

TARTU ÜLIKOOL  
Arvutiteaduse instituut  
Informaatika õppekava

**Kevin Kekki**

**Programmeerimise kursuse  
silmaringimaterjalide täiendamine**

**Bakalaureusetöö (9 EAP)**

Juhendaja: Reimo Palm

Tartu 2022

## **Programmeerimise kursuse silmaringimaterjalide täiendamine**

### **Lühikokkuvõte:**

See bakalaurusetöö kirjeldab, kuidas täiendati Tartu Ülikooli kursuse „LTAT.03.001 Programmeerimine“ lisamaterjale ning pakub ideid, kuidas saaks neid edaspidi tõhusamalt rakendada. Lisamaterjalide eesmärk on laiendada üliõpilastele informaatikaalast silmaringi. Töö raames loodi lisamaterjalid võistlusprogrammeerimise, veebisisu parsimise, heli- ja pilditöötuse, kompileerimise, kodeerimise, krüpteerimise ja suurte andmehulkade töötlemise kohta. Materjalide tõhusamaks kasutamiseks pööratakse kursusel edaspidi lisamaterjalidele rohkem tähelepanu pöörata ja uuritakse, kuidas õpilased neid kasutavad ning mida neist arvavad.

### **Võtmesõnad:**

Programmeerimine, õppematerjalide koostamine, lisamaterjalid

**CERCS:** P175 Informaatika, süsteemiteooria; S270 Pedagoogika ja didaktika

## **Supplementing Additional Learning Materials for the Programming Course**

### **Abstract:**

This paper describes how a new set of supplementary learning materials was created for the University of Tartu course „LTAT.03.001 Computer Programming“ and offers proposals regarding their utilisation in the future. The purpose of the additional materials is to broaden the knowledge of students in the field of informatics. The created materials encompass topics such as competitive programming, web content parsing, sound and image processing, code compiling, encoding, encryption and big data processing. In order to use the additional learning materials more effectively, the course lecturers will bring the students' attention to these materials and examine how students use them and how they feel about them.

### **Keywords:**

Introductory programming course, learning materials creation, additional learning materials

**CERCS:** P175 Informatics, system theory; S270 Pedagogy and didactics

## Sisukord

1.	Sissejuhatus .....	4
1.1	Töö definitsioonid ja kontekst .....	4
1.2	Kursuse sisu uuendamine .....	5
1.3	Silmaringimaterjalide olulisus.....	5
2.	Materjalide koostamise metoodika .....	7
2.1	Varasemad õppematerjalide koostamise põhimõtted .....	7
2.2	Eeldatav sihtgrupp .....	8
2.3	Valitud põhimõtted .....	8
3.	Uued materjalid .....	9
3.1	Uute teemade valik .....	9
3.1.1	Veebisisu parsimine .....	9
3.1.2	Võistlusprogrammeerimine .....	9
3.1.3	Pilditöötlus .....	10
3.1.4	Helitöötlus ja -süntees .....	11
3.1.5	Kodeerimine ja krüptograafia .....	12
3.1.6	Pythoni siseehitus .....	12
3.1.7	NumPy massiivid .....	13
3.2	Uus peatükkide järjekord.....	14
3.3	Ülesanded ja enesekontrollitised .....	16
4.	Silmaringimaterjalide tulevik.....	17
4.1	Materjalide kasutamine ja sihtgrupp .....	17
4.2	Väljajäänud teemad .....	17
4.3	Teiste kursuste silmaringimaterjalid.....	19
5.	Kokkuvõte .....	20
6.	Viidatud kirjandus .....	21
	Lisad .....	23

## 1. Sissejuhatus

Tartu Ülikooli informaatika õppekava kursuse „LTAT.03.001 Programmeerimine“ [1] jaoks on koostatud lisamaterjalid (edaspidi vastavalt programmeerimiskursus ja silmaringimaterjalid), mida õpilastel on võimalik iseseisvalt uurida [2]. Kursuse jooksul jõuavad seda teha vähesed või uuritakse ainult mõnda materjali kursuse raames loodava projekti jaoks. Samas ongi materjalid mõeldud pigem õpilastele, kellele kursuse teemad juba mingil määral selged on, aga sellised õpilased suunatakse eeleksamile. Kui eeleksam sooritatakse tulemusega A, siis ei pea õpilane kursusel osalema. Seega ei puutu nad tõenäoliselt kokku silmaringimaterjalidega või ei tunne põhjust neid uurida.<sup>1</sup>

Programmeerimiskursuse uuendamise seotud suletud koosolekul otsustati, et kursuse sisu tuleks uuendada. Üks ettepanek oli koondada põhi- ja silmaringimaterjalid ühte kohta, ning leiti, et iga nädala õppematerjali juurde võiks kuuluda vähemalt üks silmaringiteema. Programmeerimiskursusel on 14 nädala jagu õppematerjale ning 6. ja 12. nädalal toimub kontrolltöö. Silmaringimaterjalidega on kaetud ainult 8 nädalat. [2]

Selle töö eesmärk on täiendada olemasolevaid silmaringimaterjale, nii et iga nädala põhimaterjali kohta oleks üks silmaringimaterjal. Lisaks pakutakse mõtteid selle kohta, kuidas võiks silmaringimaterjale tulevikus paremini kasutada.

Selle töö teises peatükis kirjeldatakse materjalide koostamise põhimõtteid ja metoodikat. Kolmandas peatükis antakse ülevaade koostatud silmaringimaterjalidest ja selgitatakse teemade valikut. Sama peatüki teises pooles kirjeldatakse kursuse materjalide uut järjekorda. Järgmises peatükis arutletakse, mida silmaringimaterjalidega edaspidi teha võiks. Sellele järgneb kokkuvõte.

### 1.1 Töö definitsioonid ja kontekst

Tartu Ülikooli kursusel „LTAT.03.001 Programmeerimine“ õpetatakse programmeerimise põhitõdesid programmeerimiskeele Python abil. See **programmeerimiskursus** toimub iga aasta sügissemestril ning osalejaid on üldjuhul mitmelt erialalt, tavaliselt on neid üle 300. Kursuse 8. – 15. nädalal tuleb osalejatel paaristööna teha projekt vabalt valitud teemal. Projektiks tuleb koostada Pythoni programm. [1]

---

<sup>1</sup> Info selgus kursuse õppejõududega vesteldes.

**Silmaringimaterjalide** all mõistetakse siin töös kirjalikke veebipõhiseid lisaõppematerjale. Nende läbimine ei ole kursusel osalejatele kohustuslik. Need peaksid aitama laiendada huviliste informaatika- ja programmeerimisalast silmaringi ja tutvustama teemasid Pythoni abil.

## **1.2 Kursuse sisu uuendamine**

Töö on seotud programmeerimiskursuse materjalide uuendamisega. Siiani on kursuse õpikuna kasutatud Aivar Annamaa kirjutatud programmeerimise õpikut ja selle teemajärjekorda [3]. 2022. aasta sügissemestriks luuakse kursusele eraldi õpik ja uus teemade järjekord. Muuhulgas lisatakse põhiteemade hulka ka objektorienteeritud programmeerimine.

Mahuliselt on programmeerimiskursusele silmaringimaterjale loodud 8 nädala jagu. Olemasolevad silmaringimaterjalid käsitlevad Pythoni standardteeki ja mooduleid, rakendusliideseid, regulaaravaldisi, SQL-andmebaase, veebirakenduste loomist raamistiku Flask abil, objektorienteeritud programmeerimist, graafiliste mängude loomist teegi PyGame abil ja erinevaid keerulisemaid Pythoni võimalusi ja keelekonstruktsioone. Kuna uuel õppekaval on objektorienteeritud programmeerimine üks põhiteema, tuleks samateemaline silmaringimaterjal eemaldada. Järelikult tuleb kokku koostada 7 uut silmaringimaterjali, et neid jätkuks 14 nädala jaoks. [2, 4]

## **1.3 Silmaringimaterjalide olulisus**

Kursuse põhimaterjalid õpetavad programmeerimise põhilisi tahke ja selgitavad üldiselt, mille jaoks neid kasutatakse. Lisamaterjalid pole kursuse läbimise jaoks kohustuslikud. Selle kursuse kahte liiki lisamaterjale nimetatakse omakorda lisalugemisteks ja silmaringimaterjalideks.

- Lisalugemine on lühem materjal, mis täiendab vastavat põhipeatükki. Seal kirjeldatakse tavaliselt peatükiga seotud täiendavaid programmeerimise kontseptsioone või Pythoni võimalusi, mille tundmine võiks kasulik olla.

- Silmaringimaterjal võib olla pikem ja kaudsemalt seotud nädala põhiteemaga. Seal on eraldi ära märgitud eelnevad põhipeatükid, mis peavad olema läbitud ja mõnel juhul järgnevad peatükid, millega võiks põgusalt tutvuda, et materjali paremini mõista. Silmaringimaterjal tutvustab mõnda informaatikaga seotud või programmeerimisega seotavat teemat. Materjalis tuuakse praktilisi näiteid, kus ja kuidas mingeid programmeerimisvõtteid või andmestruktuure kasutatakse.

Silmaringimaterjalid on kursuse juures olulised mitmel põhjusel. Olemasolevate silmaringimaterjalide autor mainib oma töös [4], et materjalid aitavad kursuse projekti jaoks teemat valida ja neist on kasu selle realiseerimisel. Samuti aitab see teha aine läbimise huvitavamaks õpilastele, kelle õppekavas see kohustuslik on, kuid kellele on paljud kursuse teemad juba selged. Materjalide kaudu on õpilastel ka juba võimalik saada ettekujutus oma õppekava ülejäänud ainetest. Erinevate lisamaterjalidega tutvudes on võimalik otsustada, millised teised informaatika eriala kursused võiksid neid huvitada.

Lisaks aitavad materjalid õppuritel üsna konkreetselt aru saada, kuidas saab programmeerimisega lahendada praktilisi probleeme. See võib aidata aru saada ja leida võimalusi, kuidas programmeerimisoskust oma erialal või hobi tarvis rakendada. Materjalide lugejad saavad ise näidete abil keerukamaid programme koostada. Kui õpilased saavad aru, et nad suudavad juba olemasolevate oskustega midagi huvitavat või praktilist teha, oskavad nad õpitut paremini väärtustada. See võib neid innustada edasi õppima.

Silmaringimaterjalid võivad kirjeldada mingi teema tausta. Nii on kursuse osalejatel võimalik lisaks programmeerimisoskusele saada ka teisi temakohaseid teadmisi või laiendada oma silmaringi. See muudab õppimiskogemuse vaheldusrikkamaks.

## 2. Materjalide koostamise metoodika

### 2.1 Varasemad õppematerjalide koostamise põhimõtted

Kursuse veebipõhiste õppematerjalide koostamise üldised põhimõtted tuleks eelnevalt läbi mõelda. Iraklis Paraskakis ja Thanos Hatziapostolou kirjeldavad oma uurimistöö ettekandes [5], kuidas Kreeka Avatud Ülikool otsustas oma arvutiteaduse sissejuhatava kursuse üles ehitada nii, et kaardistas kõigepealt kursuse sihtrühma ja valis selle põhjal sobivaimad õpetamismeetodid. Näiteks arvestati elustiiliga, peamiselt sellega, et enamik kursusel osalejaid on täiskasvanud, kes õpivad töö kõrvalt ja on väga erineva tasemega. Seejärel pandi kirja kursuse olulised pedagoogilised printsiibid ning loodi nende põhjal interaktiivne veebiõpik. Jälgiti, et sisu oleks mitmekesine, kohe täies mahus uuritav, interaktiivne ja välja oleks toodud selged õpiesmärgid, peatükid sisaldaks palju kokkuvõtteid ja poleks liiga infoküllased. Ida-Illinoisi Ülikooli 1997. aasta juhend [6] kirjeldab, et lisaks tuleks tähelepanu pöörata ka materjalide vormistusele, kasutajaliidesele ning õppijate huvi ja tähelepanu hoidmisele. Selleks võiks õppematerjali kirjutamisel kasutada metafoore ja huumorit ning lühikesi ja lihtsaid lauseid. Juhendi autorid kirjeldavad lisaks, et tekst ja veebileht võiksid olla mugavalt loetavad ja navigeeritavad ning ei tohiks olla eksitavad. Näiteks võiks nähtaval hoida sisukorda ning vältida üleliigseid linke teksti sees, et lugeja peatükkide läbimisel ära ei eksiks või liialt kõrvale ei kalduks.

Andri Soone on oma materjalide [4] koostamisel enamjaolt lähtunud Wikibooksi viiest õpiku koostamise põhimõttest [7], mis rõhutavad

- materjalide ühtlast struktuuri (raamistike reegel),
- olemasolevate üheselt mõistetavate terminite kasutust (täendusrikaste nimede reegel),
- 4-6 uue idee tutvustamist peatüki kohta (hallatavate arvude reegel),
- võimalust tugineda varasemalt õpitule (hierarhia reegel),
- tähtsamate ideede tihedat kordamist (kordamise reegel).

Hong Kongi haridusameti kvaliteetsete õpikute kirjutamise juhendis [8] peetakse lisaks eelnimetatule oluliseks ka peatükkide loogilist järjekorda ning seda, et materjalide läbimine ja ülesanded oleksid kursuse sihtgrupile jõukohased.

Lugejale võib olla kasulik teada, et ka Hariduse Infotehnoloogia Sihtasutuse kodulehelt võib leida juhendeid kvaliteetse õppevara loomiseks [9].

## **2.2 Eeldatav sihtgrupp**

Silmaringimaterjalide sihtrühmaks võib pidada üliõpilasi, kes osalevad programmeerimiskursusel või oskavad kursuse läbinutega võrreldaval tasemel programmeerida ning kellel on aega ja huvi uurida lisamaterjale. Materjalid püüti koostada üsna konkreetsed ja praktilised, sest programmeerimiskursus on suunatud peamiselt informaatika, arvutitehnika ja matemaatika erialade tudengitele. Konkreetsemate näidetega praktilistest materjalidest võiks nendele õpilastele rohkem kasu olla, sest mingil hetkel puutuvad nad tõenäoliselt nende teemadega sellel tasemel kokku. Koodinäidetest võib kasu olla kasvõi kursuse projekti realiseerimisel või järgmistel oma eriala kursustel. Samas jälgiti materjale koostades, et vähemalt osad teemad oleksid huvitavad ka kursusel osalevatele mitteinformaatikutele.

## **2.3 Valitud põhimõtted**

Materjalide koostamisel otsustati lähtuda Wikibooks'i põhimõtetest [7] ning näiteks ühtlase struktuuri tagamiseks jälgida, et uued materjalid oleks vormi ja stiili poolest sarnased olemasolevate silmaringimaterjalidega. Jälgiti, et igas peatükis leiduks ka pilte või jooniseid, et järjestikust teksti poleks liiga palju ja materjal oleks vaheldusrikkam. Igasse uude peatükki lisati õpitu rakendamiseks vähemalt üks vastava teemaga seotud programmeerimisülesanne ja interaktiivsed enesekontrollitendid. Iga peatüki algusesse lisati lõik, mis annab sellest peatükist ülevaate ja annab mõista, mida saab peatükist õppida.

Teemad on valdavalt üksteisest sõltumatud, nii et peatükke ei pea läbima kindlas järjekorras. Sellegipoolest pandi peatükid kindlasse järjekorda, et iga teema oleks seotud programmeerimiskursuse vastava nädala teemaga ning oleks valdavalt läbitav selleks hetkeks olemasolevate teadmistega. Mõnel juhul, kus see kasulik võiks olla, on silmaringimaterjalid üksteisega seotud viidete abil.

### **3. Uued materjalid**

#### **3.1 Uute teemade valik**

Siinses töös valiti uute materjalide teemad isikliku huvi põhjal ja lähtudes sellest, milliseid TÜ informaatika õppekava kursustel ning arvutiteaduse ja matemaatika valikainetes käsitletavaid teemasid saaks Pythoni abil tutvustada [1]. Lisaks valiti teemasid nende hulgast, mida pakkus tuleviku jaoks välja olemasolevate silmaringimaterjalide autor oma töös [4]. Uuriti ka, mida käsitlevad silmaringiteemadena teised TÜ arvutiteaduse instituudi programmeerimise õpetamise töörühma koostatud programmeerimise kursused [10, 11, 12] ning millised lisamaterjalid on olemas Aivar Annamaa õpikus [3]. Kui valiti teema, mis oli mõnel kursusel kasutusel silmaringiteemana, püüti vastav materjal kirjutada programmeerimiskursuse jaoks praktilisem ning konkreetsem.

##### **3.1.1 Veebisisu parsimine**

Esimeseks uueks materjaliks valiti veebilehtede sisu parsimise teema, mis tutvustab HTML- ja XML-vormingut. Ühtlasi antakse ülevaade XPath- ja CSS-selektoritest, mis aitavad vormingute struktuurist paremini aru saada. Praktiliseks näiteks on toodud RSS-vood.

Peatükk sobiks kursusel läbimiseks vahetult enne või pärast olemasolevat rakendusliideste peatükki, kuna teemad on omavahel seotud – mõlemas peatükis tuleb teha päringuid veebiaadressitele. Teema valikul lähtuti sellest, et HTML-vormingu tundmaõppimine aitab veebisisu paremini mõista ja on vajalik ka veebilehtede loomisel. XML-vormingu tundmine aitab lugeda sellel kujul olevaid andmeid ja faile, näiteks RSS-vooge, SVG-vormingus vektorgraafikafaile või teatud andmebaasifaile. HTML-iga tutvutakse ka näiteks TÜ informaatika õppekava kursustel „LTAT.03.002 Sissejuhatus erialasse“ [13], „LTAT.05.004 Veebirakenduse loomine“ [14] ja kursusel „MTAT.03.297 Veebilehtede loomine“ [15].

##### **3.1.2 Võistlusprogrammeerimine**

Teine uus peatükk tutvustab võistlusprogrammeerimist ja võistlustel kasutatavaid algoritme ning annab soovitusi võistlustel osalemiseks. Võistlusülesannete lahendamine on paljude jaoks põnev viis oma programmeerimisoskust arendada. Peamiselt õpetab see kiiret probleemide lahendamise oskust, algoritmilist mõtlemist ning algoritmide ja andmestruktuuride kasutamist. Sel viisil on teema seotud kursustega „LTAT.03.005

Algoritmid ja andmestruktuurid“ [16] ja „MTAT.03.269 Programmeerimisvõistlused” [17]. Kuna see on ainus silmaringimaterjal, mis eeldab rekursiooni mõistmist, siis on see paigutatud teemade järjekorras vastava õpiku peatüki juurde.

### 3.1.3 Pilditöötlus

Kolmas peatükk on seotud piltide töötlemise ja kuvamisega Pythoni teegi PIL abil. Tutvustakse ka värvisüsteeme ja pildivorminguid. Peatükis kasutatakse läbiva näitena Pythoni logo, millele rakendatakse mitmesuguseid teisendusi. Näide sellest on kujutatud joonisel 1.

#### Lihtsamad teisendused

Järgnev kood teeb Pythoni logo erinevatest teisendustest järjendi ja kuvab iga teisenduse:

```
1 from PIL import ImageOps
2
3 teisendused = [
4     img.rotate(45, expand=True), # 45-kraadine pööre
5     img.transpose(Image.Transpose.FLIP_TOP_BOTTOM), # Vertikaalne peegeldus
6     img.resize( (img.width//2, img.height//2) ), # Kahekordne vähendus
7     img.crop( (50, 50, img.width-50, img.height-50) ), # Väljalõige keskelt
8     ImageOps.grayscale(img), # Mustvalges värviskaalas
9     ImageOps.invert(img), # Vastandvärvides
10    ImageOps.expand(img, 5, "#00ff00") # Rohelise raamiga
11 ]
12
13 for t in teisendused:
14     t.show()
```



Joonis 1. Pythoni abil Pythoni logo teisendamine.

Kuigi keerulisemate arvutigraafika teemadeni ei jõuta, võiks teema olla hea sissejuhatus arvutigraafika valdkonda ja pildiandmetega töötamiseks. Näiteks aitab peatükk lugejail erinevaid pildifailivorminguid, värvisüsteeme ja pilditöötlusoperatsioone eristada. Materjal võiks seega olla kasulik ka tudengitele nendelt erialadelt, mis pole informaatikaga seotud. Lähemalt uuritakse pilditöötlust näiteks instituudi kursustel „MTAT.03.015 Arvutigraafika” [18] ja „MTAT.03.132 Multimeedia“ [19]. Värvsüsteemide ja -harmooniate tundmine on tõenäoliselt kasulik ka õpilastele, kes on huvitatud kasutajaliidest ja veebi disainimisest.


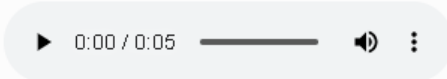

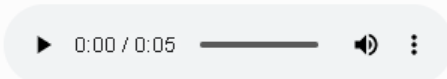




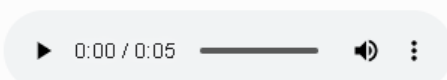
### 3.1.4 Helitöötlus ja -süntees

Järgmine peatükk on seotud heli loomise ja töötlemisega, käsitletakse ka heliandmete visualiseerimist. Selle peatüki jaoks on loodud ja kasutatud ka helinäiteid, mida on võimalik kuulata. Joonisel 2 on väljalõige peatükist, kus on näha tabel lainevormidega. Joonisel 3 on kujutatud mõned peatükis leiduvad koodiread ja vastavad helisignaalid, mida nende abil on võimalik tekitada.

Peatükk aitab aru saada digitaalse heli salvestamise ja töötlemise alustest ning selgitab õppijale lähemalt, kuidas heli matemaatilisel kujul väljendatakse. Seda teemat uuritakse laiemalt arvutiteaduse instituudi aines „MTAT.03.132 Multimeedia” [19]. See silmaringimaterjal on enamikest teistest keerulisem ja selle näited on paremini mõistetavad, kui on läbitud objektorienteeritud programmeerimise peatükid. Seetõttu paigutati materjal kursuse 15. nädala peatüki „Objektorienteeritud programmeerimine 2“ juurde.

#### Lainevormid

Lisaks sagedusele ja amplituudile mõjutab heli kõla ka selle lainevorm. Peale siinuslaine eristatakse ka teisi lainevorme, kuigi tegelikult on needki väljendatavad erinevate siinuslainete summana. Teisisõnu, kõiki lainevorme peale siinuslaine võib põhimõtteliselt nimetada akordideks. Tabelis on toodud tuntuimad lainevormid:

Lainevorm	Valem	Kuju	Kõla (220Hz) Allikas: <a href="#">Wikimedia Commons</a>
Siinuslaine	$f(x) = \sin(2\pi * x)$		
Kandiline ehk impulsslaine	$f(x) = 2 * (2 * \lfloor x \rfloor - \lfloor 2 * x \rfloor) + 1$		
Kolmnurklaine	$f(x) = 2 *  (2x - 0.5) \bmod 2 - 1  - 1$		
Saehammaslaine	$f(x) = -(2x \bmod 2) + 1$	 või 	

Joonis 2. Lainevormide näited.



Joonis 3. Digitaalselt loodavate helisignaali näited.

### 3.1.5 Kodeerimine ja krüptograafia

Viienda peatüki sisu hõlmab andmete kodeerimist, krüpteerimist ja rääkimist ja tutvustab vastavaid Pythoni mooduleid ning teke. Muuhulgas tutvustatakse lühidalt ploki ahelat. Ühe näitena kasutatakse krüpteeritud sõnumite vahetamise protsessi. Näidet seletatakse peatükis kirjelduse, joonise ja vastava programmikoodi abil.

Nii on peatüki teema seotud arvutiteaduse instituudi valikkursusega „LTAT.05.021 Sissejuhatus Blockchaini tehnoloogiasse” [20] ja erinevate krüptograafia kursustega. Peatükk võib aidata täpsemini mõista, kuidas toimub arvutis info salvestamise, lugemise ja (de)kodeerimise protsess. Kuna tutvustatakse peamiselt teksti kujul andmete ja failide teisendamist, siis sobib peatükk sõne- ja failitöötamise põhiteemaga samasse nädalasse. Pärast vastava põhiteema läbimist võib see silmaringipeatükk aidata paremini vahet teha erinevatel lihtandmetüüpidel nagu näiteks arvud ja sõned.

### 3.1.6 Pythoni siseehitus

Selles peatükis selgitatakse lähemalt Pythoni kompilaatorite ja interpretaatorite toimimist. Sealjuures tutvustatakse ka Pythoni baitkoodi ning Pythoni seost programmeerimiskeelega C. Joonisel 4 on näide sellest, kuidas tutvustatakse Pythoni funktsioonide abil Pythoni baitkoodi.

## Pythoni baitkood

Järgnevates näidetes vaatamegi peamiselt, mismoodi tegutseb Pythoni koodiga selle ametlik implementatsioon CPython. Kasutame sisseehitatud moodulit `dis`, mille abil saab mingi funktsiooni baitkoodi osadeks võtta ja loetavalt väljastada:

```
1 | def liida(a, b):
2 |     print("Liidan!")
3 |     return a+b

>>> import dis
>>> dis.dis(liida)
2          0 LOAD_GLOBAL           0 (print)
          2 LOAD_CONST             1 ('Liidan!')
          4 CALL_FUNCTION          1
          6 POP_TOP

3          8 LOAD_FAST              0 (a)
         10 LOAD_FAST              1 (b)
         12 BINARY_ADD
         14 RETURN_VALUE
```

Igale koodireale vastab üldjuhul mitu käsku:

1. Tabeli vasakus veerus on kirjas reanumbrid, millele järgmised baitkoodi käsud vastavad.
2. Teises veerus olevad arvud on positsioonid, kus vastav käsk baitkoodis asub.
3. Keskel on baitkoodikäsu nimi.
4. Neljandas veerus võib olla käsu argument (täpsemini indeks, mis viitab selle asukohale mälus).
5. Paremal võib olla argumenti väärtus või muutuja nimi.

Joonis 4. Pythoni koodi kuvamine baitkoodi kujul.

Teema valiti isikliku huvi tõttu ja eeldati, et see aitab programmeerimiskeelte ja programmide olemust ja ehitust paremini mõista. Pythoni programmide pakkimist tutvustab ka lisapeatükk Aivar Annamaa õpikus [3]. Kompileerimine või pakkimine võimaldab programme käivitada ka arvutites, kuhu pole paigaldatud Pythoni interpretaatorit. Teema sobib sissejuhatuseks kursusele „LTAT.03.006 Automaadid, keeled ja translaatorid” [21]. Peatükk on pandud viimaseks silmaringimaterjaliks, sest seda on kõige parem läbida siis, kui lugeja on juba tuttav kõikide Pythoni keele põhikonstruktsioonidega.

### 3.1.7 NumPy massiivid

See peatükk annab ülevaate Pythoni teegist NumPy. Seda teeki ja NumPy massiivi andmestruktuuri on oluline mõista, et andmeteaduse või masinõppe ülesannetes andmeid töödelda. Esialgsete silmaringimaterjalide autor jättis andmeteadusega seotud teema enda teemade valikust välja [4]. Kuivõrd see on laiem teema ja ainuüksi NumPy kasutamine võib juba üsna keeruliseks osutada, siis keskendubki see peatükk peamiselt NumPy-le. Samas loetleb ja julgustab see kasutama sellega seotud teeki, sest kui NumPy-ga tuttav olla, on ka nende tööriistade kasutamine lihtsam. Leiti, et teeki NumPy pole järgnevatel informaatika õppekava kursustel nagu „LTAT.02.002 Sissejuhatus andmeteadusesse” [22] ja „LTAT.01.003 Tehisintellekt” [23] eraldi piisavalt seletatud.

Selles peatükis käsitletakse mitmemõõtmelisi järjendeid ja massiive, mistõttu peab kindlasti olema läbitud järjendite peatükk. Kahemõõtmeliste andmestruktuuride itereerimiseks on tihti vaja kasutada kahekordset tsüklit ja kursusel uuritakse kahekordset tsüklit käsitleval nädalal uuesti ka mitmemõõtmelisi järjendeid. Seetõttu sobib NumPy massiivide teema ilmselt kõige paremini vastava õpiku peatüki silmaringimaterjaliks.

### 3.2 Uus peatükkide järjekord

Kuna õpikut uuendati ja teemad järjestati ümber, tuli ka silmaringimaterjalide järjekorda muuta. Tabelid 1 ja 2 kirjeldavad vastavalt peatükkide endist ja uut järjestust.

Tabel 1. Programmeerimiskursuse senine peatükkide ja silmaringimaterjalide järjekord [2]

Nädal	Teema	Silmaringimaterjal
1.	Sissejuhatus	Standardteek ja moodulid
2.	Avaldised ja lihtlaused	-
3.	Tingimus- ja korduslaused	Rakendusliidesed
4.	Funktsioonid	-
5.	Algoritmid	Regulaaravaldised
6.	Kontrolltöö	-
7.	Järjendid	Andmebaasid
8.	Järjendid 2	-
9.	Järjendite muutmine	Veebirakenduste loomine
10.	Andmestruktuurid	-
11.	Andmestruktuurid 2	Objektorienteeritud programmeerimine
12.	Kontrolltöö	-
13.	Rekursioon	Graafiliste mängude loomine
14.	Rekursioon 2	-
15.	Mitmesuguseid algoritme	Keerulisemad Pythoni võimalused
16.	Kordamine. Projektide esitlused	-

Tabel 2. Programmeerimiskursuse nädalad, uued teemad ja vastavad silmaringimaterjalid

Nädal	Teema	Silmaringimaterjal
1.	Sissejuhatus	Standardteek ja moodulid
2.	Tingimuslause	Regulaaravaldised
3.	Funktsioon	Andmebaasid
4.	Korduslause	Veebirakenduste loomine
5.	Sõned. Lihtsam failitöötlus	Kodeerimine ja krüpteerimine
6.	Kontrolltöö	-
7.	Järjendid	Graafiliste mängude loomine
8.	Järjendid 2	Keerulisemad Pythoni võimalused
9.	Kahekordne tsükel. Failitöötlus	NumPy massiivid
10.	Andmestruktuurid	Rakendusliidesed
11.	Andmestruktuurid 2	Veebisisu parsimine
12.	Kontrolltöö	-
13.	Rekursioon	Võistlusprogrammeerimine
14.	Objektorienteeritud programmeerimine	Pilditöötlus
15.	Objektorienteeritud programmeerimine 2	Helitöötlus ja -süntees
16.	Kordamine. Projektide esitlused	Pythoni siseehitus

Silmaringimaterjalid paigutati nädalate juurde nii, et igast materjalist aru saamiseks oleks tarvis võimalikult vähe järgmiste nädalate õpiku peatükkidega tutvuda. Iga materjali algusesse on lisatud ka lõik, mis annab lugejale teada, millised eelnevad õpiku peatükid peavad olema läbitud ning millistega järgmistest on soovitatav vähemalt põgusalt tutvuda.

Leiti, et õpiku peatükkide uues järjestuses võib mitmed peatükid nihutada varasemaks. Peatükk „Rakendusliidesed“ tõsteti samas hilisemaks, 10. nädalale, sest sellel nädalal käsitletakse sõnastiku andmestruktuuri, mis aitab peatükist paremini aru saada. Põhimõtteliselt võib 10. ja 11. nädalate materjalid vahetada ka 14. ja 15. materjalidega, aga kokkuvõtvalt leiti, et heli- ja pilditöötlus on keerulisemad teemad ning sobivad paremini kursuse lõppu. Tabelis 2 kirjeldatud nädalate ja peatükkide järjekord kinnitati kursuse uuendamise koosolekul.

Materjalide seis enne tööd on nähtav kursuse eelmise aasta lehel:

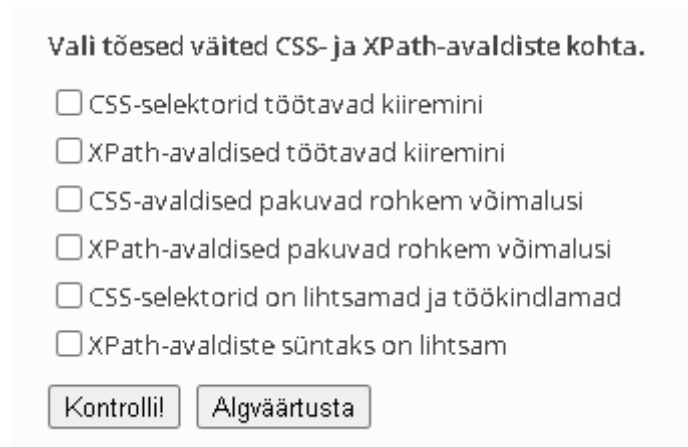
<https://courses.cs.ut.ee/2021/programmeerimine/fall/Main/Silmaringimaterjalid>

Selle töö tarvis täiendatud silmaringimaterjalid asuvad 2022. aasta programmeerimiskursuse veebilehel (Lisa I).

### 3.3 Ülesanded ja enesekontrollitesti

Igasse silmaringipeatükki lisati 4-7 interaktiivset enesekontrolliküsimust ja 2-4 programmeerimisülesannet. Erand on võistlusprogrammeerimise peatükk, kus leiti, et piisab ühest ülesandekirjeldusest, mis suunaks lahendama peatükis mainitud võistluskeskkonnades leiduvaid ülesandeid.

Kokkuvõtet peatükkidesse üldjuhul ei lisatud ning tähtsamad ideed, mida lõpus uuesti korrata taheti, vormistati mõnel juhul enesekontrolliküsimustena. Näiteks veebisisu parsimise peatükis eristatakse ja kasutatakse nii CSS- kui ka XPath-avaldisi. Ühel korral võrreldakse neid teksti sees ja peatüki lõpus on küsimus, kus tuleb neid ise võrrelda (joonis 5).



Vali tõesed väited CSS- ja XPath-avaldiste kohta.

- CSS-selektorid töötavad kiiremini
- XPath-avaldised töötavad kiiremini
- CSS-avaldised pakuvad rohkem võimalusi
- XPath-avaldised pakuvad rohkem võimalusi
- CSS-selektorid on lihtsamad ja töökindlamad
- XPath-avaldiste süntaks on lihtsam

Kontrolli! Algväärtusta

Joonis 5. Enesekontrolliküsimus veebisisu parsimise peatükis

Programmeerimisülesanded pakuvad praktilisi või huvitavaid ideid, kuidas õpitud rakendada. Ühes peatükis võib olla erineva raskusastmega ülesandeid, aga üldiselt on sealjuures jälgitud, et kõik ülesanded oleksid vastaval kursuse nädalal õppijatele jõukohased.

## 4. Silmaringimaterjalide tulevik

### 4.1 Materjalide kasutamine ja sihtgrupp

Esimesed silmaringimaterjalid loodi programmeerimiskursuse jaoks 2020. aastal [4]. Siiani pole korrapäraselt uuritud, kui paljud õpilased neid lugenud või kasutanud on. Üksikud kursuse läbinud õpilased, keda juhuslikult ja pisteliselt küsitleti, ei olnud isegi teadlikud silmaringimaterjalide olemasolust või polnud neid uurinud<sup>2</sup>. Üldiselt võib sellel olla mitu põhjust:

- Kursuse jooksul ei osutatud silmaringimaterjalidele tähelepanu.
- Materjalid paiknesid kursuse veebipõhisest õpikust eraldi, mistõttu õpilased neid ei avastanud.
- Õpilastel polnud aega või huvi lisamaterjale uurida.

Uuendatud kursusel on põhi- ja silmaringimaterjalid koondatud ühte kohta, nii et need võiksid olla kergemini leitavad. Kursuse jooksul hakatakse lisamaterjale õpilastele rohkem reklaamima. Kursuse õppejõud on mõelnud ka sellele, et võiks pakkuda näiteks võimalust lahendada mõne kodutöö või praktikumiülesande asemel vastava nädala silmaringipeatüki ülesanne. See võib olla hea võimalus muuta kursus huvitavamaks nendele üliõpilastele, kellele on paljud kursuse teemad niigi selged, kuid kes mingil põhjusel eeleksamit ei sooritanud.

Samuti oleks huvitav teha kursusel iganädalasi küsitlusi ka selle kohta, kui palju silmaringimaterjale uuritakse. Selle abil oleks võimalik täpsemini määrata materjalide sihtgrupp. Kui peaks selguma, et sellel kursusel vaatavad materjale tõesti väga vähesed, saaks ehk materjalidest koostada uue valikkursuse. See kursus saaks olla sama struktuuriga kui programmeerimiskursus ja seda saaksid läbida näiteks need, kes läbisid eeleksami teatud tulemusega. Kursus saaks materjale edasijõudnutele tutvustada Pythoni abil, mis õpetaks ühtlasi põhjalikumalt selle keele vahendeid.

### 4.2 Väljajäänud teemad

Vahest oleks lisamaterjalide lugejaid rohkem, kui iga nädal oleks valida enamate teemade vahel. Mõned neist võivad olla ka lühemad, nädala teemaga tihedamalt seotud

---

<sup>2</sup> Info selgus autori tutvuste kaudu

lisalugemised. Selles bakalaaurusetöös kaaluti veel järgnevaid teemasid, mida saaks Pythoni abil tutvustada:

- Numbrilised meetodid
- Häkkimine
- TeX
- Koostöö tarkvaraarenduses
- Masinõpe ja tehisintellekt
- Kvantarvutid ja -algoritmid

Valimata jäid need teemad peamiselt isiklike eelistuste ja selle tõttu, et mõned teemad tundusid autorile liiga võõrad ning nende kohta materjali kogumine oleks olnud liiga ajakulukas. Mõne teema kohta tekkisid siiski konkreetsemad ideed.

Näiteks tarkvaraarenduse peatükis oli plaanis tutvustada versioonihaldust, erinevaid Pythoni arenduskeskkondi nagu näiteks PyCharm, samuti lintimist ning programmeerimisparadigmasid. See aitaks õpilastel tutvuda tänapäevase tarkvaraarendusprotsessiga ning mõnda versioonihaldussüsteemi oleks hiljem võimalik kasutada isikliku või kursuse jaoks koostööna loodava projekti arendamisel. Avatud lähtekoodiga programmi arendusse saab panustada igäüks – seda saaks kasutada ühe tarkvaraarenduse koostöövormi näitena. Kui teema paigutada kursuse lõpu poole, kus õpilastel on piisavalt oskusi, saaks ehk mingi ülesande siduda avatud lähtekoodi panustamisega.

Tehisintellektiga seotud peatükis saaks selgitada masinõppe ja tehisintellektiga seotud mõisteid ning näidata, kuidas kasutada näiteks teeki Scikit-learn. Samuti saaks praktiliste näidete abil selgitada tehisintellekti ohutuse ja eetilise teemat. Häkkimise teema puhul oleks ka saanud Pythoni abil erinevaid praktilisi näiteid tuua ja sealjuures turvariskidele tähelepanu pöörata.

A. Soone nimetas oma töös ka mõned materjalide teemad, mis valimata jäid ja mille kohta uusi silmaringimaterjale ei loodud. Nende seas on muuhulgas teemad, mis hõlmaksid koodistiili, dokumenteerimise tavasid ja andmete visualiseerimist. [4]

### 4.3 Teiste kursuste silmaringimaterjalid

Järgmiste teemade valimisel saaks lähtuda ka teiste sarnase ülesehitusega programmeerimiskursuste [10, 11, 12] silmaringipeatükkidest. Materjale leidub näiteks järgmiste teemade kohta:

- Programmeerimise ajalugu
- Arvusüsteemid
- Tõeväärtused
- Labürint
- Navigeerimine
- Kodumasinad
- Isejuhtivad autod
- Keeletehnoloogia ja keeletöötlus
- Ametid ja õppimisvõimalused
- Erinevad programmeerimiskeeled
- Juhuslikkus
- Masinõpe

Kuigi kõik nimetatud teemad ei pruugi tingimata täiendada osalejate programmeerimisoskust, võib selle kursuse jaoks otsida viise, kuidas vastavad teemad mingite konkreetsemate koodinäidetega siduda. Samas tasub vahest mõelda ka muudele lisamaterjalide liikidele lisaks selles töös defineeritud lisalugemistele ja silmaringimaterjalidele.

Tasub mainida, et ka siiani programmeerimiskursusel kasutatud Aivar Annamaa õpikus leiduvad lisamaterjalid, mida saaks kursuse jaoks silmaringimaterjalideks vormistada [3]. Teine võimalus on programmeerimiskursusele uute lisamaterjalide loomise asemel lisada kursuse õpikusse viiteid mujal leiduvatele õppematerjalidele, mis õpilastele huvi võiksid pakkuda.

## 5. Kokkuvõte

Töö eesmärk oli täiendada Tartu Ülikooli informaatika õppekava kursuse „LTAT.03.001 Programmeerimine“ veebipõhiseid lisamaterjale ja pakkuda ideid, kuidas võiks neid tõhusamalt kasutada. Selle töö käigus loodi seitse uut lisamaterjali. Materjalide eesmärk on võimaldada uudishimulikel kursuse osalejatel täiendada oma informaatikaalast silmaringi ja programmeerimisoskust. Uued materjalid hõlmavad veebisisu parsimist, võistlusprogrammeerimist, pildi- ja helitöötlust, info kodeerimist, krüpteerimist, kompileerimist ja suurte andmehulkade töötlemist. Töö jooksul koondati ja pakuti välja omapoolsed mõtted selle kohta, kuidas võiks loodud materjale edaspidi paremini kasutada ning milliseid lisamaterjale oleks veel võimalik luua. Eelkõige tuleks suunata kursusel osalejate tähelepanu lisamaterjalidele ja uurida küsitluste abil põhjalikumalt materjalide kasutamist üliõpilaste seas.

## 6. Viidatud kirjandus

- [1] Programmeerimine LTAT.03.001 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.03.001/details> (08.05.2022)
- [2] Programmeerimine LTAT.03.001 - Tartu Ülikooli arvutiteaduse instituut.  
<https://courses.cs.ut.ee/2021/programmeerimine/> (10.05.2022)
- [3] Annamaa, Aivar. Programmeerimise õpik.  
<https://progeopik.cs.ut.ee/> (08.05.2022)
- [4] Soone, Andri 2020. Programmeerimise kursusele lisamaterjalide koostamine. TÜ arvutiteaduse instituudi bakalaureusetöö.  
[https://comserv.cs.ut.ee/ati\\_thesis/datasheet.php?id=69946&year=2020](https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=69946&year=2020) (10.12.2021)
- [5] Paraskakis, Iraklis et al. 2008. „Facilitating Access to Distance Education: Creating Pedagogically Driven Courseware Using Simple Technologies“. International Conference on Advanced Learning Technologies conference paper.  
<https://ieeexplore-ieee-org.ezproxy.utlib.ut.ee/stamp/stamp.jsp?tp=&arnumber=4561730&tag=1>  
(10.12.2021)
- [6] Gibbs, William J. et al. 1997. Virtual Courses and Visual Media. Eastern Illinois University research paper.  
<https://files.eric.ed.gov/fulltext/ED408965.pdf> (15.11.2021)
- [7] Wikibooks: Textbook considerations.  
[https://en.wikibooks.org/wiki/Wikibooks:Textbook\\_considerations](https://en.wikibooks.org/wiki/Wikibooks:Textbook_considerations) (10.12.2020)
- [8] Guiding Principles for Quality Textbooks (Revised June 2016).  
<https://www.edb.gov.hk/en/curriculum-development/resource-support/textbook-info/GuidingPrinciples/index.html> (10.12.2021)
- [9] Õppevara kvaliteedi juhendid - Hariduse Infotehnoloogia Sihtasutus.  
<https://www.hitsa.ee/ikt-hariduses/digitaalne-oppevara/oppevara-kvaliteet/oppevara-kvaliteedi-juhendid> (10.05.2022)
- [10] Programmeerimisest maalähedaselt, 2021 - Tartu Ülikooli arvutiteaduse instituut.  
<https://courses.cs.ut.ee/2021/progmaa/> (09.05.2022)

- [11] MOOC Programmeerimise alused, 2018 - Tartu Ülikooli arvutiteaduse instituut.  
<https://courses.cs.ut.ee/2018/eprogalused/> (09.05.2022)
- [12] MOOC Programmeerimise alused II, 2019. - Tartu Ülikooli arvutiteaduse instituut.  
<https://courses.cs.ut.ee/2019/eprogalused2/> (09.05.2022)
- [13] Sissejuhatus erialasse LTAT.03.002 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.03.002/details> (09.05.2022)
- [14] Veebirakenduste loomine LTAT.05.004 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.05.004/details> (2022.05.09)
- [15] Veebilehtede loomine MTAT.03.297 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/MTAT.03.297/details> (09.05.2022)
- [16] Algoritmid ja andmestruktuurid LTAT.03.005 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.03.005/details> (09.05.2022)
- [17] Programmeerimisvõistlused MTAT.03.269 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/MTAT.03.269/details> (09.05.2022)
- [18] Arvutigraafika MTAT.03.015 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/MTAT.03.015/details> (09.05.2022)
- [19] Multimeedia MTAT.03.132 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/MTAT.03.132/details> (09.05.2022)
- [20] Sissejuhatus Blockchaini tehnoloogiasse LTAT.05.021 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.05.021/details> (09.05.2022)
- [21] Automaadid, keeled ja translaatorid LTAT.03.006 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.03.006/details> (09.05.2022)
- [22] Sissejuhatus andmete adusesse LTAT.02.002 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.02.002/details> (09.05.2022)
- [23] Tehisintellekt LTAT.01.003 - Tartu Ülikooli õppeinfosüsteem.  
<https://ois2.ut.ee/#/courses/LTAT.01.003/details> (09.05.2022)

## **Lisad**

### **I. Koostatud materjalide asukoht**

Koostatud silmaringimaterjalid on koos ülejäänud kursuse materjalidega kasutusel ja leitavad programmeerimiskursuse arvutiteaduse instituudi veebilehel, kus neid võidakse mingil määral parandada ja muuta. Samas keskkonnas on eraldi lehel olemas ka materjalide koopiad nende valmimise hetkest:

<https://courses.cs.ut.ee/2022/programmeerimine/fall/Main/Silmaringimaterjalid>

## II. Litsents

### **Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks**

Mina, **Kevin Kekki**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose  
**Programmeerimiskursuse silmaringimaterjalide täiendamine**  
mille juhendaja on Reimo Palm,  
reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi  
DSpace kuni autoriõiguse kehtivuse lõppemiseni.
2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks  
Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative  
Commonsi litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost  
reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja  
kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega  
isikuandmete kaitse õigusaktidest tulenevaid õigusi.

*Kevin Kekki*

**10.05.2022**