

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Nikita Kislõi
Programmeerimise nädalatestide
vastuste analüüsija
Bakalaureusetöö (9 EAP)

Juhendaja: Reimo Palm

Tartu 2025

Programmeerimise nädalatestide vastuste analüüsija

Lühikokkuvõte: Tartu Ülikooli bakalaureuseõppe aines Programmeerimine (LTAT.03.001) kasutatakse õppimise toetamiseks iganädalasi Moodle'i teste. Testide efektiivsus sõltub küsimuste kvaliteedist, millele praegused analüüsivahendid ei võimalda alati anda piisavalt täpset hinnangut. Käesoleva töö raames loodi Pythonis tööriist, mis analüüsib testiküsimuste statistilisi näitajaid klassikalise testiteooria ja üksikvastuste teooria põhjal. Programm võrdleb ka iga õppija esimest ja parimat katset, mis võimaldab hinnata küsimuse õppimist toetavat mõju. Testimise käigus selgus, et suur osa küsimustest olid liiga lihtsad või vähese eristusvõimega, mistõttu ei täitnud need oma eesmärki piisavalt hästi. Loodud rakendus aitab õpetajatel tuvastada nõrgad küsimused ning parandada testide kvaliteeti.

Võtmesõnad: programmeerimise algkursus, Moodle'i testid, klassikaline testiteooria, üksikvastuste teooria

CERCS: P160 Statistika, operatsioonanalüüs, programmeerimine, finants- ja kindlustusmatemaatika, S270 Pedagoogika ja didaktika

Analyser for answers to weekly quizzes in the programming course

Abstract: In the bachelor's course Programming (LTAT.03.001) at the University of Tartu, weekly Moodle-based tests are used to support learning. The success of such tests depends on the quality of individual questions, but current analysis tools are often insufficient to assess it. This thesis presents a Python-based tool that analyzes test question quality using statistical indicators from classical test theory and item response theory. The tool also compares each student's first and best attempt to assess whether a question promotes learning. Testing showed that a large portion of questions were too easy or had low discrimination, failing to fulfill their purpose effectively. The developed application supports teachers in identifying weak questions and improving overall test quality.

Keywords: introductory programming course, Moodle quizzes, classical test theory, item response theory

CERCS: P160 Statistics, operation research, programming, actuarial mathematics, S270 Pedagogy and didactics

Sisukord

Sissejuhatus.....	6
1. Vajaduse kirjeldus.....	7
2. Analüüsiks kasutatavad statistilised meetodid.....	8
Vastuste arv.....	8
Tulemuste aritmeetiline keskmine.....	8
Tulemuste standardhälve.....	9
Klassikaline testiteooria.....	9
Üksikvastuste teooria.....	9
Punkt-biseriaalne korrelatsioonikordaja.....	10
Raskus.....	11
Eristusvõime.....	11
Oletamisparameeter.....	11
Üksikvastustefunktsioon.....	12
Informatsioonifunktsioon.....	12
3. Analüüsiks kasutatava programmi eeskujud.....	13
Moodle'i testide statistika.....	13
Cronbachi alfa kalkulaatorid.....	14
Iteman.....	14
Xcalibre.....	15
4. Nõuete kirjeldus.....	16
5. Süsteemi ülesehitus.....	19
Google Colab.....	19
Os.....	19
Pandas.....	19
Sys.....	20
NumPy.....	20
SciPy.....	20
Google Colab Drive.....	20
PyIrt.....	20
Matplotlib.....	21
6. Realiseerimise protsess.....	22
Arendusloogika ja funktsionaalsused.....	22
Tehnilised raskused ja nende lahendamine.....	24

7. Testimine.....	25
8. Arutelu.....	27
Tulemus.....	27
Rakenduse kasutamine.....	29
Piirangud ja edasiarendamise võimalused.....	29
Kokkuvõte.....	30
Viidatud kirjandus.....	31
Lisad.....	33
I. Programmi kättesaamine ja juhend.....	33
II. Litsents.....	35

Sissejuhatus

Tartu Ülikooli informaatika bakalaureuseõppe kursusel Programmeerimine (LTAT.03.001) on kasutusel iganädalased testid, mille abil saavad üliõpilased kontrollida, kui hästi on nad kursuse materjalist aru saanud. Kuna testid on osa õppimise protsessist, on oluline, et nende küsimused oleksid kvaliteetsed.

Õigesti koostatud testiküsimus mitte ainult ei mõõda teadmisi, vaid toetab ka õppimise protsessi. Samas ei ole selliste küsimuste loomine lihtne ning ka head küsimused võivad aja jooksul vananeda või kaotada oma funktsiooni, kui õppematerjal või sihtrühm muutub. Regulaarsel küsimuste analüüsil on seetõttu oluline roll testide kvaliteedi säilitamisel.

Selle töö eesmärk on luua tööriist, mis võimaldab hinnata testiküsimuste kvaliteeti põhjalikumalt. Töö jaguneb mitmeks osaks. Esmalt tutvustatakse probleemi ja põhjendatakse töö vajalikkust. Seejärel kirjeldatakse kasutatavaid meetodeid ja olemasolevaid tööriistu, mis sarnaseid ülesandeid täidavad. Edasi käsitletakse loodud programmi ülesehitust ja realiseerimist, tuues välja ka esinenud tehnilised raskused. Lõpuks esitatakse testimise tulemused ja arutelu selle üle, kui hästi tööriist probleemi lahendab ning millised on võimalikud edasiarendused.

Töö koostamisel kasutati ChatGPT mudeli 4o abi mitmel eesmärgil. Kirjalikus osas aitas see leida sobivaid teaduslikke ja usaldusväärseid allikaid, mis toetaksid töö teoreetilist osa. Praktilises osas kasutati ChatGPT-d eelkõige selleks, et leida infot kasutatud teekide dokumentatsiooni kohta ning otsida selgitusi vigade kohta, mis ilmnesid rakenduse koostamise käigus.

1. Vajaduse kirjeldus

Üks olulisematest ainetest, mida esmakursuslased Tartu Ülikooli bakalaureuseõppe informaatika õppekaval läbivad, on aine nimega Programmeerimine (LTAT.03.001). Sellest võtab igal aastal osa umbes 300 õpilast. Õpitava materjali kinnistamiseks on aines kasutusel nädalatestid, kus igas testis on hulk erinevaid küsimusi. Iga test põhineb konkreetse nädala materjalil. Nädalatestide peamine eesmärk on anda õpilasele võimalus ise kontrollida, kuidas ta sai aru vastava nädala teemast. Lisaks sellele aitavad testid õpilastel harjutada ja analüüsida vigu, mis esinevad testi tegemise ajal. Aine kontrolltööde üks osa on test, mis põhineb selleks hetkeks läbitud nädalate testide materjalil. Samuti üks osa aine eksamist on test, kus küsimused põhinevad kogu kursuse materjalil. Seega on nädalatestid osa kontrolltöödeks ja eksamiks ettevalmistusest.

Probleem seisneb selles, et need nädalatestid ei ole optimeeritud. Testides esinevad küsimused, mille statistika näitab, et nendest on õppimisel vähe kasu. Seega tuleb asendada need küsimused sellistega, mis hindavad õpilase teadmisi paremini.

Selle töö põhieesmärk on luua programm, mis analüüsib õpilaste vastuseid testide kaupa, kasutades erinevaid statistilisi meetodeid. Tehtud arvutuste põhjal loob programm PDF-faili, kus iga küsimuse jaoks on välja toodud küsimuse kvaliteeti iseloomustavad näitajad ja graafikud. Kui näitaja väärtuse põhjal ilmneb, et küsimus on probleemne, siis saab õppejõud selle põhjal testide küsimusi parandada. Kasutades programmi, saab nädalateste optimeerida, mille tulemusena on hinnang õpilaste teadmistele täpsem ning õpilaste harjutamine läbi testi on efektiivsem.

2. Analüüsiks kasutatavad statistilised meetodid

Õpikeskkonnas Moodle on olemas statistika eraldi iga testi ja iga konkreetse testiküsimuse kohta. See statistika sisaldab muu hulgas ka suurusi, mida võiks kasutada analüüsiks antud töö raames. Iga konkreetse testiküsimuse kohta on olemas järgmised huvipakkuvad andmed ja meetodid:

- vastuste arv;
- tulemuste aritmeetiline keskmine;
- tulemuste standardhälve.

Lisaks sellele on plaanis kasutada uusi meetodeid klassikalisest testiteooriast (*classical test theory*) ning üksikvastuste teooriast (*item response theory*) ja täpsemalt näitajaid:

- punkt-biseriaalne korrelatsioon (*point-biserial correlation* või ka *item-total correlation*);
- raskus (*difficulty*);
- eristusvõime (*discrimination*);
- oletamisparameeter (*pseudo-guessing parameter*).

Üksikvastuste teooriaga seotud graafikute joonistamiseks kasutatakse üksikvastustefunktsiooni (*item response function*) ja informatsioonifunktsiooni (*item information function*).

Vastuste arv

Vastuste arv näitab, kui palju vastuseid on antud konkreetsele testiküsimusele. Vastuste arvu kasutatakse teistes statistilistes meetodites. See näitaja on oluline andmeanalüüsis, kuna see mõjutab tulemuste usaldusväärust. Näiteks kui andmete hulk andmeanalüüsi jaoks on liiga väike, siis pole analüüsi tulemused piisavalt usaldusväärsed.

Tulemuste aritmeetiline keskmine

Aritmeetiline keskmine on arvvaartuste x_1, x_2, \dots, x_n summa ja nende väärtuste arvu jagatis. Konkreetse testiküsimuse arvvaartuste x_1, x_2, \dots, x_n aritmeetiline keskmine on selle testiküsimuse kõikide vastuste punktide summa ja nende väärtuste arvu jagatis. See meetod on fundamentaalne testide ja tulemuste analüüsimisel, kuna see on lihtne ja kiire viis grupi soorituse üldise taseme määramiseks. Testiküsimuste analüüsi kontekstis käsitletakse vastaja saadud punktide arvu konkreetse testiküsimuse eest juhusliku suurusena X . Juhuslik suurus on matemaatiline funktsioon, mis seostab iga võimaliku tulemuse kindla arvvaartusega [1].

Näiteks kui küsimuse eest saab maksimaalselt 5 punkti, siis X võib omandada väärtusi vahemikus 0 kuni 5. $E(X)$ on antud testiküsimuse aritmeetiline keskmine, mis arvutatakse kõikide vastuste punktide summa jagamisel vastuste arvuga.

Tulemuste standardhälve

Standardhälve on dispersiooni positiivne ruutjuur [2]. Dispersioon on juhusliku suuruse varieeruvuse mõõt keskvaartusest hälvimise mõõdu kujul, st juhusliku suuruse X ja ta keskvaartuse $E(X)$ vahe ruudu keskvaartus $D(X) = E((X - E(X))^2)$ [3]. Standardhälve näitab, kui palju andmed hajuvad keskmise ümber. Kui standardhälve on suur, siis viitab see suurele varieeruvusele andmestikus. Samas väike standardhälve tähendab, et andmed on keskmise lähedal.

Klassikaline testiteooria (*classical test theory*)

Klassikaline testiteooria on testimise teooria, mis põhineb ideel, et inimese saadud punktisumma testis on tema tegeliku taseme ja juhusliku vea summa [4]. Valemina esitatult tähendab see, et indiviidi saadud skoor X koosneb tõelisest komponendist T ehk tegelikku taset kirjeldavast osast ja juhusliku vea komponendist E ehk vea suurust kirjeldavast osast: $X = T + E$. Tõeline komponent T on antud isiku keskmine tulemus, kui tal oleks kasutada lõpmatu arv katseid, mis on praktikas saavutamatu ideaal. Vea komponent E on juhuslik viga, mis tuleneb mõõtmise ebatäpsusest, juhuslikkusest või muudest faktoritest.

Üksikvastuste teooria (*item response theory*)

Üksikvastuste teooria on teooria, mis kirjeldab, kuidas inimesed reageerivad erinevatele küsimustele või ülesannetele [5]. Allikas [6] toob välja, et üksikvastuste teooria mudeli abil saab hinnata õpilaste võimekust ja testiküsimuste raskusastet. Võimekust või vastavalt küsimuse raskusastet kirjeldab üks reaalarv, mis võib omandada ükskõik milliseid väärtusi pideval skaalal. Selle teooria idee on saada matemaatiline valem, mis põhineb eksamineeritava võimel valida oma vastuseid testiküsimustele [5]. Üksikvastuste teooria sai oma nime sellest, et ta keskendub üksikutele ülesannetele, erinevalt klassikalise testiteooria üldisest testi tasandile suunatud lähenemisest. Erinevalt lihtsamatest alternatiividest, mis eeldavad, et kõik küsimused on võrdse keerukusega, ei tee üksikvastuste teooria selliseid eeldusi.

Kõige populaarsemad üksikvastuste teooria mudelid on logistilised 1-parameetiline (Raschi mudel), 2-parameetiline (2PL) ja 3-parameetiline (3PL) mudel [7]. Mudelite parameetrite tähendust selgitatakse edaspidi eraldi jaotistes. Mudelite keskmeks on üksikvastustefunktsioon, mis hindab tõenäosust, et küsimusele vastatakse õigesti, st $X = 1$, tingimusel, et vastaja võimekus on θ .

Raschi mudel kirjeldab testiküsimusi raskuse b kaudu. Raschi mudeli üksikvastustefunktsiooni esitus valemina on $P(X = 1|\theta, b) = \frac{e^{(\theta-b)}}{1+e^{(\theta-b)}}$, kus θ on vastaja võimekus, b on testiküsimuse raskus ja e on naturaallogaritmide alus ehk Euleri arv.

2PL mudel kirjeldab testiküsimusi raskuse b ja eristusvõime a kaudu. 2PL mudeli üksikvastustefunktsiooni esitus valemina on $P(X = 1|\theta, a, b) = \frac{e^{a(\theta-b)}}{1+e^{a(\theta-b)}}$, kus θ on vastaja võimekus, b on testiküsimuse raskus, a on eristusvõime ja e on naturaallogaritmide alus.

3PL mudel kirjeldab testiküsimusi raskuse b , eristusvõime a ja oletamisparameetri c kaudu. 3PL mudeli üksikvastustefunktsiooni esitus valemina on $P(X = 1|\theta, a, b, c) = c + (1 - c) \frac{e^{a(\theta-b)}}{1+e^{a(\theta-b)}}$, kus θ on vastaja võimekus, b on testiküsimuse raskus, a on eristusvõime, c on oletamisparameeter ja e on naturaallogaritmide alus.

Punkt-biseriaalne korrelatsioonikordaja (*point-biserial correlation coefficient*, klassikalise testiteooria parameeter)

Punkt-biseriaalne korrelatsioonikordaja on Pearsoni korrelatsioonikordaja erijuht, kus üks muutuja on pidev ja teine muutuja on dihhotoomne ja nominaalne. Valemina esitus on $r_{pb} = (Y_1 - Y_0) \cdot \sqrt{(pq)} / \sigma_Y$, kus Y_1 ja Y_0 on pideva muutuja keskvärtused nende vastajate rühmas, kelle puhul dihhotoomse muutuja väärtus on vastavalt 1 või 0, p on nende vastajate osakaal, kelle puhul dihhotoomse muutuja väärtus on 1, ja $q = 1 - p$ ning σ_Y on pideva muutuja standardhälve. Tavaliselt lihtsustuvad arvutused seetõttu, et dihhotoomse muutuja puhul kasutatakse väärtusi 1 ja 0 [8]. Punkt-biseriaalset korrelatsioonikordajat kasutatakse olukordades, kus eesmärk on hinnata seost ühe pideva muutuja ja ühe dihhotoomse muutuja vahel. Näiteks pidev muutuja on testi kogutulemus ja dihhotoomne muutuja on ühe küsimuse õige või vale vastus. Kui punkt-biseriaalne korrelatsioonikordaja on positiivne, siis need vastajad, kelle üldine testitulemus on parem, vastasid ka sellele

küsimusele paremini, kuid negatiivse väärtuse puhul on seos vastupidine. Negatiivne väärtus viitab seega, et küsimuses võib olla mingi viga, sest nõrgemad vastajad vastasid sellele küsimusele paremini kui tugevamad.

Raskus (*difficulty*, üksikvastuste teooria parameeter)

Laila Issayeva [9] on kirjutanud, et raskus (*difficulty*) on üks kolmest peamisest parameetrist, mida kasutatakse üksikvastuste teorias küsimuste omaduste kirjeldamiseks. Raskus on näitaja, mis kirjeldab ülesande keerukust ehk võimekuse taset, mille juures vastajatel on 50% tõenäosus ülesandele õigesti vastata, eeldades, et juhuslikku arvamist ei toimu. Õige vastuse tõenäosus tuleneb vastajate võimekuse θ ja ülesande raskuse b vastastikusest mõjust. Mida suurem on raskuse väärtus, seda kõrgemat võimekust nõutakse vastajalt, et tal oleks 50% tõenäosus ülesande õigeks lahendamiseks. Raskus aitab modelleerida ülesande asukohta võimekuse skaalal. Näiteks kui ülesande raskuse väärtus on -1, tähendab see, et keskmisest madalama võimekusega isikud vastavad sellele tõenäolisemalt õigesti. Vastupidi, suurema väärtuse korral on küsimus raskem, nõudes kõrgemat võimekust, et saavutada sama õigete vastuste tõenäosus.

Eristusvõime (*discrimination*, üksikvastuste teooria parameeter)

Järgnev lõik tugineb Laila Issayeva artiklile [10], kus on kirjas, et eristusvõime a kirjeldab, kui hästi konkreetne testiküsimus suudab eristada vastajaid, kelle võimekus θ on erineval tasemel. Suurem a tähendab, et küsimus on eriti tundlik vastajate väikestele erinevustele võimekuses. Väiksem a viitab nõrgemale eristusvõimele, mistõttu sellised küsimused ei pruugi tõhusalt eraldada kõrge ja madala võimekusega vastajaid. Eristusvõime on oluline testide kvaliteedi hindamisel, kuna see aitab välja valida ülesandeid, mis suurendavad kogu testi mõõtmistäpsust. Allika järgi kasutatakse madala a väärtusega küsimusi sissejuhatavates küsimustes, kus eristusvõime pole esmatähtis.

Oletamisparameeter (*pseudo-guessing parameter*, üksikvastuste teooria parameeter)

Laila Issayeva artikli [11] järgi võivad vastajad mõne valikvastustega küsimuse õigesti vastata juhusliku pakkumise teel. Seetõttu võib õige vastuse tõenäosus sisaldada komponenti, mida nimetatakse oletamiseks. Oletamisparameeter c näitab tõenäosust, et väga madala võimekusega vastaja arvab ära õige vastuse lihtsalt juhusliku pakkumise teel.

Oletamisparameetri väärtus jääb vahemikku 0 kuni 1. See on oluline eriti valikvastustega küsimuste puhul, kus mitmest vastusevariandist tuleb valida üks.

Üksikvastustefunktsioon (*item response function*)

Nathan Thompson [12] kirjutab, et üksikvastustefunktsioon kasutab üksikvastuste teooria valemit, et teisendada parameetrid kõveraks, mida saab esitada graafikuna. See funktsioon on kirjeldatud valemina ülalpool. Sõltuvalt mudelist ehk parameetrite arvust on üksikvastustefunktsiooni esitus valemina erinev.

Informatsioonifunktsioon (*item information function*)

Informatsioonifunktsioon hindab üksikvastustefunktsiooni tuletist [12]. Informatsioonifunktsiooni kuju aitab aru saada, kui hästi küsimus katab oskuste vahemikku ning millise oskuste taseme juures see kõige efektiivsemalt toimib. Sõltuvalt mudelist ehk parameetrite arvust on informatsioonifunktsiooni esitus valemina erinev. Teck Kiang Tani artikli [13] järgi Raschi mudeli jaoks see on $I_i(\theta) = D^2 \cdot p(\theta) \cdot q(\theta)$, kus D on skaleerimistegur, mille väärtuseks määratakse tavaliselt 1, $p(\theta)$ on $P(X = 1|\theta, b)$ ning $q(\theta)$ on $1 - p(\theta)$. 2PL mudeli jaoks on see $I_i(\theta) = D^2 \cdot a_i^2 \cdot p(\theta) \cdot q(\theta)$, kus D on skaleerimistegur, mille väärtuseks määratakse tavaliselt 1, a on eristusvõime, $p(\theta)$ on $P(X = 1|\theta, a, b)$ ning $q(\theta)$ on $1 - p(\theta)$. 3PL mudeli jaoks on see $I_i(\theta) = \frac{D^2 \cdot a_i^2 \cdot q(\theta)}{p(\theta)} \cdot \left\{ \frac{(p(\theta) - c_i)^2}{(1 - c_i)^2} \right\}$, kus D on skaleerimistegur, mille väärtuseks määratakse tavaliselt 1, a on eristusvõime, c on oletamisparameeter, $p(\theta)$ on $P(X = 1|\theta, a, b, c)$ ning $q(\theta)$ on $1 - p(\theta)$.

3. Analüüsiks kasutatava programmi eeskujud

Enne analüüsiks kasutatava programmi loomist on oluline uurida olemasolevaid lahendusi ja meetodeid, mida kasutatakse sarnaste ülesannete täitmiseks. Käesolevas peatükis tutvustatakse analüüsiks kasutatava programmi eeskujudena tuntud meetodeid ja tööriistu. Eesmärk on välja tuua nende peamised omadused, kasutegurid ja piirangud, et luua kindel alus edasise programmi funktsionaalsuste ja ülesehituse kavandamiseks.

Moodle'i testide statistika

Moodle'i õpikeskkond sisaldab sisseehitatud tööriistu, mis võimaldavad analüüsida testide ja küsimuste statistikat (joonis 1). Allikas [16] toob välja, et statistika aruande kaudu saab hinnata, kuidas õpilased testile vastasid, tuvastada võimalikke probleemseid küsimusi ning parandada üldist testi kvaliteeti. Moodle'i testi statistika aruanne võimaldab ka analüüsida individuaalsete küsimuste vastuseid, tuues esile, kuidas õpilased iga küsimuse puhul vastasid. See aitab tuvastada küsimusi, mis võivad olla liiga lihtsad, liiga rasked või mitte piisavalt eristavad, võimaldades õpetajatel teha vajalikud kohandused testi kvaliteedi parandamiseks. Moodle'i statistilises aruandes arvutatakse ainult lihtsamaid kirjeldava statistika mõõdikuid, aga ei kasutata üksikvastuste teooriat. Samuti ei ole seal graafikuid.

Testi struktuuri analüüs										
Salvesta tulemuste tabel kui <input type="text" value="Komaeraldusega fail (.csv)"/> <input type="button" value="Laadi alla"/>										
Knr	Küsimuse nimi	Katsete arv	Facility index (Küsimuse keskmine skoor protsentides. Mida kõrgem see indeks on, seda lihtsam oli nende vastajate jaoks see küsimus)		Standardhälve	Juhusarvamise punktisumma	Plaanitud kaal	Tegelik kaal	Discrimination index (Näitab seost testi tulemuse ja konkreetse küsimuse tulemuse vahel)	
1	Juhuslik küsimus	346		97.25%	15.69%		10.00%	6.64%	20.34%	33.08%
	Statistikavahemik nende küsimuste jaoks	346 – 346		97.25% – 97.25%	15.69% – 15.69%		10.00% – 10.00%		32.73% – 32.73%	54.12% – 54.12%
	Vaata detaile									
2	Juhuslik küsimus	346		94.80%	22.24%		10.00%	9.39%	29.73%	44.39%
	Statistikavahemik nende küsimuste jaoks	171 – 175		92.57% – 97.08%	16.90% – 26.30%		10.00% – 10.00%		33.95% – 52.56%	61.26% – 71.57%
	Vaata detaile									
3	Juhuslik küsimus	346		88.73%	27.24%		10.00%	11.91%	43.07%	50.30%
	Statistikavahemik nende küsimuste jaoks	164 – 182		85.98% – 91.21%	23.61% – 30.61%		10.00% – 10.00%		54.13% – 64.29%	64.66% – 74.09%
	Vaata detaile									
4	Juhuslik küsimus	346		95.38%	19.97%		10.00%	8.62%	28.45%	41.23%
	Statistikavahemik nende küsimuste jaoks	48 – 71		91.15% – 98.59%	11.87% – 27.04%		10.00% – 10.00%		12.71% – 71.65%	24.28% – 82.42%
	Vaata detaile									

Joonis 1. Sisseehitatud tööriist Moodle'i õpikeskkonnast.

Cronbachi alfa kalkulaatorid

Jim Frost [14] on kirjutanud, et Cronbachi alfa (Cronbach's alpha) on statistiline mõõdik, mis hindab küsimustike ja testide sisemist kooskõllalisust ehk seda, kui hästi mõõdavad erinevad küsimused sama omadust või konstrukti. See koefitsient aitab kindlaks teha, kas küsimuste kogum annab usaldusväärseid ja stabiilseid tulemusi. Valemina esitus on $\alpha = (N \cdot \bar{c}) / (\bar{v} + (N - 1) \cdot \bar{c})$, kus N on küsimuste arv, \bar{c} on küsimustevaheliste kovariatsioonide keskmine ning \bar{v} on keskmine dispersioon. α väärtus asub lõigus 0 kuni 1, kus väärtus 0 tähendab, et küsimuste vahel puudub korrelatsioon ehk need on täiesti sõltumatud, ja väärtus 1 tähendab, et ühe vastuse põhjal saab täielikult ennustada teisi vastuseid. Tulemust $\alpha \geq 0,7$ peetakse sageli usaldusväärseks ning tulemus $\alpha \geq 0,95$ võib viidata küsimuste liigsele sarnasusele. Kasutades Cronbachi alfa valemit, on loodud erinevaid kalkulaatori kujul programme (joonis 2), mida kasutatakse laialdaselt uute küsitluste või hindamisvahendite väljatöötamisel ja testimisel.

Online Statistics Calculator 100% data security

Enter new Data Export / Import Transform Data Settings

Case	nominal	metric	metric	nominal	metric	nominal	ordinal			
1	Female	1500	33	Chicago	80	BMW	Bachelor			
2	Female	1200	33	Chicago	82.5	Ford	No			
3	Male	2200	34	New York	100.8	BMW	Bachelor			
4	Male	2100	42	New York	90	BMW	Master			
5	Female	1500	29	Chicago	67	Ford	Master			
6	Female	1700	19	Washington	60	Ford	Master			
7	Male	3000	50	Washington	77	Ford	No			
8	Male	3000	55	Washington	77	Ford	Bachelor			
9	Female	2800	31	New York	87	Ford	Bachelor			
10	Male	2900	46	New York	70	GM	Master			
11	Female	2780	36	Washington	57	BMW	No			
12	Male	2550	48	New York	64	GM	Master			
13										
14										

Descriptive Charts Hypothesis tests Correlation Regression ANCOVA Mediation/Moderation PCA Reliability Cluster

Metric Variables

Salary Age Weight

Ordinal Variables

Academic degree

Nominal Variables

Gender Place Company

Calculate

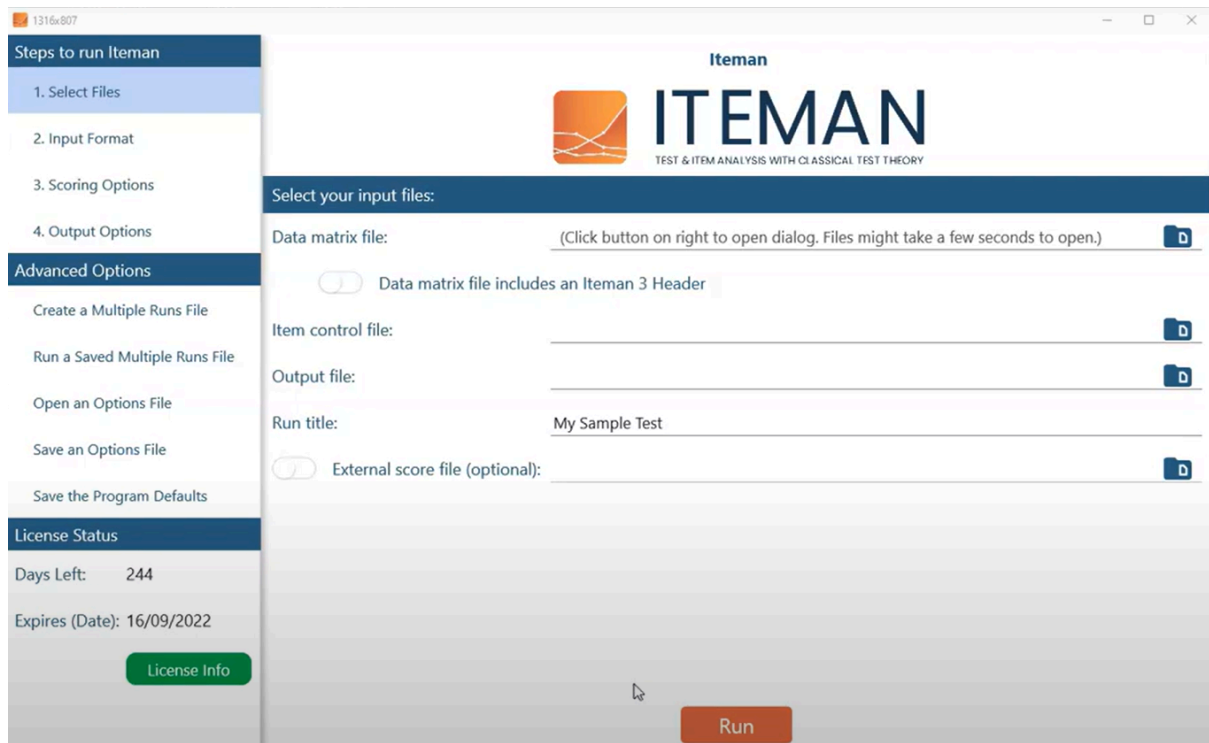
Cronbach's alpha Intra-Class-Correlation

Joonis 2. Näide Cronbachi alfa kalkulaatorist [15].

Iteman

Allikas [17] kirjutab, et Iteman on tarkvaraplatvorm, mis automatiseerib psühhomeetriliste aruannete koostamist, kasutades klassikalist testiteooriat (joonis 3). Tarkvara ei piirdu ainult statistiliste näitajate arvutamisega, vaid loob automaatselt põhjaliku tehnilise aruande, mis sisaldab graafikuid, tabeleid ja selgitavaid tekste, aidates tuvastada liiga keerulisi või segaseid küsimusi. See sobib ideaalselt klassikalise testiteooria õpetamiseks või õppimiseks ning

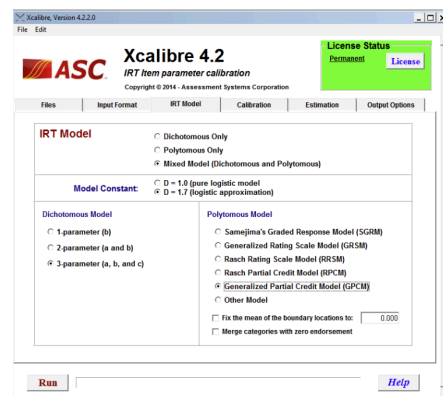
väiksema mahuga testide analüüsimiseks. Tarkvara aitab tuvastada ja märgistada küsimusi, mis ei vasta standarditele, võimaldades parandada testi üldist kvaliteeti.



Joonis 3. IteMan'i kasutajaliides [17].

Xcalibre

Allika [18] järgi on Xcalibre kasutajasõbralik tarkvara, mis lihtsustab üksikvastuste teooria analüüsi, võimaldades automaatselt luua põhjalikke tehnilisi aruandeid (joonis 4). Tarkvara esitab tulemused tabelite, graafikute ja selgitavate narratiivide kujul, hõlbustades tulemuste mõistmist ja edasist analüüsi. Xcalibre toetab kõiki peamisi üksikvastuste teooria mudeleid. See on võimas tööriist, mis muudab üksikvastuste teooria analüüsi kättesaadavaks ka väiksemate eelteadmistega kasutajatele.



Joonis 4. Xcalibre'i kasutajaliides [18].

4. Nõuete kirjeldus

Programm hakkab analüüsima testi tulemusi küsimuste kaupa. Esiteks peab olema võimalus programmile ette anda andmeid, mille põhjal programm teeb arvutusi. Õpikeskkonnas Moodle saab alla laadida tabeli lähteandmetega csv-vormingus, mille põhjal saab arvutada konkreetse testi kohta vastuste arvu, tulemuste keskmise, tulemuste standardhälbe ja kus on kirjas teised andmed statistiliste meetodite jaoks. Selline tabel on programmi sisendiks ning programm arvutab välja mitmesugused statistilised suurused, mis küsimusi iseloomustavad. Tulemusena tagastab programm kasutajale PDF-faili analüüsiga, kus iga küsimuse kohta on kirjas vastavad statistilised näitajad ja graafikud. Kasutajal tuleb valida Google Drive'i kaust, kus asub fail, milles asuvad andmed õpikeskkonnast Moodle. Programm palub kasutajal veenduda, et Moodle'ist allalaaditud csv-fail nimega 'andmed.csv' on valitud Google Drive'i kaustas. Kui etteantud fail ei eksisteeri valitud Google Drive'i kaustas või faili nimi ei ole 'andmed.csv', siis annab programm kasutajale veateate selle kohta ning palub proovida uuesti. Programm töötab igasuguse Moodle'ist allalaaditud failiga, sealhulgas ka näiteks tühja failiga või failiga, kus on ainult üks testikatse. Kui faili ei ole võimalik lugeda, siis annab programm kasutajale sobiva teate.

Iga näitaja jaoks on olemas piir, mille ületamine näitab, et küsimuses esineb mingi probleem. Programm annab võimaluse neid piire muuta vastavalt kasutaja soovile või valida vaikimisi valitud piiride komplekti. Täpsemalt palub programm kasutajal valida, kas ta soovib töötada vaikimisi valitud piiride komplektiga või ise sisestada piire ühekaupa ettenähtud reale. Kui kasutaja proovib sisestada ebasobivat arvu või muud, näiteks teksti, siis annab programm kasutajale veateate selle kohta ning palub sisestada arvu uuesti, kuni valiidne arv on sisestatud. Probleemsed väärtused märgib programm ära tagastatud failis, et kasutajale oleks selge, mis näitaja ületas piiri konkreetse küsimuse korral.

Üksikvastuste teooriaga seotud analüüsi jaoks tuleb Moodle'ist allalaaditud andmed teisendada binaarseteks. Selleks tuleb kasutajal programmis määrata lävend, millest alates loetakse vastus õigeks. Lävend peab jääma vahemikku 0.00 kuni 0.05. Kui kasutaja proovib sisestada ebasobivat arvu või muud, näiteks teksti, siis annab programm kasutajale veateate selle kohta ning palub sisestada arvu uuesti, kuni valiidne arv on sisestatud.

Lisaks sellele analüüsib programm õpilaste esimese katse ja parima katse tulemusi. Nende tulemuste kombinatsioon aitab välja tuua probleemseid küsimusi. Täpsemalt, kui esimene katse on hea ning esimese ja parima katse vahe on väike, siis see tähendab, et õpilasel oli juba

enne testi lahendamist teadmisi küsimuse kohta ja võib öelda, et küsimus on liiga lihtne õpilaste jaoks. Samas, kui esimene katse on halb ning esimese ja parima katse vahe on väike, siis see tähendab, et küsimus ei anna õpilastele teadmisi ja võib öelda, et küsimusest ei ole õpilase jaoks õppimisel kasu. Näitajate väärtused arvutatakse üks kord osalejate esimeste katsete põhjal ja teine kord parimate katsete põhjal.

Eelmainitud näitajate piiride põhjal oskab programm tuvastada probleemseid küsimusi. Programm tagastab faili, kus on tekstina kirjas selgitused, miks küsimus on probleemne ning mis näitaja on sobimatu väärtusega.

Programmi graafiline osa on üksikvastuste teooriaga seotud graafikute joonistamine. Graafikud aitavad aru saada, kas küsimus katab laia või kitsa oskuste skaala või millise oskuste taseme puhul küsimus kõige paremini töötab. Tagastatud fail on PDF-kujul. Graafikute joonistused asuvad tagastatud failis ning nende lisamiseks faili on kasutusel teek nimega Matplotlib.

Programm on eraldiseisev ning programmil on tekstiline kasutajaliides, mille kaudu kasutaja saab programmile ette anda andmeid ning saada tagasi tulemusi.

Kokkuvõttes alustab programm teatega kasutajale kujul „Veenduge, et Moodle'ist allalaaditud csv-fail nimega 'andmed.csv' on üleval valitud Google Drive kaustas. Faili nimi peab olema 'andmed.csv'. Muidu loeb programm faili ebavaliidseks.”. Kui faili nime kontrollimisel tuvastatakse, et see ei ole valiidne või fail ei eksisteeri valitud Google Drive'i kaustas, kuvab programm veateate: „Faili nimega 'andmed.csv' ei leitud! Proovige uuesti.”. Muidu kuvab programm teate: „Fail 'andmed.csv' loetud edukalt!”.

Kui fail on valideeritud, siis analüüsib programm selle sisu. Kui fail on tühi, kuvab programm veateate: „Sisendfail on tühi. Kontrollige failis sisalduvaid andmeid ja proovige uuesti.” või kui fail on mittedobiv, kuvab programm veateate: „Viga faili lugemisel: '{vea_nimi}'. Proovige uuesti.” ning seejärel programmi töö lõpeb.

Kui fail on sobilik, siis järgmine programmi teade on „Palun valige: vaikimisi valitud piiride komplekt (1) või sisestage piirid käsitsi (2): ”. Kasutajal tuleb sisestada arv 1 või arv 2 ning kui on sisestatud midagi muud, siis kuvab programm teate: „Sisestatud väärtus ei ole sobiv. Tuleb sisestada arv 1 või arv 2. Proovige uuesti.”. Programm kordab küsimist, kuni on sisestatud arv 1 või arv 2 või programmi töö katkestatakse.

Kui on sisestatud arv 1, siis programm kasutab vaikimisi valitud piire ning kuvab teate: „Kasutatakse vaikimisi piirväärtusi. Kasutatavad piirväärtused: '{piiride_väärtuste_loetelu}'”.

Kui kasutaja valik oli arv 2, siis esitatakse ükshaaval küsimusi iga piiri kohta kujul: „Sisestage näitaja '{piiri_nimi}' piir ('{piiri_kirjeldus}'): ”. Kui kasutaja sisestab sobimatu väärtuse, kuvab programm veateate: „Sisestatud väärtus ei ole sobiv. Proovige uuesti.”. Programm kordab küsimist, kuni kasutaja sisestab kehtiva väärtuse iga näitaja jaoks või katkestab programmi töö. Kui kõik piirid on valitud, siis kuvab programm teate: „Kasutatakse käsitsi sisestatud piirväärtusi. Kasutatavad piirväärtused: '{piiride_väärtuste_loetelu}'”.

Selleks, et teisendada Moodle'ist allalaaditud andmed binaarseteks, küsib programm kasutajalt lävendit, millest alates loetakse vastus õigeks kujul „Sisestage lävend, millest alates loetakse vastus õigeks (nt 0.025): ”. Kui kasutaja sisestab sobimatu väärtuse, kuvab programm veateate: „Lävend peab jääma vahemikku 0.00 kuni 0.05. Proovige uuesti.”. Kui on sisestatud midagi muud, siis kuvab programm teate: „Sisestus ei ole korrektne. Proovige uuesti.”. Programm kordab küsimist, kuni kasutaja sisestab sobiva väärtuse.

Seejärel väljastab programm teate „Analüüsitakse küsimusi, esimese ja parima katse tulemusi ning joonistatakse graafikuid. Palun oodake...”. Programm kirjutab: „PDF koos analüüsi ja graafikutega salvestatud: analüüs_koos_graafikutega.pdf”. Tulemusena loob programm kasutaja Google Drive'i kausta PDF-faili, kus iga küsimuse kohta on kirjas statistilised näitajad koos selgitustega probleemsete küsimuste kohta ning graafikutega, mille põhjal saab kasutaja teha analüüsi.

5. Süsteemi ülesehitus

Programmi loomiseks kasutatakse Pythoni programmeerimiskeelt ning keskkonda Google Colab, mis on eriti populaarne ja mugav juhul, kui põhieesmärk on andmetöötlus. See sobib programmi loomiseks, kuna Python võimaldab kasutada mooduleid, mis annavad võimaluse mugavalt töötada õpikeskkonnast Moodle pärit andmetega ja Google Colab võimaldab alla laadida neid mooduleid ilma keeruliste seadistusteta, tagades, et programm töötab enamikel arvutitel. Lisaks sellele on olemas moodulid, mis on konkreetselt seotud analüüsiks kasutatavate statistiliste meetoditega ehk klassikalise testiteooriaga ning üksikvastuste teooriaga ja vastavate parameetritega. On plaanis kasutada järgmisi mooduleid: Os, Pandas, Sys, NumPy, SciPy, Google Colab Drive, PyIrt ja Matplotlib.

Google Colab

Google Colab on Google'i poolt pakutav pilvepõhine platvorm, mis võimaldab Pythoni koodi kirjutamist ja käivitamist otse veebibrauseris. See keskkond on eriti kasulik andmetealastele, masinõppe uurijatele ja programmeerijatele, kes soovivad kasutada võimsaid arvutusressursse ilma keerulise seadistamiseta. Colab toetab Jupyter Notebooki formaati, võimaldades kasutajatel kombineerida koodi, teksti, matemaatilisi valemeid ja visualiseeringuid ühes dokumendis. Lisaks ühildub see sujuvalt Google Drive'iga, võimaldades failide hõlpsat salvestamist ja jagamist.

Os

Pythoni sisseehitatud moodul, mis pakub operatsioonisüsteemi spetsiifiliste funktsioonide kasutamiseks ühtset liidest. See võimaldab arendajatel suhelda failisüsteemi ja keskkonnamuutujatega, hallata protsesse ning teostada muid süsteemitaseme toiminguid, ilma et oleks vaja kirjutada platvormispetsiifilist koodi. Käesolevas programmis kasutatakse moodulit eelkõige failisüsteemiga suhtlemiseks ning failide olemasolu kontrollimiseks.

Pandas

Andmetöötluseks ja analüüsiks mõeldud teek, mis võimaldab töödelda ja analüüsida tabelandmeid. Programmi raames saab seda kasutada CSV-failide lugemiseks ja kirjutamiseks, andmete filtreerimiseks ning rühmitamiseks.

Sys

Moodul Sys võimaldab suunata programmi käitumist ja suhtlemist süsteemi keskkonnaga. Moodul pakub erinevaid funktsioone, mis võimaldavad käsitleda näiteks käsurea argumente, standardväljundit ja -sisendit, süsteemiteavet ning katkestada programmide täitmist. See on sisseehitatud moodul, mis ei vaja täiendavat paigaldamist ning kuulub Pythoni standardteeki. Käesolevas programmis kasutatakse seda moodulit juhul, kui vajalikku sisendfaili ei leita. Kui faili pole, siis mooduli abil lõpetatakse programmi töö.

NumPy

Matemaatiliste ja statistiliste arvutuste teek, mis pakub tuge mitmemõõtmeliste massiividele ja erinevatele matemaatilistele funktsioonidele. Programmi raames saab seda kasutada aritmeetilise keskmise ja standardhälbe arvutamiseks.

SciPy

Teaduslike ja tehniliste arvutuste teek, mis sisaldab laia valikut statistilisi funktsioone. Programmi raames saab seda kasutada punkt-biseriaalse korrelatsioonikordaja arvutamiseks.

Google Colab Drive

Google Colabi keskkonnas kasutatav moodul `google.colab`, täpsemalt selle alam moodul `google.colab.drive`, võimaldab suhelda Google Drive'i failisüsteemiga, võimaldades kasutajatel paigaldada oma pilveteek Colabi tööruumi. Moodul kuulub Google Colabi keskkonna spetsiifiliste tööriistade hulka ja ei ole osa Pythoni standardteegist, mistõttu saab seda kasutada vaid Colabi keskkonnas. Programmi raames luuakse ühendus kasutaja Google Drive'i kontoga, mille tulemusel seotakse Drive'i failisüsteem Colabi tööruumi konkreetse kaustaga. Selline lähenemine võimaldab kasutajal töötada otse oma Google Drive'i salvestatud andmefailidega.

PyIrt

Üksikvastuste teooria mudelite rakendamiseks mõeldud teek, mis võimaldab hinnata küsimuste raskust ja eristusvõimet. Programmi raames saab seda kasutada üksikvastuste teooria meetodite rakendamiseks.

Matplotlib

Matplotlib on teek, mis võimaldab luua mitmesuguseid visualiseeringuid, sealhulgas staatilisi, animeeritud ja interaktiivseid graafikuid. Programmi raames saab seda kasutada üksikvastustefunktsiooni ning informatsioonifunktsiooni graafikute loomiseks.

6. Realiseerimise protsess

Selle lõputöö praktilise osa käigus loodi analüüsirakendus, mille nõuded on kirjeldatud 4. peatükis ja kasutatud tehnoloogiad 5. peatükis. Käesolev peatükk kirjeldab realiseerimise protsessi ehk kuidas tarkvara loomine käis, mis raskusi tekkis ja kuidas need ületati ning millised olid praktilise osa kriitilised kohad. Arendusprotsess jagunes mitmeks loogiliseks etapiks, millest igapähele olid omad tehnilised ja meetoodilised väljakutsed ning lahendused.

Arendusloogika ja funktsionaalsused

Esmalt oli vaja CSV-vormingus sisendfail programmile ette anda ning ära valideerida. Arendus viidi läbi Google Colab'i keskkonnas. See võimaldas lihtsat juurdepääsu andmetele läbi Google Drive'i.

Järgmisena tuli andmete töötlemine statistiliste näitajate arvutamiseks. Esialgsetes andmetes esinesid tühjad väärtused ning ebavajalikud veerud.

Selle järel tuli statistiliste näitajate piiride määramise osa. Sellel etapil tuli valida vaikumisi valitud piiride väärtused (joonis 5) ning lisada funktsionaalsus, et kasutaja saaks ise neid väärtusi valida. Piiride vaikeväärtused valiti järgmiselt. Madalaim lubatud keskmine on 0.01, kuna maksimaalne võimalik väärtus ühe küsimuse kohta oli 0.05 ja 0.01 moodustab 20% võimalikust tulemusest. Samal põhjusel kõrgeim lubatud keskmine on 0.04, mis tähendab, et keskmiselt saadud tulemus küsimuse eest on 80%. Madalaim standardhälve on 0.005. Kui standardhälve on alla 0.005, siis tulemuste varieeruvus on alla 10% kogu skaala ulatusest. Kõrgeim standardhälve on 0.015. See väärtus on 30% maksimaalsest võimalikust väärtusest. Minimaalne punkt-biseriaalne korrelatsioon on 0.2, kuna allikas [19] kirjutab, et väärtused vahemikus 0 kuni 0.19 võivad viidata sellele, et küsimus ei erista vastajaid piisavalt hästi. Allika [20] järgi jäävad raskusparameetri tüüpilised väärtused vahemikku -3 kuni +3. Väärtused, mis on -3 lähedal, vastavad väga lihtsatele küsimustele, samas väärtused, mis on +3 lähedal, viitavad küsimustele, mis on õpilaste jaoks väga rasked. Seega madalaim raskusparameeter üksikvastuste teooria mudelis on -3 ning kõrgeim raskusparameeter üksikvastuste teooria mudelis on 3. Allikas [21] on kirjutanud, et eristusparameeter jääb tavaliselt vahemikku umbes 0.5 kuni 2. Selle põhjal valitud minimaalne eristusparameeter üksikvastuste teooria mudelis on 0.5.

```
# Vaikimisi piirid
default_thresholds = {
    "keskmine_alumine": 0.01,
    "keskmine_ylemine": 0.04,
    "std_alumine": 0.005,
    "std_ylemine": 0.015,
    "min_item_total_corr": 0.2,
    "difficulty_lower": -3,
    "difficulty_upper": 3,
    "discrimination_min": 0.5
}
```

Joonis 5. Vaikimisi valitud piiride väärtused.

Järgmisena toimus analüüsi osa, kus iga küsimuse kohta arvutati statistilised näitajad, sealhulgas aritmeetiline keskmine, standardhälve, punkt-biseriaalne korrelatsioonikordaja ning üksikvastuste teooria mudeli parameetrid. Kasutati 2PL mudelit, kus iga küsimuse kohta hinnati raskust ja eristusvõimet. Nende parameetrite jaoks sooritati andmete ettevalmistus, mille käigus teisendati andmed binaarseteks.

Järgmises etapis teostati õpilaste esimese katse ja parima katse tulemuste analüüs. Väikese edasimineku piiriks määrati 0.01, väga hea esmase tulemuse piiriks 0.045 ning väga kehv esmase tulemuse piiriks 0.015 (joonis 6). Küsimus loetakse liiga lihtsaks, kui õpilaste keskmine tulemus esimesel katsel ületab 0.045 ja erinevus esimese ning parima katse vahel jääb alla 0.01. Õppimist mitte toetavaks loetakse küsimus, mille puhul esmane keskmine tulemus jääb alla 0.015 ning pärast parimat katset ei ole näha sisulist edasiminekut ehk tulemuste erinevus on alla 0.01.

```
# Lävi väikese vahe jaoks (nt kui erinevus on < 0.01, siis pole edasiminekut)
small_diff_threshold = 0.01
high_first_threshold = 0.045 # väga hea esmane tulemus
low_first_threshold = 0.015 # väga kehv esmane tulemus
```

Joonis 6. Õpilaste esimese katse ja parima katse tulemuste analüüsi piiride väärtused.

Analüüside järel tuli probleemsete küsimuste tuvastamine. Kui mõni näitaja ületas eelnevalt määratud piirväärtused, siis genereeris programm selle kohta selgituse, mis aitab tuvastada võimalikke probleeme küsimuse kvaliteedis. Teine selgitus lisati varem teostatud esimese ja parima katse analüüsi põhjal.

Viimasel etapil koostati PDF-fail, mis sisaldab iga küsimuse kohta nii tekstilist analüüsi kui graafilist visualiseeringut. Iga testiküsimuse kohta genereeriti kaks järjestikust lehekülge. Esimesel lehel kuvatakse küsimuse statistilised näitajad ja probleemsete küsimuste tuvastamise jooksul loodud selgitused selliste küsimuste jaoks. Teisel lehel visualiseeriti iga küsimuse üksikvastustefunktsioon ja informatsioonifunktsioon.

Tehnilised raskused ja nende lahendamine

Arendusprotsessi käigus ilmnis mitmeid raskusi.

Esimene väljakutse oli seotud andmete failis esinevate puuduvate väärtustega. Moodle'ist allalaaditud fail sisaldas mitmeid tühje või mittetäisarvulisi kirjeid. Selliste väärtustega töötamine võib põhjustada vigu andmetöötluses. Lahendamiseks loodi protsess, mille käigus asendati kõik tühjad väärtused väärtustega NaN ning teisendati vastavad veerud ujukomaarvudeks, et võimaldada täpseid arvutusi.

Järgmine probleem oli seotud sellega, et osad moodulid PyIrt teegist eeldasid binaarseid andmeid, aga Moodle'ist eksporditud testitulemused olid osalise skooriga (näiteks 0.03 maksimaalsest 0.05-st). Selle lahendamiseks küsiti kasutajalt lävendit, millest alates loetakse vastus õigeks ning teisendati andmeid binaarseteks.

Arendusprotsessis tekkis raskusi 3-parameetrilise (3PL) mudeli rakendamisel. Algne plaan oli kasutada analüüsis oletamisparameetrit. Praktikas ilmnis, et kasutatud PyIrt teek ei võimaldanud korrektselt 3PL mudelit sobitada. Katsed lisada oletamisparameeter tekitasid mudeli loomisel vigu ja takistasid küsimuste parameetrite korrektset hindamist. Seetõttu otsustati lõplikus lahenduses piirduda 2-parameetrilise (2PL) mudeliga, kus iga küsimust iseloomustati ainult raskusparameetri ja eristusparameetri kaudu.

Viimane raskus puudutas PDF-faili loomist. Algne idee oli kasutada moodulit FPDF, aga see ei suutnud töödelda pikki ridu. Probleemi lahendamiseks asendati moodul FPDF mooduliga Matplotlib, mis võimaldas igale küsimusele lisada nii pikad tekstilised selgitused analüüsi kohta kui ka graafilised kujutised.

7. Testimine

Arendatud analüüsirakenduse töökindluse ja korrektsuse kontrollimiseks viidi läbi mitmekülgne testimine. Testimise eesmärgiks oli veenduda, et rakendus töötab korrektselt erinevate sisendandmetega ning käsitleb ootamatuid olukordi informatiivselt.

Esiteks testiti rakendust käesoleva aasta programmeerimise kursuselt pärit andmetega. Õpikeskkonnas Moodle on 16 erineva testi tulemused, millest valiti juhuslikult 5 faili ja analüüsiti neid loodud rakenduse abil. Kõigi viie faili puhul genereeris rakendus korrektsed PDF-failid, mis sisaldasid igale küsimusele vastavaid statistilisi näitajaid, selgitusi ning graafikuid. Selle testimise käigus selgus, et olulise osa küsimuste puhul oli keskmine tulemus väga kõrge, mis viitab sellele, et küsimus oli liiga lihtne. Standardhälve oli paljudel juhtudel alla 0.015, mis tähendab, et kõik vastajad sooritasid küsimuse sarnaselt. Lisaks esines üsna sageli olukordi, kus õpilaste esimese ja parima katse tulemused ei erinenud.

Üks andmefail põhjustas analüüsi katkestuse. Selle struktuur erines oluliselt teistest testidest. Test sisaldab 15 küsimust, kus iga küsimuse kohta antakse 0.03 punkti, võrreldes eeldatud 10 küsimuse ja 0.05 punktiga nagu kõikides teistes testides. Lisaks ainult selles testis on palju küsimusi kursuse struktuuri ja korralduse kohta, mitte materjali kohta. Seega tulevikus võiks rakendust täiendada, et see suudaks analüüsida erineva struktuuriga faile paindlikumalt.

Teiseks testiti failinime valideerimise mehhanismi. Muudetud failinimega katse näitas, et rakendus suudab õigesti tuvastada vale failinime ja lõpetab töö ning kuvab selge veateate, et faili ei leitud.

Kolmandaks kontrolliti failide sisu korrektsuse valideerimist. Testiti vigase sisuga faili ning tühja faili analüüsi. Vigase faili puhul andis rakendus vastava veateate lugemise ebaõnnestumise kohta ning tühja faili korral kuvati korrektselt teade, et failis puuduvad andmed.

Neljandaks hinnati piiride määramise funktsionaalsust. Kasutaja saab valida vaikimisi valitud piiride ja käsitsi määratud piiride vahel. Testimisel kinnitati, et vale sisendi korral programm esitab küsimuse uuesti ning jätkab, kuni kasutaja sisestab kas arvu 1 või 2. Vaikimisi valitud piiride korral kasutas rakendus korrektselt eelmääratud väärtusi ning käsitsi määramise korral suunati kasutaja õigesti vastavate väärtuste sisestamisele.

Viiendaks kontrolliti käsitsi piiride sisestamise protsessi. Kui kasutaja sisestas vigase väärtuse, siis palus programm uut sisendit kuni kehtiva väärtuse saamiseni.

Kuuendaks testiti andmete binaarseteks tegemise protsessi. Programm küsib lävendi väärtust, mis peab jääma vahemikku 0.00 kuni 0.05. Kui kasutaja sisestas arvu, mis on vahemikust väljas, siis programm andis teada, et sisestus on õige, aga valitud arv peab jääma vahemikku ning nõudis uut sisestust. Samas kui kasutaja sisestas kohe mitteamulise väärtuse, siis programm andis teada, et sisestus ei ole õige ning samamoodi nõudis uut sisestust. Lisaks läbiviidud katsed kinnitasid, et erinevad lävendi valikud mõjutasid erinevalt analüüsi tulemusi ja lõpptulemusena genereeritud PDF-faile.

Kokkuvõttes kinnitas läbiviidud testimine, et arendatud analüüsirakendus töötab ettenähtud viisil, suudab edukalt käsitleda erinevaid sisendandmeid ja võimalikku vigast sisestust.

8. Arutelu

Tulemus

Töö tulemusena loodi rakendus, mis võimaldab põhjalikult analüüsida testide küsimuste kvaliteeti. Enne arendustööd piirdus Moodle'i sisseehitatud statistika ainult keskmiste ja standardhälvete esitamisega, aga loodud rakendus võimaldab kasutada klassikalise testiteooria ja üksikvastuste teooria parameetreid (joonis 7). Veel üks oluline lisandus on õpilaste esimese katse ja parima katse tulemuste analüüs, mille abil võib tuvastada, et küsimus on liiga lihtne õpilaste jaoks või küsimus ei anna õpilastele teadmisi (joonis 7). Lisaks võimaldab rakendus visualiseerida iga küsimuse üksikvastustefunktsiooni ja informatsioonifunktsiooni, mis toetab probleemsete küsimuste tuvastamist märksa täpsemalt (joonis 8 ja joonis 9). Ühe testi analüüs võtab tüüpiliselt aega kuni 1 minut. Tulemusena suurenes võimalus hinnata küsimuste kvaliteeti sügavamalt.

Küsimus: K. 1 /0.05

Keskmine: 0.048, mis on 96.1%

Standardhälve: 0.005

Item-total korrelatsioon: 0.622

IRT raskus (b): 2.000

IRT eristus (a): 2.000

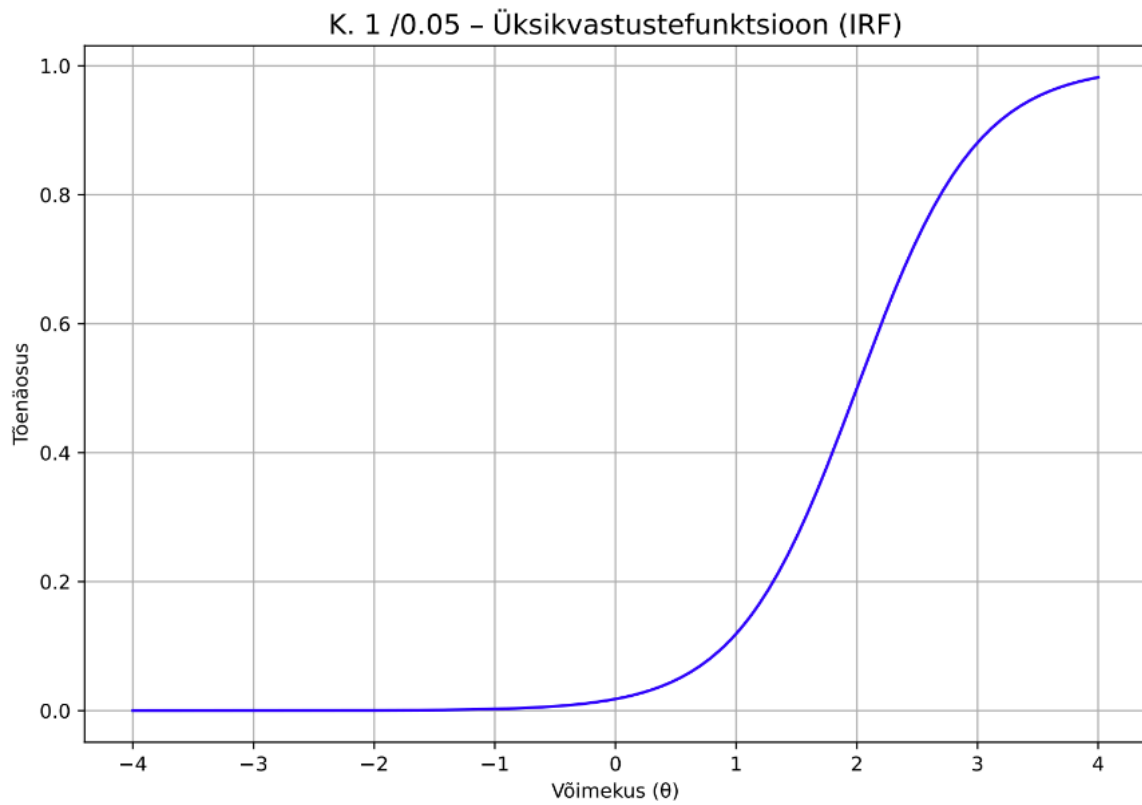
Piiride põhjal:

Keskmine (0.048) on liiga kõrge → küsimus võib olla liiga lihtne.

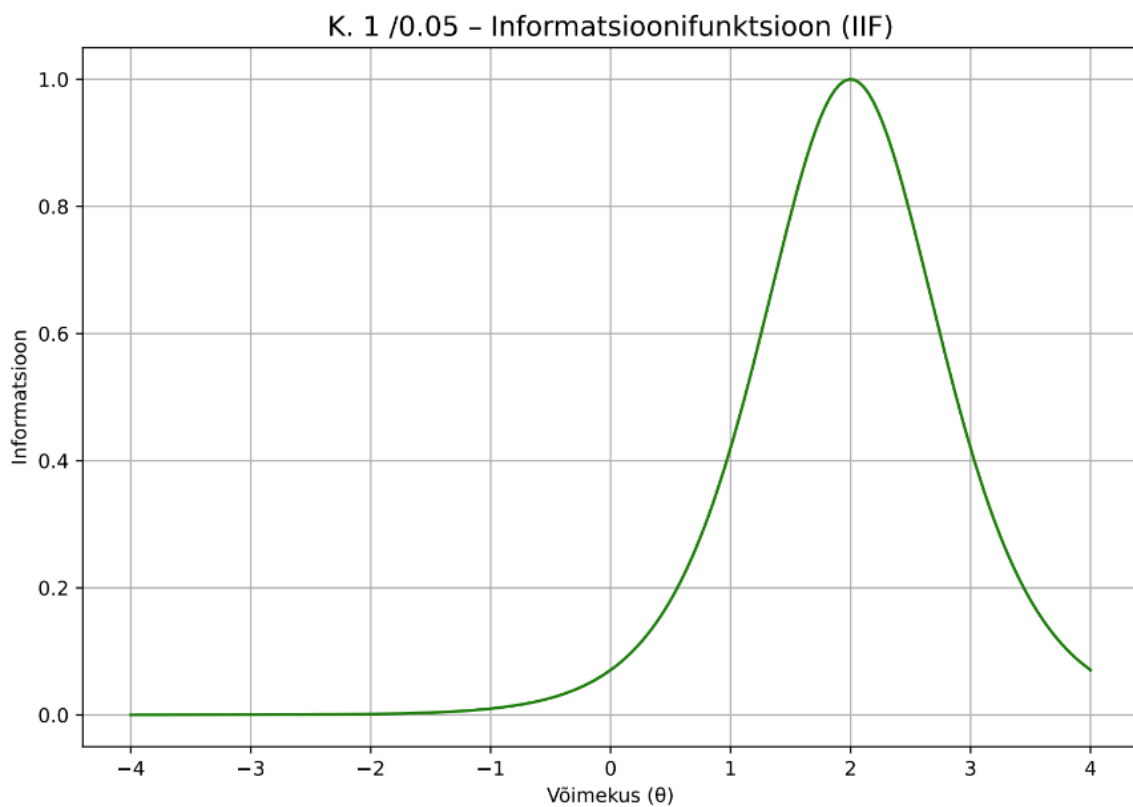
Esimese-parima põhjal:

Õpilased said küsimusele kohe hästi pihta → küsimus võib olla liiga lihtne.

Joonis 7. Programmi genereeritud PDF-faili esimene lehekülg.



Joonis 8. Programmi poolt genereeritud üksikvastustefunktsiooni graafik.



Joonis 9. Programmi poolt genereeritud informatsioonifunktsiooni graafik.

Rakenduse testimise käigus töötati läbi 5 nädalatesti, milles kokku oli hinnatud 50 küsimust. Kvaliteedi hindamisel selgus, et märkimisväärne osa küsimustest (ligikaudu 60–70%) osutus liiga lihtsaks, kuna nende keskmine tulemus ületas piirväärtuse. Lisaks esines mitmel juhul väga väikest standardhälvet, mis tähendab, et kõik vastajad sooritasid küsimuse sarnaselt ning esines mitmeid küsimusi, mille standardhälve oli liiga suur, viidates sellele, et vastuste hajuvus oli liialt suur ja tulemused on ebatühtlased.

Esimese ja parima katse analüüs näitas, et sageli vastasid õpilased küsimustele õigesti juba esimesel katsel, mis viitab sellele, et küsimus ei toetanud uut õppimist või teadmiste kinnistamist.

Seega võib üldistatult öelda, et testides sisaldus küsimusi, mille õppimist toetav väärtus oli küsitav ning täpsemate järelduste tegemiseks tuleks sooritada põhjalikum analüüs.

Rakenduse kasutamine

Rakendust saab edaspidi kasutada testiküsimuste süstemaatiliseks hindamiseks ja parendamiseks. Juhend rakenduse jaoks asub lisa I. Järgmine loogiline samm oleks teostada analüüs rakenduse abil kõigi 15 erineva testi kohta ning dokumenteerida tuvastatud probleemsed küsimused. Kui probleemsed küsimused on parandatud, siis võiks korrata analüüsi, et aru saada, kas tulemus läks paremaks küsimuste parandamisega või mitte.

Piirangud ja edasiarendamise võimalused

Rakendusel esineb piiranguid. Praegune versioon eeldab, et testis on 10 küsimust, kus iga küsimuse maksimaalne võimalik punktisumma on 0.05 ning kogu testi maksimaalne võimalik punktisumma on 0.5. Mõne teise struktuuriga testide korral võib programm katki minna. Samuti jäi realiseerimata 3-parameetrilise (3PL) mudel oletamisparameetriga, kuna kasutatud PyIrt teek ei võimaldanud seda stabiilselt rakendada.

Edasiste arenduste võimalusi on mitmeid. Esiteks võiks arendada funktsionaalsust, mis võimaldab automaatselt kohandada analüüsi erineva ülesehitusega failidele. Teiseks võiks kasutada alternatiivseid üksikvastuste teooria mudelite teke, mis võimaldavad 3-parameetrilise (3PL) mudeli loomist ning oletamisparameetri korrektset hindamist. Samuti võiks lisada rakendusele graafilise kasutajaliidese, et kasutajal oleks mugavam ja lihtsam programmiga töötada.

Kokkuvõte

Töö eesmärgiks oli luua rakendus, mis võimaldaks hinnata programmeerimise kursuse nädalatestides sisalduvate küsimuste kvaliteeti põhjalikumalt, kui seda pakub Moodle'i sisseehitatud statistika. Arendusprotsessi käigus sai valmis programmeerimiskeeles Python loodud rakendus, mida saab kasutada Google Colabi keskkonnas. Programm töötleb Moodle'ist eksporditud testide CSV-faile ning arvutab iga küsimuse kohta statistilised näitajad. Lisaks võrdleb rakendus iga õppija esimest ja parimat katset, mis võimaldab hinnata, kas küsimus toetab õppimist või mitte. Väljundina koostatakse PDF-vormingus raport koos graafilise visualiseeringuga.

Testimise käigus leidis rakendus mitmeid küsimusi, mis ei täitnud oma eesmärki. Need olid kas liiga lihtsad, ei eristanud õppijaid või ei toetanud teadmiste arengut. See näitab, et loodud rakendus suudab tuvastada erinevaid probleeme, mis testides esinevad, ja võib anda testide koostajatele tõhusat abi.

Kokkuvõttes valmis töö tulemusena rakendus, mis võimaldab oluliselt täpsemalt hinnata testiküsimuste sobivust. Kuigi rakendus on suunatud ühe konkreetse kursuse testide analüüsile, on see üles ehitatud viisil, mis võimaldab selle lihtsat kohandamist ka teistele kursustele.

Viidatud kirjandus

- [1] Random variable. Math Insight.
https://mathinsight.org/definition/random_variable (05.12.2024)
- [2] Standardhälve. Andmekaitse ja infoturbe portaal.
<https://akit.cyber.ee/term/13606-standardhalve> (05.12.2024)
- [3] Dispersioon. Andmekaitse ja infoturbe portaal.
<https://akit.cyber.ee/term/7080> (05.12.2024)
- [4] Classical test theory. National Council on Measurement in Education.
https://web.archive.org/web/20170722194028/http://www.ncme.org/ncme/NCME/Resource_Center/Glossary/NCME/Resource_Center/Glossary1.aspx?hkey=4bb87415-44dc-4088-9ed9-e8515326a061#anchorC (05.12.2024)
- [5] Hegde, V. & Shushruth, S. (2022) Evaluation of Student's Performance in Programming Using Item Response Theory. Hassan, IEEE.
<https://ieeexplore.ieee.org/document/9915978>
- [6] Binh, H. T. & Duy, B. T. (2016) Student ability estimation based on IRT. Danang, IEEE.
<https://ieeexplore.ieee.org/document/7725667>
- [7] Wood, J. Logistic IRT Models.
https://quantdev.ssri.psu.edu/sites/qdev/files/IRT_tutorial_FA17_2.html (22.12.2024)
- [8] Calkins, K. G. (2005) Applied Statistics - Lesson 13 More Correlation Coefficients. Berrien Springs, Andrews University.
<https://www.andrews.edu/~calkins/math/edrm611/edrm13.htm#POINTB>
- [9] Issayeva, L. The IRT Item Difficulty Parameter. Assessment Systems.
<https://assess.com/irt-item-difficulty-parameter/> (08.12.2024)
- [10] Issayeva, L. The IRT Item Discrimination Parameter. Assessment Systems.
<https://assess.com/irt-item-discrimination-parameter/> (09.12.2024)
- [11] Issayeva, L. The IRT Item Pseudo-guessing Parameter. Assessment Systems.
<https://assess.com/irt-item-pseudo-guessing-parameter/> (28.12.2024)
- [12] Thompson, N. IRT Test Information Function. Assessment Systems.
<https://assess.com/irt-test-information-function/> (04.01.2025)

- [13] Tan, T. K. (2024) Evaluating assessment via item response theory utilizing information function with R. *The Quantitative Methods for Psychology*. Centre for Embodied Learning and Living, National University of Singapore.
<https://www.tqmp.org/RegularArticles/vol20-1/p033/p033.pdf>
- [14] Frost, J. Cronbach's Alpha: Definition, Calculations & Example. *Statistics By Jim*.
<https://statisticsbyjim.com/basics/cronbachs-alpha/> (06.01.2025)
- [15] Online Statistics Calculator. DATAtab.
<https://datatab.net/statistics-calculator/descriptive-statistics> (09.01.2025)
- [16] Moodle. Quiz statistics calculations.
https://docs.moodle.org/dev/Quiz_statistics_calculations (06.01.2025)
- [17] Iteman: Automated Test & Item Analysis with Classical Test Theory. *Assessment Systems*. <https://assess.com/iteman/> (06.01.2025)
- [18] Xcalibre: Item Response Theory Software with User-Friendly UI and Automated Reports. *Assessment Systems*. <https://assess.com/xcalibre/> (06.01.2025)
- [19] Psychometrics 101: Item Total Correlation. *Questionmark*.
<https://www.questionmark.com/resources/blog/psychometrics-101-item-total-correlation/> (24.04.2025)
- [20] Issayeva, L. The IRT Item Difficulty Parameter. *Assessment Systems Corporation*.
<https://assess.com/irt-item-difficulty-parameter> (24.04.2025)
- [21] Hays, R. D. & Morales, L. S. & Reise, S. P. Item Response Theory and Health Outcomes Measurement in the 21st Century. *National Library of Medicine*.
<https://pmc.ncbi.nlm.nih.gov/articles/PMC1815384/> (24.04.2025)

Lisad

I. Programmi kättesaamine ja juhend

Töös loodud analüüsiprogramm on kättesaadav järgmise Google Drive'i kausta kaudu: <https://drive.google.com/drive/folders/17c0NNNqOGuhtxuJYzBkiXzAeCUGhs8DU?usp=sharing>

Kaust on avalik kõigile, kellel on link, ning sisaldab .ipynb formaadis faili, mida saab avada ja käivitada Google Colab keskkonnas.

Kasutusjuhend

- Avamine ja käivitamine:

Ava kaustas olev .ipynb fail Google Colabi kaudu (vajadusel logi sisse Google'i kontoga). Esimese sammuna ühendatakse Google Drive Colabi tööruumiga – kasutaja peab andma õiguse ja valima Drive'i kausta, kuhu andmed on salvestatud.

- Sisendfail:

Programm ootab, et valitud kaustas asuks Moodle'ist eksporditud CSV-fail nimega andmed.csv. Kui faili ei leita või nimi ei vasta nõuetele, annab programm veateate.

- Piirväärtuste määramine:

Kasutaja saab valida vaikimisi piiride komplekti või sisestada ise kõik soovitud piirväärtused. Programm kontrollib, et sisestatud väärtused oleksid sobivas formaadis (arvud, sobivad vahemikud).

- Teisendamine binaarseks:

Programm küsib lävendit (0.00–0.05), millest alates loetakse küsimuse vastus õigeks. Lävendit kasutatakse andmete binaarseteks teisendamiseks analüüsi jaoks.

- Tulemusfail:

Kui kõik sisendid on korrektsed, analüüsib programm testi küsimusi ja arvutab erinevad näitajad (keskmine, standardhälve, punkt-biseriaalne korrelatsioon, raskus, eristusvõime). Lisaks analüüsitakse iga õpilase esimese ja parima katse tulemuste erinevust.

- Väljund:

Lõpptulemusena luuakse PDF-fail nimega analüüs_koos_graafikutega.pdf, kus iga küsimuse kohta on esitatud statistilised näitajad, selgitused ja graafikud. Probleemsed küsimused on PDF-is eraldi märgitud ja nende kohta lisatud automaatsed selgitused.

II. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, Nikita Kislõi,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose **Programmeerimise nädalatestide vastuste analüüsija**, mille juhendaja on Reimo Palm, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada Tartu Ülikooli digitaalarhiivi kuni autoriõiguse kehtivuse lõppemiseni;
2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi kaudu Creative Commons'i litsentsiga CC BY NC ND 4.0, 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.

Nikita Kislõi

13.05.2025