suhelda; austades üksteise vajadusi tarkvara järele ja ka autorsust; aidates üksteist ka vajaliku tarkvaraga, teadmistega läbi avatud lähtekoodi, jne.

Vaba tarkvara litsentsid (GPL, jt) annavad seadusliku õiguse nimetatud vabadusteks, sh üksteisele jagamine.

Jagamine hariduses: õpetaja, õppejõud saab õppuritele jagada seaduslikult tarkvara, õppurid saavad ka ise alla laadida ja omakorda edasi jagada.

Lisateave: <https://www.gnu.org/education/>

GPL - <https://www.gnu.org/licenses/gpl.html>

# Vaba tarkvara olulisus hariduses, 1

**Sõltuvuse ja lisakulutuste tekkimise vältimine läbi tootjalukustuse puudumise vabatarkvaras.**

Haridusasutustel üldjuhul ei ole ülearu raha. Kuigi raha ei ole kaugeltki mitte peamine, siis haridusasutuste jaoks ikkagi oluline. Kuigi tarkvara annetamisega võidakse rahaline pool kõrvaldada, siis tootjalukustust ja hilisemat pealesunnituid maksmist ei kõrvalda miski. Pealesunnituid, kuna tähtsad andmed on lukustatud omandvaralistesse vormingutesse.

Vaba tarkvara kasutades ollakse hiljem tööturule sisenedes vaba pealesunnituid sõltuvusest kasutada omandvara. See aitab parandada konkurentsivõimet tööturul läbi teenuste, toodete odavama omahinna, mis on võimalik saavutada kui ettevõttes rakendatakse vaba tarkvara. Lisaks rahale säästetakse ka aega jm ressursse.

Märkida tasub ka pahavarast tulenevate kahjude sisuliselt vältimise kuna vabatarkvara korral on see pea olematu. Sisuliselt ei teki tööseisakuid ega sellest tulenevat kahju. Samuti ei teki muid kahjusid, näiteks lunavara kahjud, tööstusspionaaž, jm.

# Vaba tarkvara olulisus hariduses, 2

**Vaba tarkvara abil IT-teadmiste omandamine laiendab oluliselt silmaringi.**

TalTechi IT Kolledži kogemus kinnitab, et GNU/Linuxiga taustaga üliõpilased saavad oluliselt paremini IT-ülesannetega hakkama nii õppeainetes kui ka tööturul. See on ka üks põhjus, miks IT Kolledži lõpetanuid hinnatakse tööturul.

**Siit ka ettepanek kasutada nii informaatika kui digipädevuste õpetamisel kõikidel haridustasemetel põhilise tarkvarana vaba tarkvara.** Omandvara tarkused omandatakse seeläbi oluliselt kergemalt. Haridusasutused ei pea sisuliselt vaeva nägema omandvara õpetamisega, kuna samade teadmistega saab seda ka kasutada. Kui aga vastupidi teha: esmalt omandvara ja seejärel vaba tarkvara, siis on informaatika kui ka digipädevuste õppimiskurv pikem, hilisem toimetulek kehvem. Teadusuuringuid siin taga ei ole, kuid kohati on see vahe väga suur.

Lisateave: <http://tiny.cc/tarkvara-veebis-arvutis>

<https://www.gnu.org/education/>



# Vaba tarkvara olulisus hariduses, 3

**Vaba tarkvara võimaldab õppida programmeerimist teiste kogemuste pealt.**

Tänu lähtekoodi avatusele on võimalik õppida selle pealt, kuidas teised tarkvaraarendajad on lähtekoodi kirjutanud ja seeläbi tarkvara loonud. Kokkuvõttes on see inimkonna teadmus, hindamatu väärtusega kogemus, millest on võimalik kõigil huvilistel õppida. Seejuures on võimalik ka teiste tarkvaraarendajatega kergemini suhelda. Seega oluline ei ole mitte niivõrd lähtekoodi avatus, kui selle ümber tekkiv sotsiaalne suhtlus (*Eric Steven Raymond*).

**Kõrvalefekt on see, et tarkvara lähtekood saab ka kvaliteetsem, kui seda jälgib rohkem silmapaare (vt ka Linuse seadus).** Ühtlasi aitab lähtekoodi avatus ka vigu tarkvaras kergemini avastada ja neid ka parandada.

Lisateave: [https://en.wikipedia.org/wiki/Linus%27s\\_law](https://en.wikipedia.org/wiki/Linus%27s_law)

[https://en.wikipedia.org/wiki/Eric\\_S.\\_Raymond](https://en.wikipedia.org/wiki/Eric_S._Raymond)

# Mehhaaniline vs loogiline metoodika

## Siin teeme juttu mehhaaniline vs loogiline metoodikast õpetamisest kui ka õppimisest informaatikas ja seeläbi õpitud abituse vältimisest.

### Mõtlemine aeglaselt ja kiirelt

Isaadi postitused Daniel Kahnemani on 2011 saanud võimsa nimega: "Thinking, Fast and Slow" (Mõtlemine, aeglaselt ja kiirelt)

Kaks viis, kuidas aju moodustab mõtte:

- Süsteem 1**
  - kiire, automaatne, sagedane, emotsionaalne, stereotüüpne, alateadlik
  - Näide:**
    - üks kiireks, et objekt on lihtsalt kaugemal
    - 2+2=4
    - abi järele mõtlemiseks
    - huljalise suuna kindlaks tegemine
    - loogiline
    - kiirde
- Süsteem 2**
  - aeglane, pinguline, harkvaenev, loogiline, kalkuleeriv, teadlik
  - Näide:**
    - üks kiireks, et objekt on lihtsalt kaugemal
    - 2+2=4
    - abi järele mõtlemiseks
    - huljalise suuna kindlaks tegemine
    - loogiline
    - kiirde

### Mehhaaniline õpetamise metoodika

Mida tähendab mehhaaniline õpetamise metoodika?

- Õpetamine kasutatakse mehaanilist meetodit tegevise ja eesmärgi vahel.
- Näide:**
  - lihtsate formaadi muutmine LibreOffice Writeris valmis 5. põlvkonnale vastavalt, 10 väärt ülevaet
- Miks see on halb?**
  - Õmandatakse mehhaaniline seos, mida on vaja eesmärgi saavutamiseks teha. Näiteks, kui valitu asukohta muutub, on see teadmine kasutu.
- Miks see on hea?**
  - Kirja vis mõeldust koostamisel kirj panna mehhaaniline seos tegevise ja eesmärgi vahel.

### Loogiline õpetamise metoodika

Mida tähendab loogiline õpetamise metoodika?

- Õpetamine kasutatakse loogilist meetodit tegevise ja eesmärgi vahel.
- Näide:**
  - lihtsate formaadi muutmine LibreOffice Writeris valmis 5. põlvkonnale vastavalt, 10 väärt ülevaet
- Miks see on halb?**
  - Õmandatakse mehhaaniline seos, mida on vaja eesmärgi saavutamiseks teha. Näiteks, kui valitu asukohta muutub, on see teadmine kasutu.
- Miks see on hea?**
  - Õmandatakse loogiline seos, mida on vaja eesmärgi saavutamiseks teha. Kui valitu asukohta muutub, on see teadmine kasutu.

### Loogiline õpetamine tarkvara kasutamisel

- Isa rakendus, süsteemitarkvara kasutajalõpetel on oma kasutusloogika.
- Mõistlik on tuvastada see loogika ja seda õpetamisest ning õppimisest ka rakendades.
- Kuna tarkvara on erinev, siis ka kasutusloogika on erinev.
- Loogiline õpetamine, õppimine on -25% kulusum võrreldes mehhaanilise.
- Harjumused on suured muutused nõuab sõltuvalt -2x rohkem energiat võrreldes esialgse omandamisega. Loogiliselt tähendus on see siiski oluliselt lühem!

### Õpitud abitus, 1

Õpitud abitus on olukorras, kus inimene kogeb raskusi õppimisel või tegevuses, mis ta ei ole võimeline lahendama. See võib olla tingitud erinevatest põhjustest, mis võivad olla seotud õppimise või tegevusega.

Õn juhul, et erinevatel põhjustel võivad inimesed kogeda õpitud abitus sümptomeid ka muudatõppimise tegevuses (näiteks kooli õppimise kontrollimisel).

### Õpitud abitus, 2

Õpitud abitus võib tekkida ka siis, kui inimene kogeb raskusi õppimisel või tegevuses, mis ta ei ole võimeline lahendama. See võib olla tingitud erinevatest põhjustest, mis võivad olla seotud õppimise või tegevusega.

Õn juhul, et erinevatel põhjustel võivad inimesed kogeda õpitud abitus sümptomeid ka muudatõppimise tegevuses (näiteks kooli õppimise kontrollimisel).

### Õpitud abitus, 3

Õpitud abitus võib tekkida ka siis, kui inimene kogeb raskusi õppimisel või tegevuses, mis ta ei ole võimeline lahendama. See võib olla tingitud erinevatest põhjustest, mis võivad olla seotud õppimise või tegevusega.

Õn juhul, et erinevatel põhjustel võivad inimesed kogeda õpitud abitus sümptomeid ka muudatõppimise tegevuses (näiteks kooli õppimise kontrollimisel).

### Õpitud abitus, 4

Õpitud abitus võib tekkida ka siis, kui inimene kogeb raskusi õppimisel või tegevuses, mis ta ei ole võimeline lahendama. See võib olla tingitud erinevatest põhjustest, mis võivad olla seotud õppimise või tegevusega.

Õn juhul, et erinevatel põhjustel võivad inimesed kogeda õpitud abitus sümptomeid ka muudatõppimise tegevuses (näiteks kooli õppimise kontrollimisel).

# Mõtlemine aeglaselt ja kiirelt

israeli psühholoog Daniel Kahneman on 2011 saanud valmis raamatu:  
*"Thinking, Fast and Slow" (Mõtlemine, aeglaselt ja kiirelt)*

Kaks viisi, kuidas aju moodustab mõtted:

## süsteem 1

kiire, automaatne, sagedane, emotsionaalne, stereotüüpne, alateadlik

*Näiteks:*

- teha kindlaks, et objekt on teisest kaugemal
- $2+2=?$
- auto juhtimine tühjal tänaval
- heliallika suuna kindlaks tegemine
- hingamine
- kõndimine
- jne.

**mehhaaniline mõtlemine**

## süsteem 2

aeglane, pingutav, harvaesinev, loogiline, kalkuleeriv, teadlik

*Näiteks:*

- jooksuks ettevalmistumine
- heli tuvastamine
- säilitada tavapärasest kiirem kõndimiskiirus
- tähe A esinemiste arv tekstis
- kellelegi oma telefoninumbri andmine
- jne.

**loogiline mõtlemine**

# Mehhaaniline õpetamise metoodika

## Mida tähendab mehhaaniline õpetamise metoodika?

Õpetamisel kasutatakse **mehhaanilist** seost tegevuse ja eesmärgi vahel.

Näide:  
lehe formaadi muutmiseks LibreOffice Writeris valime 5.rippmenüü vasakult, 10.valik ülevalt

## Miks see on halb?

Omandatakse mehhaaniline seos, mida on vaja eesmärgi saavutamiseks teha. **Niipea, kui valiku asukoht muutub, on see teadmine kasutu.**

## Miks see on hea?

Kiire viis mõtlemist koormamata kirja panna mehhaaniline seos tegevuse ja eesmärgi vahel.

# Loogiline õpetamise metoodika

## Mida tähendab loogiline õpetamise metoodika?

Õpetamisel kasutatakse loogilist seost tegevuse ja eesmärgi vahel.

Näide:

leht on olemasolev osa dokumendist, mille muutmiseks kasutame rippmenüüd *Vormindus* ja leiame sealt leheküljestiili muutmise valiku

## Miks see on halb?

Peab mõtlema seose leidmiseks tegevuse ja eesmärgi vahel. Mõtlemine on raske (!).

## Miks see on hea?

Omandatakse loogiline seos, mida on vaja eesmärgi saavutamiseks teha. **Kui valiku asukoht juhtub muutuma, on see teadmine endiselt kasulik.**

# Loogiline õpetamine tarkvara kasutamisel

Igal rakenduse, süsteemitarkvara kasutajaliidesel on oma kasutusloogika.

Mõistlik on tuvastada see loogika ja seda õpetamisel ning õppimisel ka rakendada.

Kuna tarkvara on erinev, siis ka kasutusloogika on erinev.

Loogiline õpetamine, õppimine on ~25x tõhusam võrreldes mehaanilisega.

Harjumusel on suur jõud: muutmine nõuab üldjuhul ~2x rohkem energiat võrreldes esialgse omandamisega. Loogiliselt lähenedes on see siiski oluliselt lihtsam!

# Õpitud abitus, 1

**Õpitud abitus** on üksikisiku passiivsus, mis kujuneb korduval sunnitud viibimisel ebameeldivas olukorras, mida ta ei saa muuta. Sel viisil kujunenud õpitud abitus kandub üle ka uutesse olukordadesse, kus indiviidi vabadusaste on suurem. Termin võttis kasutusele USA psühholoog Martin Seligman (sündinud 1942). *Allikas: EE, 2006*

On leitud, et erinevalt loomadest võivad inimestel kujuneda õpitud abituse sümptomid ka mudelõppimise tagajärjel (nähes teist inimest kontrollimatus olukorras).

*Bandura A. (1986). Social foundations of thought and action: A social cognitive theory. Englewood Cliffs, NJ: Prentice-Hall.*

# Õpitud abitus, 2

Õpitud abitus tekib peamiselt kolmel moel.

**Esiteks**, kunagi on inimene proovinud võtta initsiatiivi ja saanud siis selle eest vastu näppe. Enam ta ei proovi.

**Teine viis** on läbiõppimine – inimene on intelligentne olend ja suudab õppida ka teistelt. Ehk et kui keegi teine on võtnud initsiatiivi ja saanud nina pihta, siis ka kõrvaltvaataja teeb oma järeldused.

**Kolmas viis** on eriti salakaval – kui inimene harjub toetuma otsustes ja tegemistes kellelegi ning lõpetab ise otsustamise.

Raimo Ülavere coaching

*Wikicell* sisaldab näidisjuhist, kuidas õpitud abitusest üle saada.

Seligman viitab negatiivsele kogemusele, mis õpitud abitust põhjustab (*kliinikum.ee*).

<b>Õpitud abitus (kognitiiv- käitumuslik)</b>	Seligman	Ebafunktsionaalne omistamisstiil	Negatiivsed sündmused	Negatiivsete sündmuste sisemised, stabiilsed ja globaalsed atributsioonid tekitavad ootuse, et tegevus ei anna tulemusi – aktiivsuse ja meeleolu langus.
---	----------	-------------------------------------	-----------------------	--



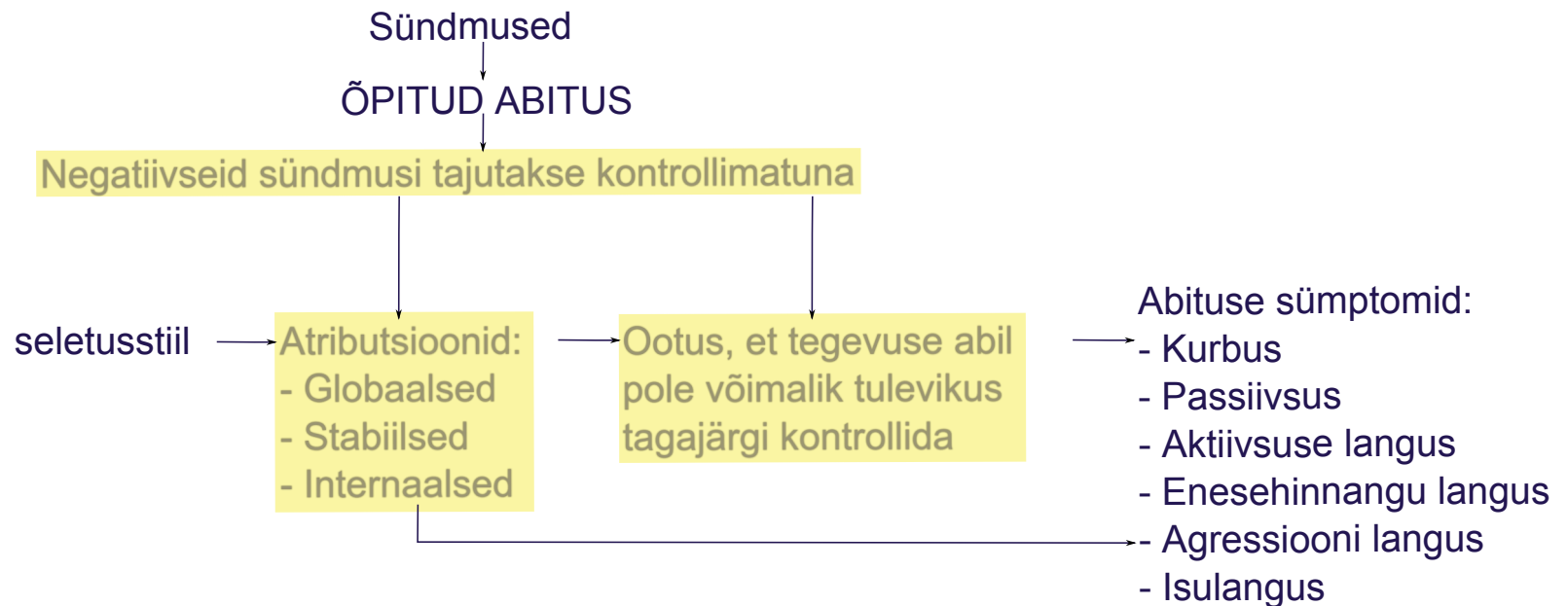
# Õpitud abitus, 3

Seligman kirjeldas õpitud abitust kui kogemuse kaudu omandatud ootust, et sündmusi on võimatu kontrollida ja tegevus ei vii eesmärkideni. Õpitud abitus mõjutab depressiooni teket koos seletusstiiliga.

[https://www.kliinikum.ee/psyhiaatriakliinik/lisad/ravi/Kliiniline\\_psyhholoogia/psychhol/ps\\_ph\\_seletused.htm](https://www.kliinikum.ee/psyhiaatriakliinik/lisad/ravi/Kliiniline_psyhholoogia/psychhol/ps_ph_seletused.htm)

## Õpitud abituse seos depressiooniga

Abi saab psühholoogilt!



# Õpitud abitus, 4

Kui teadmised on omandatud mehhaanilisel teel, siis on teise tarkvara kasutamisel ebaõnnestumine (negatiivne sündmus) kerge tulema, sest valikud ei pruugi sama koha peal asuda või sama nimega olla. Sellest tekib omakorda frustratsioon kuna ei ole õpetatud sellises olukorras toime tulema ehk siis loogiliselt mõtlema.

Selle tulemusena tekivad eespool nimetatud õpitud abituse sümptomid.

**Loogilise metoodika omandamisel on õpitud abitus kõrvaldatav:** enam ei ole lootusetuse tunnet kuna loogilise mõtlemise tulemusena on võimalik lahendus leida ja toime tulla. Frustratsiooni asemel tekib eduelamus.

Kuid loogiline mõtlemine on raske (*süsteem 2* tüüpi mõtlemine).



# Haridustasemed ja informaatika

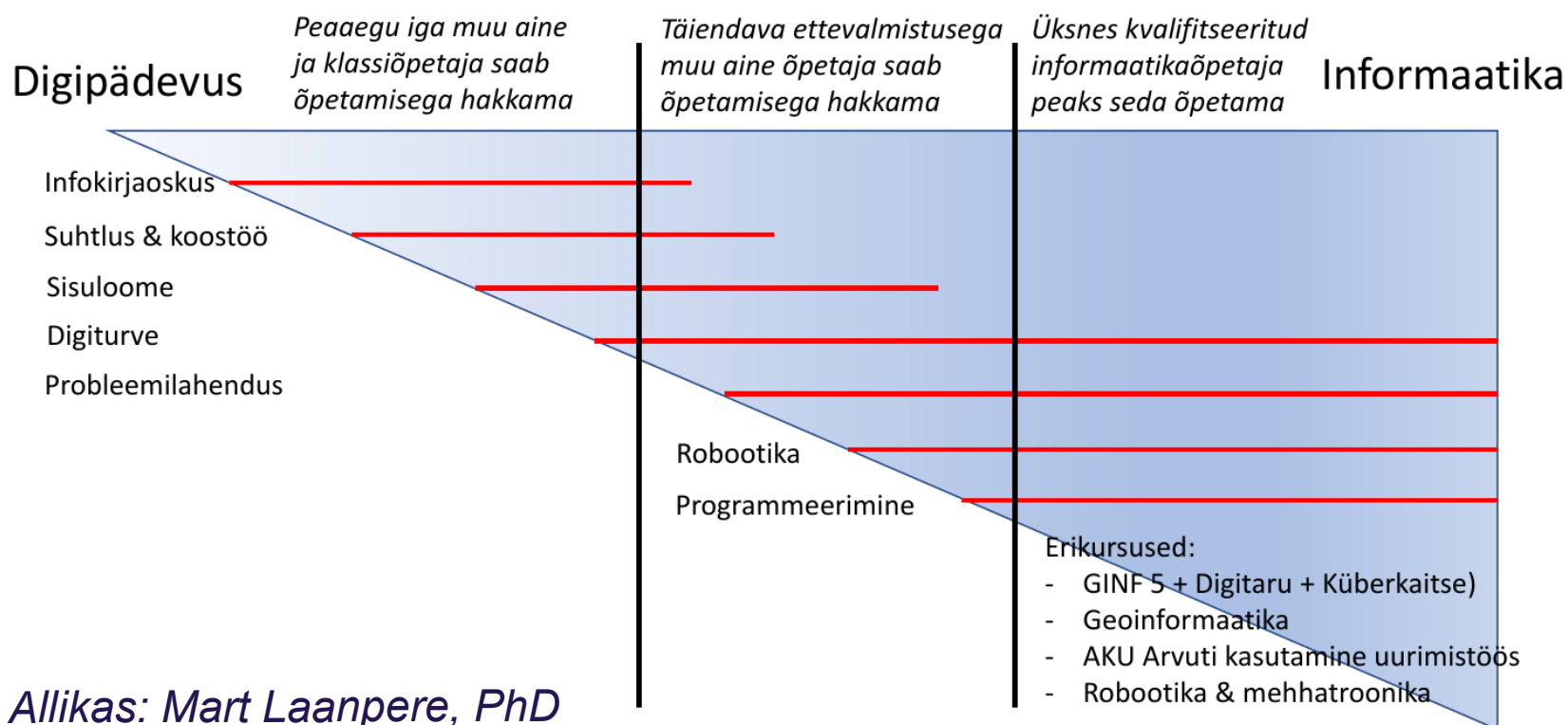
## Haridustasemed

- alusharidus
- põhiharidus (hariduse I tase)
- keskharidus (hariduse II tase)
- kõrgharidus (hariduse III tase)

## Põhikooliastmed

- I kooliaste – 1.–3. klass
- II kooliaste – 4.–6. klass
- III kooliaste – 7.–9. klass
- Gümnaasium** – 10.–12. klass

Kui alus- ja põhihariduses tegeletakse rohkem digipädevustega, siis kesk- ja kõrghariduses juba informaatika õpetamisega.



*Allikas: Mart Laanpere, PhD*

# Loogiline lähenemine kontoritarkvara puhul

**Kontoritarkvara** (tekstitöötlus, tabelarvutus, esitlus jm komplektis) on üsna ebaõnnestunud termin eesti keeles. Kui keegi teab paremat pakkuda, on see väga teretulnud.

**Millegi uue loomisel (mida veel ei ole) :**

- asetame kursori esmalt soovitud asukohta
- otsida lahendust rippmenüüst *Lisamine (Insert)*

NB! Failivahetust ei rajata impordile-ekspordile!

vt ka kuvastik

**Olemasoleva muutmisel (mis juba on olemas) :**

- märkida ära, teha aktiivseks olemasolev tekst; pilt vms (graafiline) objekt
- otsida lahendust rippmenüüst *Vormindus (Format)*

Mõnikord võib menüüdel olla teine nimetus sõltuvalt programmist või on need hoopis teisiti lahendatud. Näiteks LibreOffice Math'i puhul *Lisamine* menüü puudub, kuna selle asemel on eraldi vastav paneel.

Kontoritarkvara on kasutusel kõikidel haridustasemetel.

# Sisuloome tarkvara metoodika

Siin mõtleme mistahes tarkvara, millega saab mingit sisu luua. Olgu selleks siis tekst, pilt (vms graafika), heli, video, vms. Seetõttu on ka kontoritarkvara sisuliselt sisuloome tarkvara. Tihti enamuses haridusasutuses kasutatavat tarkvara on sisuloomeks kasutatav.

Üldiselt kehtivad samad põhimõtted, mis eespool kontoritarkvara puhul: leida see menüüde loogika ja rakendada seda.

Tulenevalt rakenduste erinevusest on ka menüüd jm kasutajaliidese osade käitumine erinev. Hüplikaknas näidatavat teksti ei pea õpetuses uuesti kirjutama.

Informaatikaõpetajatena tuleb meil leida see loogika, mida kasutajaliidese disainerid on rakendanud ja seda edasi õpetada.

Õpetus peaks olema koostatud selliselt, et õpitud teadmistega saab praktiliselt mistahes sisuloome rakendusega hakkama.

Sisuloome tarkvara on tihti kasutusel kõikidel haridustasemetel.

# Kontoritarkvara metoodika näited, 1

## Pildi lisamine LibreOffice Writeris

### Mehhaaniline

Vasakult 4.menüü ja sealt 3.valik ülevalt lisab pildi.

### Loogiline

Esmalt tekstikursor asukohta, kuhu pilti lisada soovime.

Seejärel rippmenüüst **Lisamine** valik **Pilt**.

## Lahtri vorminduse muutmine LibreOffice Calcis

### Mehhaaniline

Vasakult 5.menüü ja sealt 6.valik ülevalt avab lahtrivorminduse.

### Loogiline

Esmalt märkida lahter või lahtrid. Seejärel rippmenüüst

**Vormindus** valik **Lahtrid**.

## NB! Mõistlik oleks

**kontoritarkvaras konkreetset fonti nõudena mitte ette anda.**

*See kipub läbiv probleem olema paljudes kohtades. Pigem tuleks määrata nõuded*

*loetavusele, nt seriifidega või*

*ilma, eesti tähestikku täies mahus*

*toetav, vms. Microsofti fondid ei ole kaugeltki mitte igas arvutis!*

*Tihti konkreetse fondi nõude*

*täitmist ei kontrollita tegelikult. Nõudeid võiks olla nii vähe kui*

*võimalik ja nii palju kui vajalik.*

Saab lisada, et levinumad tegevused on saadaval ka nupudena tööriistaribal, kiirklahvina, hiire kiirmenüüs.

# Kontoritarkvara metoodika näited, 2

Siin näide mehhaanilisest lähenemisest.



Palju sõltub ka kooliastmest, kus õpetada. Kuid eriti oluline on algajaid loogiliselt õpetada, et hiljem nende õpetamine lihtsam oleks. Siin oleks piisanud hiiresündmuse õpetamisest.

Isegi kui ei ütle täpselt nii, nagu näites toodud, siis graafiliselt tihti seletatakse mehhaaniliselt, näidates sama teksti, mis muidu hüpikaknana ilmub, kui hiirekursorit valiku kohal hoida mõne hetke. Sellise õppematerjali tegemine on ajamahukam.

Selline seletus on paremini õigustatud siis, kui muud võimalust õpetamiseks tõesti ei ole. Kui tekib olukord, kus võidakse vääriti mõista ja selle vältimiseks ei ole muud loogilist võimalust. Tänapäeval kasutatakse sellisel puhul tihti animatsiooni, videot, mis on mõistlikum valik kui hakata ajamahukat graafilist materjali looma. **Siiski tasub igal juhul eraldi analüüsida loodud õppematerjali hilisema muutmise lihtsust ja sellest tulenevalt valida ka lahendus.**



# OSi graafilise kasutajaliidese näited

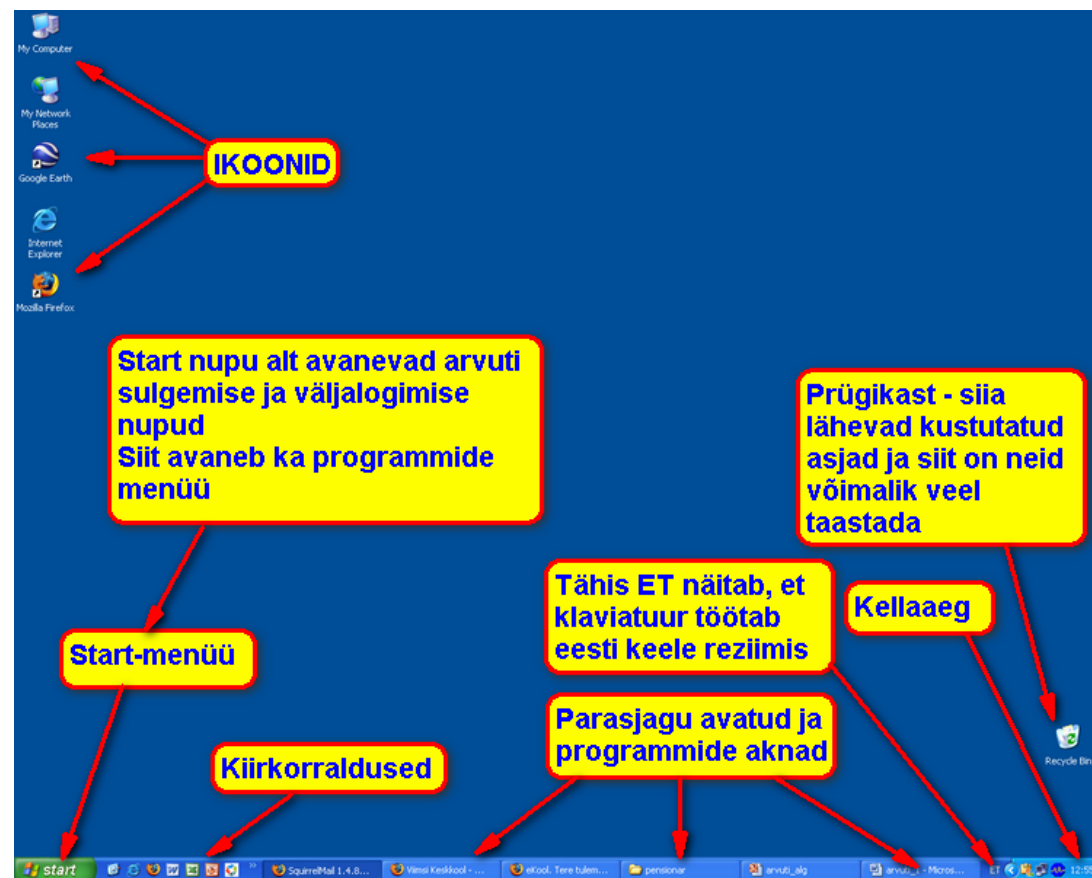
## Hiirekursori liigutamisel kaasnevad hiiresündmused

- sõltuvalt asukohast (mitte alati) kuvatakse selle peatamisel ka selgitus hüpikteatena, näiteks nupud, olekuriba -, paneeli elemendid, vms
- hiire kiirmenüüst leiab sõltuvalt asukohast levinumad tegevused

## OSi graafiline kasutajaliides

- töölaud: graafiline osa, füüsilise töölaua metafoor; samal ajal ka kataloog
- töölaua valitud servas asub paneel, mille osad on tavaliselt peamenüü, kiirkäivitusikoonid, tegumiriba, süsteemisalv. Sõltuvalt töölauakeskkonnast võib paneele olla ka mitu ja nende osi saab muuta. Mitme ekraani korral saab valida, kus paneele kuvatakse.

*Siin näites siis tekstina kirjeldatud loogiline ja pildil mehhaaniline õpetus.*



# Metoodika tähelepanekud, 1

Üldiselt siiski kõige hullem olukord ei ole. Sageli on sisuloomet võimaldava tarkvara juhised menüü - ja selle valikute nimetuste kaudu edasi antud.

Kui lisatakse ka illustreeriv kuvatõmmis ja tõstetakse oluline koht esile, siis ei ole veel midagi katki. Täna kasutatakse juba ka animatsiooni, videot keerulisemate asjade õpetamiseks ja see on hea.

Oluline moment saabub seletusega, mida sinna juurde räägitakse. Kui seletatakse ka kasutajaliidese loogikat, kuidas põhimõtteliselt asjad on paigutatud, siis on hästi.

Eriti oluline - kuidas abi saab, kui midagi ei õnnestu, kohe üles ei leia. **Kui on võimalik probleem sõnastada, siis on võimalik seda ka lahendada.** Tasub õpetada küsimusi esitama, seda mitte ainult sisuloome tarkvara õpetades. Õigesti küsides saab seda teha ka interneti otsimootoris ja sealtki abi leida lisaks tarkvara enda võimalustele.

# Metoodika tähelepanekud, 2

## Tarkvaravalikus tuleks õpetamisel eelistada vaba tarkvara. Miks?

- vabatarkvaraga kaasnevad vabadused, nt saab õppurile seaduslikult jagada
- tööks vajalik tarkvara on alati saadaval sõltumata rahalisest seisust, seda nii õppetöös kui hilisemas tööelus
- **vabavara teadmised toetavad omandvara kiiremat omandamist**  
(isiklik 20+ aastat õpetamist ja ka IT Kolledži õppejõudude kogemus)

Tööturul saavutatakse konkurentsieelis võrreldes omandvara kasutajatega:

- rahaline sääst: ei ole kaugeltki peamine tegur kuid siiski oluline
- ajaline sääst - Tallinna 2014/2015 katsetuse kogemus: haldusvõimekuse vahe ~3x
- närvide sääst: pahavara ei ole eriti levimisvõimeline tänu kaitseargumentidele
- tootjalukustuse puudumine



# Eesmärgi ja vahendi segi ajamine

Kuidas mitte segi ajada eesmärki ja vahendit informaatika õpetamisel?

**Lühidalt öeldes: tuleb õpetada põhimõtteid ja mitte konkreetset tarkvara!**

Näiteks tekstitöötlus ja mitte konkreetne rakendus, nt MS Word või LibreOffice Writer.

Seejuures on oluline kasutada loogilist metoodikat õpetamisel ja võimalusel vältida mehhaanilist. Näiteks kasutajaliidese loogikat kasutades.

Oluline on seejuures õpetamisel kasutada vaba tarkvara (sh failivormingud), et vabaneda omandvara sõltuvusest tootjalukustuse tõttu. Siis me ei suru tulevikus omandvara peale ka oma partneritele.

Hiljem tööturul olles on vabatarkvara kasutajatel konkurentsieelis kuna suudetakse sama asja teha säästlikumalt. Tarkvaralitsentsidele kuluva raha saab investeerida ettevõttesse (nt tootearendusse, inimestesse). Tublisti vähem probleeme on vabatarkvara puhul pahavaraga ja sellest tulenevate tööseisakute ning sellest tulenevate ärikahjudega.

# Sõnavara õpetamisel

Tihti kasutatakse rakenduse nimetust nimi- või isegi tegusõnana:

tekstitöötlus asemel öeldakse (MS) Word

tabelarvutus asemel öeldakse (MS) Excel

kuvastik asemel (MS) PowerPoint (*täpsustus: esitlus ehk esitlemine on tegevus*)

operatsioonisüsteem asemel öeldakse (MS) Windows

See põhjustab olukorra, kus inimestel tekib arusaam, et mingit muud tarkvara ei olegi olemas. Kinnistub väärteadmine, et kui räägime tekstitöötlustest, siis MS Word on ainus programm, millega seda normaalselt, tänapäevaselt teha saab. Kui siis keegi millegi muuga välja tuleb, siis suhtutakse sellesse põlgusega kuna see eristub haridusasutuses räägitust.

See piirab inimesi ja tekitab tõrksust teistsuguse tarkvara vastuvõtmisel.

See omakorda takistab ka teistsuguse tarkvara juurutamist, mis omakorda pidurdab nii inimeste kui ka ettevõtete, kokkuvõttes majanduse arengut.

Ilmselt ei ole see olukord, mida soovime. Kui ikka meie (Eesti) riik meile ka päriselt korda läheb. Tahaks uskuda, et läheb. Kuid varest sõnavarast saab see kõik alguse.

**Siit ka palve: palun räägime õigesti!**

# Neutraalne ja teadlik õpe, 1

Tulenevalt asjakohasest sõnavarast võiks informaatika õpetamisel järgida neutraalsusest ja teadlikkusest lähtuvat metoodikat:

- neutraalne operatsioonisüsteemi (OS), rakenduste suhtes: õpe on OSi-, rakendusteülene
- teadlik OSi, rakenduste valik, mitte inertsist, emotsioonist vms lähtuvalt

## Neutraalsus

Tihti kiputakse kirjutama sõnavara väärkasutusele sarnaselt, nt **OS = MS Windows** arvestamata, et on ka teisi operatsioonisüsteeme.

Selle teadvustamisega käib kaasas ka reaalne erinevate tarkvarade õpe ilma ühte või teist tarkvara halvustamata.

Ilmselt tuleb mingit tarkvara alusena kasutada ja vaba tarkvara sobib ideaalselt, kuna on oluliselt suurema kasuteguriga IT-alaste teadmiste omandamisel ehk siis osates vaba tarkvara, on teiste tarkvarade omandamine lihtsam. Seaduslikult vaba jagamisõigus õppuritele ei ole mitte vähem tähtis vaba tarkvara puhul.

# Neutraalne ja teadlik õpe, 2

## Teadlikkus

Kui valitakse konkreetne tarkvara, siis seda tehakse teadlikult ehk siis konkreetsete kriteeriumite, vajaduste alusel. Teadvustatakse ka valikute tagamaid. Näiteks MS Windows on küll mugav ja tore kuid kergesti pahavaraga nakatuv; privaatsus ja turvalisus on suure küsimärgiga ja tihti näiline. Valikut tehes antakse siis ka vastav teadlikkus, mis valitud tarkvaraga kaasas käib. Kui riskid teada, siis ehk on võimalik ka nendeks valmis olla ja ennetada niipalju kui võimalik. Omandvara puhul on see tihti oluliselt keerulisem tänu lähtekoodi suletusele - võimalik vaid tootja poolt.

## Veavaba tarkvara ei ole olemas

Ilmselgelt tuleb tunnistada, et veavaba tarkvara ei ole olemas. Seetõttu ei maksa arvata, et vabatarkvara puhul vigu ei esine. Kuid need on vabatarkvara puhul tänu lähtekoodi avatusele oluliselt lihtsam (sisuliselt võimalik) parandada. Tarkvara tuleb regulaarselt uuendada, mis on üks oluline ja seejuures lihtne võimalus vigu parandada ka kogukonna poolt. Vabatarkvara uueneb üldjuhul ka kiiremini võrreldes omandvaraga ning uuendamine ~3x vähem energiat nõudev. Siiski on vabatarkvara puhul mitmeid kaitseargumente, mis aitavad hõlpsamini vältida pahavara levikut ja sissehakkamisi.

**Kuid õpetamisel on siiski oluline õpetada põhimõtteid niipalju kui võimalik.**





# Haridusalase tarkvara valikud

Vaba tarkvara haridusvaldkonnale on palju.

*alternativeto.net* keskkonnas saab litsentsi alusel tarkvara filtreerida

## Mõned viited

Haridustarkvara *alternativeto.net*

Haridustarkvara *sourceforge.net*

Free Software Foundationi soovitusel  
*schoolforge.net*

Linuxlinks materjalid haridusele

Haridusliku vabavara projektid GitHubis

*50 Open Source Tools That Replace Popular Education Apps*

*A guide to free and open source education*

*opensourcelibs.com* hariduse projektid

*oss-watch.ac.uk* hariduse projektid

vabad AR, VR raamistikud

Open eLearning tarkvara

Lumi tarkvara (interaktiivne sisuloome)

Tarkvara veebis ja arvutis

## Kogukond

*http://teachingopensource.org/*

*https://pingviin.org/* (Eesti)

## Ajakiri

*The Journal of Open Source Education*

# Üldised põhimõtted

MKMi koosvõime raamistik  
European Interoperability Framework  
EIF2

Veebipõhine tarkvara on kasutatav sõltumata operatsioonisüsteemist.  
Sama versioon kõigile ühtviisi kättesaadav. Andmetel on muidugi ka veebis vorming.  
Omandvaralistest andmevormingutest tasub hoiduda.

Omanduslikest operatsioonisüsteemist sõltuvad rakendused saab vabavaralistes süsteemides käima vabavaraliste ühilduvuskihi või virtualiseerimise rakenduste abil.

Kui rakendusel pole väga tundmatuid sõltuvusi, saab selle ühilduvuskihi abil üldiselt tööle.  
Keerulisemad rakendused vajavad virtualiseerimist (virtuaalne arvuti).

Näiteks MS Windowsi vaba analoog on *ReactOS*. Vaatamata sellele, et arendusfaasis, saab siiski kasutada. Tasub siiski testida eelnevalt, kas soovitud valikus ja tasemel töötab.



(analoogid)



(analoogid)



# Kasutatavus alushariduses, 1

Alushariduse riiklik õppekava <https://www.riigiteataja.ee/akt/12970917?leiaKehtiv>

Arvuti koolieelses lasteasutuses ei ole reegel, kuid siiski kasutatakse.

Näiteks otsing <https://duckduckgo.com/?q=arvuti+lasteaias> (vastav kursuse TLÜs)

Valdavalt on tegemist veebipõhiste materjalidega (sh erinevad mängud), mida saab teinekord ka välja printida. Või on tegemist juhistega, kuidas midagi valmistada.

Tarkvara poolest kasutatakse rakendusi mängimiseks, joonistamiseks, nuputamiseks, jne. Ka näiteks vaadatakse multifilme (YouTube jm).

**Seda kõike saab teha ka vabatarkvaraga või veebipõhiselt.**

# Kasutatavus alushariduses, 2

## Mida on rakenduste poolelt kasutatud?

- MS Office 365 (veebipõhine) dokumentide loomiseks
- Scratch programmeerimise õpetamiseks lastele
- MS Paint, Paint.net joonistamiseks
- Xn View, MSO Picture Manager piltide töötlemiseks
- Autodesk SketchBook värvimise piltide joonistamiseks

## Vabatarkvaralised rakenduste analoogid?

- dokumentide loomine: LibreOffice, Nextcloud
- kuvastikud: Inkscape + Sozi (Prezi vaba analoog)
- animeerimine: Inkscape + Sozi, Synfig Studio, Blender, jt.
- programmeerimine: Scratch, Snap!, EduBlocks, jt
- pildid värvimiseks, joonistamine: Inkscape, Krita, GIMP, jt
- üldiselt on heli, pilt, video vabavaras hästi esindatud

*alternativeto.net* võimaldab litsentsi, operatsioonisüsteemi, ning teiste omaduste alusel filtreerida rakendusi

## Vabavara viited

LibreOffice

Inkcape Sozi

animeerimise tarkvara

maalimise tarkvara

pilditöötlustarkvara

fototöötlustarkvara

TupiTube StickyPy

Blender (analoogid)

Krita (analoogid)

TuxPaint TuxMath TuxType

GCompris KDE Edu

Childsplay Nextcloud

Scratch (analoogid)

Projektinäide Keralast, India

# Kasutatavus põhihariduses, 1

<https://courses.cs.ut.ee/t/digiopik/>

## Mida on rakenduste poolelt kasutatud?

- MS Office (nii veebis kui arvutis)
- heli-, pildi-, videotöötlus:
- MS Windowsi-põhine tarkvara
- programmeerimine:
- Visual Studio, Visual Basic, VBA, jt
- robotika: LEGO, Arduino, Raspberry Pi, jt

## Vabatarkvaralised rakenduste analoogid?

- dokumendid, pilvsalvestus: LibreOffice, Nextcloud
- heli, muusika: Audacity, Tenacity, LMMS, Ardour, MuSE, Rosegarden, MuseScore, Lilypond, jt
- video: Kdenlive, Shotcut, Openshot, Pitivi, Avidemux, LosslessCut, Cinelerra, LiVES, Handbrake, Natron, Olive, jt
- küljendamine: Scribus

Tihti võidakse sama tarkvara ka gümnaasiumis jm kasutada.

## Vabavara viited

kontoritarkvara                      Scribus  
 helitöötlustarkvara  
 helisalvestustarkvara  
 muusika loomine, noodid  
 graafikatarkvara  
 3D-modelleerimine  
 3D-printimine  
 CAD-tarkvara  
 LinuxCNC  
 CAD/CAM/CAE-tarkvara  
 videosalvestustarkvara  
 videotöötlustarkvara  
 multimeediatarkvara  
 süsteemitarkvara, riistvara  
 tarkvaraarendus  
 visuaalne programmeerimine  
 Python IDE rakendused  
 IDE rakendused                      NXT Python  
 NXT-G analoogid                      LiNXT  
 ev3dev    Lejos (Java)

# Kasutatavus põhihariduses, 2

## Vabatarkvaralised rakenduste analoogid?

- programmeerimine: LibreOffice Basic, Thonny, Eclipse, Geany, Notepad3, Notepad++, Code::Blocks, KDevelop, Atom, jt
- LEGO robotika:  
NXT: NXC/NBC  
EV3: ev3dev - Python, MicroPython, Java, Go, C++, C, Prolog, Vala, Genie, Rust, Ruby, Perl, jne
- robotika: Arduino, Raspberry Pi juba on vabavara  
ROS - Robot OS

Tihti võidakse sama tarkvara ka gümnaasiumis jm kasutada.

## Vabavara viited

- Lego NXT, Bluetooth and Linux (PDF)*
- Using Lego Mindstorms NXT with Ubuntu Linux*
- NBC Ubuntu Linuxile*
- NBC Arch Linuxile*
- NBC teistele UNIXilaadsetele OSidele*
- ROS (Robot OS)
- eLinux wiki artiklid, videod
- Robotics Software For Linux (2011)*
- Drone and Robot Software for Linux (2019)*
- robotikatarkvara sourceforge.net'is
- mBlock (veebis)
- Alice'i vabad analoogid
- robot Edison (veebipõhine tarkvara)

# Kasutatavus gümnaasiumis, 1

<https://projektid.edu.ee/pages/viewpage.action?pageId=81365502>

*Gümnaasiumi informaatika valikkursused (e-koolikott)*

*Digitaalne ohutus (e-koolikott)*

Informaatika ainekava koosneb viiest valikkursusest ja digilahenduse arendusprojektist:

- Programmeerimine (põhimõisted, koodikirjutamise alusoskused)
- Tarkvaraarendus (edasijõudnute tase, tarkvaraarenduse projekt)
- Kasutajakeskne disain ja prototüüpimine
- Tarkvara analüüs ja testimine
- Digiteenused (alusteadmised IS'dest, IS haldus, riskide ja intsidentide haldus)
- DigiTaru ehk digilahenduse arendusprojekt.

Lisaks võib olla:

- geoinformaatika
- AKU (arvuti kasutamine uurimistöös)
- robotika, mehhatroonika
- 3D-modelleerimine
- küberkaitse



# Kasutatavus gümnaasiumis, 2

## Küberkaitse

privaatsed otsimootorid, 1      DuckDuckGo      Startpage      Gibiru      Searx      SearchEncrypt  
 privaatsed otsimootorid, 2      WolframAlpha      Qwant      PrivacyWall      Peekier      MetaGer      Mojeek  
 privaatsed otsimootorid tumeveebis  
 turvatarkvara, 1      *Hacker howto* (vt FAQ 11)  
 turvatarkvara, 2      *Kuidas saada häkkeriks* (vt KKK 11)

## Geoinformaatika

GIS vabavara 1  
 GIS vabavara 2  
 Kaarditarkvara  
 Kaarditeenused

## AKU (arvuti kasutamine uurimistöös)

LibreOffice      bibliograafia  
 mõistekaardid 1      projektihaldus 1      Posterazor  
 mõistekaardid 2      projektihaldus 2      (analoogid)  
                          graafikatarkvara, 1      teadmushaldus  
                          graafikatarkvara, 2      viidete haldus

# Kasutatavus gümnaasiumis, 3

## GINF5, DigiTaru

pip'i paigaldus Ubuntu

Ubuntu wiki Pythoni artikkel

pip'i paigaldus Arch Linuxis

archinstall

Python Arch Linuxis

*Python package guidelines*

PySimpleGUI pip'i kaudu

PySimpleGUI Arch Linuxile

POV-Ray (analoogid)

koodiõppimise rakendused

disainitarkvara

UML-modelleerimine

voodiagrammid

diagrammiredaktorid

prototüüpimine

kasutajaliidese disain

kasutajakogemuse disain

graafiline disain

vabakäejoonis

projektijuhtimine

tarkvaraarendus

EDA - *Electronic Design Automation*

SCADA - *Supervisory control and data acquisition*

HMI - *Human Machine Interface*

## Robotika, mehhatroonika

NXT-G analoogid                      LiNXT

ev3dev                      Lejos (Java)

robotikatarkvara

EDA tarkvara

SCADA tarkvara

HMI tarkvara

bioinformaatika tarkvara

automatiseerimise tarkvara

simulatsioonitarkvara                      (valik 2)

testimine ja mõõtmine

ROS (*Robot Operating System*)

## 3D-modelleerimine

SketchUpi vabad analoogid

Solid Edge'i vabad analoogid

3D-modelleerimise tarkvara, 1

3D-modelleerimise tarkvara, 2

# Kasutatavus kõrg-, kutsehariduses

Ülikoolis on juba tegemist spetsialisti tasemel koolitusega. Siin ei räägi me tegelikult mitte enam informaatika vaid juba spetsiifilisema teadmise õpetamisest, mis siiski toetub informaatikas õpitud teadmistele. Sama on tegelikult ka kutsehariduses.

Üldjoontes on kontori- ja spetsiifiline tarkvara kasutusel ka siin, võrreldes põhihariduse ja gümnaasiumiga.

Erinevus tuleb sisse teaduse poolelt, mida ülikoolis kõrgemal tasemel tehakse. Mõned rakendused on kasutatavad ka juba AKU kursusest gümnaasiumis.

## Robotika

*Robotics Kits for Adults*

*Top 5 Advanced Robotics Kits*

*Raspberry Pi Robots Kits for Beginners*

## Teadus

andmete visualiseerimine

tehisintellekt

bioinformaatika

andmeanalüüs, statistika, 1

andmeanalüüs, statistika, 2

kvalitatiivne andmeanalüüs

kvantitatiivne andmeanalüüs

TeX/LaTeX tarkvara

teadmushaldus

viidete haldus

tulemusrakendused, 1

tulemusrakendused, 2

tulemusrakendused, 3



IT Kolledž  
Raja 4C  
12616 Tallinn, Eesti  
tel +372 628 5800  
[taltech.ee/itcollege](http://taltech.ee/itcollege)



Kuvastik ja lisateave

**TAL  
TECH**  
IT KOLLEDŽ