

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Diana Algma

**Hulgateooria ülesannete lisamine
matemaatilise loogika avaldiste teisendamise
keskkonnale**

Bakalaureusetöö (9 EAP)

Juhendaja: dots. Rein Prank

Tartu 2016

Hulgateooria ülesannete lisamine matemaatilise loogika avaldiste teisendamise keskkonnale

Lühikokkuvõte:

Bakalaureusetöona lisati lausearvutuse ja predikaatloogika valemite teisendamise õpiprogrammi võimalus koostada ja lahendada hulgateooria valemite teisendamise ülesandeid. Nende ülesannete lahendamiseks lisati vastav reeglite komplekt ning juhiste saamiseks täiendati automaatlahendajat.

Võtmesõnad:

Hulgateooria, õpiprogramm, teisendamine

CERCS: S281 Arvuti õpiprogrammide kasutamise meetoodika ja pedagoogika; P110 Matemaatiline loogika, hulgateooria, kombinatoorika

Adding set theory exercises to the Algebraic Manipulation Assistant for Propositional Logic and Predicate Calculus

Abstract:

This bachelor thesis added the possibility to create and solve set theoretical exercises to the Algebraic Manipulation Assistant for Propositional Logic and Predicate Calculus. For solving these exercises a corresponding set of rules were added and set theory was integrated into the existing automatic solver.

Keywords:

Set theory, learning program, formulae transformation

CERCS: S281 Computer assisted education; P110 Mathematical logic, set theory, combinatorics

Sisukord

Sissejuhatus	4
1. Sarnane programm SetSails!	5
2. Avaldiste teisendamise ülesanded matemaatilise loogika kursustes.....	10
2.1 Hulgateoreetilised tehted	10
2.2 Ülesannete tüübid	11
2.3 Programmi rakendamine matemaatilise loogika ainetes	12
3. Keskkonna eelmine versioon	13
3.1 Üldine kirjeldus	13
3.2 Õppejõu programm.....	14
3.3 Tudengi programm	16
4. Tehtud muudatused ja täiendused programmis.....	19
4.1 Abstraktne valemi esitus ja parser	19
4.2 Õppejõu programm.....	20
4.3 Reeglite komplekt hulgateooria jaoks	21
4.4 Vahetu teisendamine hulgateooria jaoks	23
4.5 Automaatlahendaja	24
4.6 Väiksemad muudatused.....	24
Kokkuvõte.....	27
Summary	28
Kasutatud materjalid	29
Lisa 1. Kokkupakitud fail programmi, keelefailide ja näiteülesandekoguga.....	30

Sissejuhatus

Aastal 2003 programmeeris Vahur Vaiksaar oma bakalaureusetööna lausearvutuse ja predikaatarvutuse valemite teisendamise keskkonna [1]. Seda programmi kasutati Tartu Ülikooli ainetes Sissejuhatus Matemaatilisse Loogikasse ja Diskreetse Matemaatika elemendid, edaspidi kasutatakse seda ka aines Matemaatiline maailmapilt. Aastal 2013 täiendas Sander Stroom Vaiksaare programmi lisades sellele automaatlahendaja, täiendades teisenduse logi, lisades kasutuskeelega muutmise võimaluse ning parandades mõningaid vigu, näiteks eemaldas üleliigseid teisendusreegleid [2].

2013. aasta versiooni saab täiendada lisades sinna võimaluse teisendada hulgateoreetilisi avaldiseid. Nende teisendamine on sarnane lausearvutuse valemite teisendamisega, mida programm juba võimaldab. Teist programmi, millega saaks hulgateoreetilisi avaldiseid teisendada, Tartu Ülikoolil hetkel pole, mistõttu on hulgateooria valemite teisendamist seni tehtud kirjalikult kas paberil või erinevate tekstiredaktoritega (nt Microsoft Word).

Käesolev bakalaureusetöö on rakendusliku ülesande lahendus. Töö eesmärgina lisas autor lause- ja predikaatarvutuse valemite teisendamise keskkonnale võimaluse teisendada ka hulgateoreetilisi valemeid. Tänu sellele, on seda programmi võimalik kasutada Tartu Ülikooli matemaatika ainetes, kus tegeletakse hulgateooria ning hulgateoreetiliste valemite teisendamisega. Hulgateooria valemeid saab teisendada nii sisestus- kui reeglipõhiselt. Hulgateoreetiliste valemite teisendamise jaoks lisati vastavad ülesandetüübid ning reeglite komplekt. Olemasoleva automaatlahendaja baasil programmeeriti automaatlahendaja, mis annab lahendajale juhiseid hulgateooria ülesannete lahendamiseks.

Esimene peatükk kirjeldab bakalaureusetöös täiendatava programmiga sarnast programmi SetSails!, millega saab tõestada lausearvutuse ja hulgateooria võrduseid. Teises peatükis on kirjeldatud, millistes ainetes matemaatilise loogika avaldise teisendamise programmi kasutatakse ning loetletud ülesannete tüübid, mida selle programmiga on võimalik koostada ja lahendada. Kolmandas peatükis kirjeldatakse aastal 2013 valminud versiooni, eraldi alapeatükkides on õppejõu ja tudengi programmi kirjeldus. Neljandas peatükis tuuakse välja käesoleva bakalaureusetööna programmis tehtud muudatused ja täiendused.

1. Sarnane programm SetSails!

Käesolev peatükk räägib BMBF-i poolt rahastatud ühisprojekti SAiL-M raames loodud programmist SetSails! [3]. SAiL-M projektis osalesid mitmed ülikoolid, näiteks RWTH Aacheni Ülikool ja Ludwigsburgi Ülikool [3]. Programmiga SetSails! saab tõestada lausearvutuse ja hulgateooria võrdusi.

Programmiga saab koostada ja tõestada võrdusi, kus on kuni neli muutujat. Hulgateoorias on võimalik sisestada ühendi, ühisosa, vahe ja täiendi märke, samuti sulge ja tühja hulga ja universaalhulga sümboleid. Ülesande koostamisel saab määrata ka reeglid, mida lahendaja saab kasutada.

The screenshot shows the 'Exercise properties' dialog box. The title bar includes a yellow warning icon and a close button. The 'Title' field contains 'New exercise'. The 'Task' section has a selected radio button for 'Prove that:' and a text input field with 'Proof the rule XY:'. The 'Equation:' section features two text input fields: the left one contains $A \cap (A \cup B)$ and the right one contains A , with an equals sign between them. Two yellow circular icons are positioned on either side of the equation. Below the equation are two identical grids of buttons. The first grid contains buttons for set operations: \cup , \cap , \setminus , and \cdot ; the second grid contains buttons for set symbols: $($, $)$, \emptyset , and U . A third grid contains buttons for variables: A , B , C , and D . The 'Available rules:' section is a list with checkboxes for: 'Absorbing elements', 'Absorption law', 'Associative law', 'Commutative law', 'Complementary elements', and 'De Morgan's laws'. At the bottom are 'OK' and 'Cancel' buttons.

Joonis 1. SetSails! hulgateooria ülesande koostamise aken

Reeglid, mis programmis on antud, on järgmised:

- 1) neelamiseadused tühihulga ja universaalhulgaga: $A \cap \emptyset = \emptyset$, $A \cup U = U$;
- 2) neelamiseadused: $A \cap (A \cup B) = A$, $A \cup (A \cap B) = A$;
- 3) assotsiatiivsuse seadused: $(A \cap B) \cap C = A \cap (B \cap C)$, $(A \cup B) \cup C = A \cup (B \cup C)$;
- 4) kommutatiivsuse seadused: $A \cup B = B \cup A$, $A \cap B = B \cap A$;
- 5) vastastikused täiendid: $A \cup A' = U$, $A \cap A' = \emptyset$;
- 6) De Morgani seadused: $(A \cap B)' = A' \cup B'$, $(A \cup B)' = A' \cap B'$;
- 7) distributiivsuse seadused: $(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$, $(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$;
- 8) topelttäiendi eemaldamine: $A'' = A$;
- 9) ühisosa idempotentsus: $A \cap A = A$;
- 10) ühendi idempotentsus: $A \cup A = A$;
- 11) ühisosa neutraalne element: $A \cap U = A$;
- 12) ühendi neutraalne element: $A \cup \emptyset = A$;
- 13) vahe: $A \setminus B = A \cap B'$;
- 14) universaal- ja tühi hulk: $\emptyset' = U$, $U' = \emptyset$.

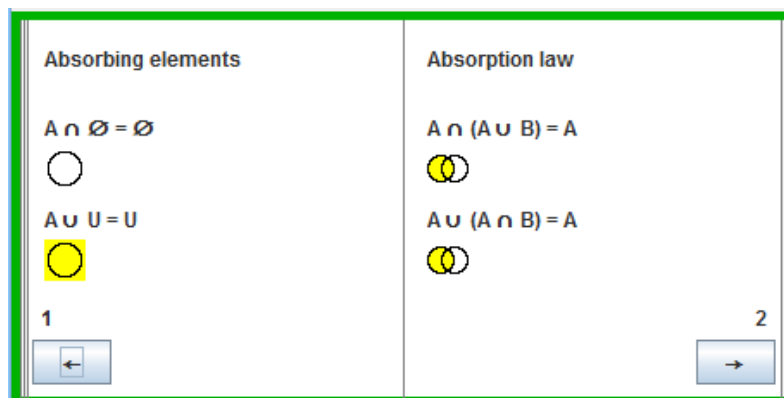
The screenshot shows the SetSails! software interface. On the left, a Venn diagram with two overlapping circles, A and B, is shown within a universal set U. The regions of A and B that do not overlap with each other are shaded yellow. Below the diagram is a green button labeled 'Rules and Definitions' and an 'Open book' button.

On the right, a proof table is visible. The table has two columns: 'Term' and 'Rule'. The proof steps are as follows:

	Term	Rule
1	$(A \setminus B) \cup (B \setminus A)$	Set difference
2	$= (A \cap B^c) \cup (B \setminus A)$	Set difference
3	$= (A \cap B^c) \cup (B \cap A^c)$	De Morgan's laws
4	$= (A \cap B^c) \cup (B \cap A^{ccc})$ $= ((A \cap B^c)^c \cap (B \cap A^c)^c)^c$ $= (A^c \cup B)^c \cap (B \cap A^c)$ $= (A \cap B^c) \cup (B^c \cup A)^c$ = Enter a custom term	
5		
6	$= (A \cup B) \cap (A \cap B)^c$	Set difference
7	$= (A \cup B) \setminus (A \cap B)$	

Joonis 2. SetSails! hulgateooria ülesande lahendamise aken

Joonisel 2 on ülesande lahendamise aken. Ülesande lahendajal on võimalik teisendada nii võrduse vasakut kui paremat poolt. Teisendamiseks peab lahendaja valima reegli, mida rakendada. Kasutada saab ka enda sisestatud reeglit, kuid programm hoiatab sel puhul kasutajat, et programm ei saa kontrollida reegli õigsust. Seejärel tuleb lahendajal ette antud valikust valida tulemus või sisestada ise valem. Joonisel on lahendaja valinud sammu tegemiseks De Morgani seaduse ja peab nüüd valima avatud valikust valemi või viimase valiku „Enter a custom term“ valimisel sisestama ise valemi. Lahendamise ajal on võimalik vaadata reeglite raamatut, kus on toodud reeglid ja nendele vastavad Venni diagrammid (vt joonis 3).



Joonis 3. Avatud reeglite raamat ülesande lahendamise aknas

Hulgateooria ülesande puhul näitab programm lahendusakna vasakul pool Venni diagrammi, mis vastab ülesande valemitele. Samuti saab kasutaja, vajutades teisenduses saadud valemi taga olevale Venni diagrammi ikoonile, vaadata Venni diagrammi, mis vastab teisendamisel saadud valemile. Lausearvutuse ülesannete puhul on Venni diagrammide asemel tõeväärtustabelid.

Juhul, kui võrduse pooled on teisendatud samale kujule ja igas teisendussammus on alg- ja lõppvalem samaväärsed, lubab programm teisendussammude jada ühendada ja tõestus on lõpule viidud.

Programm SetSails! ei sobi kasutamiseks paljude ülesannete puhul, mida Tartu Ülikooli matemaatilise loogika ainetes lahendatakse, mitmel põhjusel. Esiteks, ei ole programmiga SetSails! võimalik koostada ülesandeid, kus tuleb viia valemeid normaalkujule või avaldada

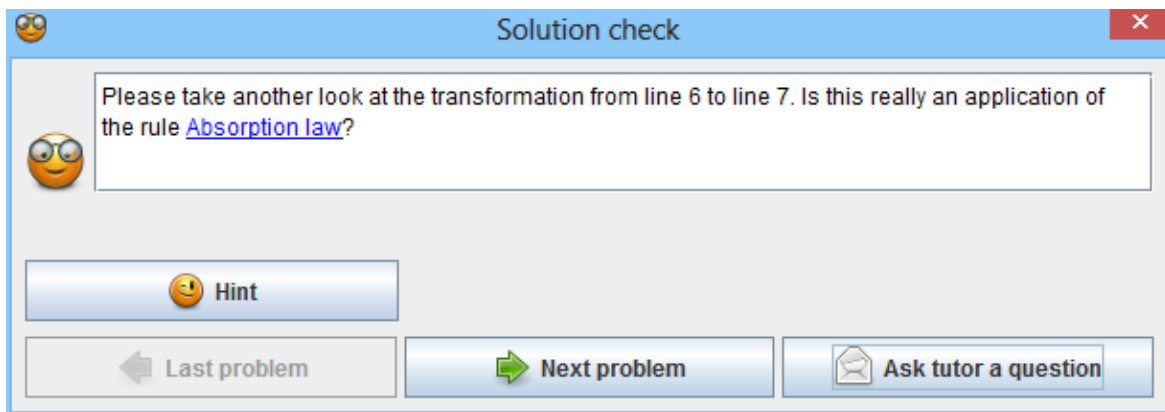
valemeid kindlate tehete kaudu ilma, et oleks ette antud lõppvastus. SetSails! seab ka piirangu muutujate arvule. Lahendajal on alati võimalus vaadata reeglite raamatust reegleid, mistõttu ei saa programmiga kontrollida, kas lahendaja reegleid tunneb või mitte.

Reeglite hulk, mida programm teab, pole eriti suur ja ei sobi hästi praktiliseks teisendamiseks, kuna teisendamine vajab väga palju samme nii väikese reeglite komplektiga teisendamiseks. Reeglite komplekt, mis loodi matemaatilise loogika teisendamise keskkonnale hulgateoreetiliste avaldiste teisendamiseks (vt alapeatükk 4.3), ei sisalda küll universaalhulga ja tühihulgaga reegleid, kuid lubab eemaldada ühisosadest liikmeid, mis on universaalhulga samaväärsed ja ühenditest liikmeid, mis on tühihulgaga samaväärsed. Samuti sisaldab loodud reeglite komplekt rohkem reegleid, mis kiirendab teisendusprotsessi. Komplektis on ka reeglid, mis sisaldavad sümmeetrilist vahet, ja reeglid, millega saab sulge vabalt lisada ja eemaldada, kui see ei riku tehete järjekorda.

Programmi SetSails! ja matemaatilise loogika teisendamise programmi reeglite rakendamine erineb veel selle poolest, et programmis SetSails! ei vali kasutaja osavalemit, millele reeglit rakendada. Selle asemel peab lahendaja teisendussammu lõppvalemite valima valemi, mis just soovitud osavalemi teisendust sisaldab ja seda seejuures korrektselt teeb. Seetõttu võib pikemate valemite peal keerukamate reeglite rakendamine programmiga SetSails! olla üsna raske.

Lahendaja tehtud vead võivad programmis SetSails! jääda märkamata, kui teisendussamme Venni diagrammi abil ei kontrollita. Isegi, kui teisendussammu alg- ja lõppvalem on samaväärsed, võib teisendussamm olla vigane, kui kasutatud reegliks määratakse vale reegel. Sellise vea avastamiseks tuleb kontrollida tõestust nupuga „Check proof ...“. Matemaatilise loogika teisendamise programm annab lahendajale vea korral koheselt tagasisidet ja ei luba lahendajal teha vigaseid teisendusi.

Programmis SetSails! pole lahendajal alati võimalik küsida soovitusi järgmiseks teisendussammuks. Nuppu „Check proof ...“ vajutades avaneb lahenduse kontrolli aken, mis vigaste teisendussammude korral ütleb, millistes sammudes on viga (vt joonis 4). Vajutades nuppu „Hint“, ütleb programm, et kasutatud reegli abil selle teisenduse tegemine on võimatu ja soovib parandada kas reeglit või valemit. Kui ühes viimastest teisendustest on valemisse jäänud topelttäiend, soovib lahenduse kontroll kasutada topelttäiendi eemaldamise reeglit. Kui teisendussammud on korrektsed, siis programm enamasti soovitusi ei anna.



Joonis 4. Lahenduse kontrolli aken

2. Avaldiste teisendamise ülesanded matemaatilise loogika kursustes

Varem kasutati antud programmi Tartu Ülikooli ainetes Diskreetse matemaatika elemendid ja Sissejuhatus matemaatilisse loogikasse. Programmi abil harjutasid tudengid nii lause- kui ka predikaatarvutuse valemite teisendamist ning programmiga viidi läbi ka kontrolltöid. Nendes ainetes teisendati lisaks lause- ja predikaatarvutuse valemitele ka hulgateoreetilisi valemeid. Valemeid pidi teisendama kirjalikult, mitte programmi abiga, kuna selleks vajalikku programmi Tartu Ülikoolil polnud. Õppeaastast 2015/2016 eelnimetatud aineid enam õppekavades pole, nende asemel on ained Matemaatiline maailmapilt ning Graafid ja matemaatiline loogika, kus edaspidi seda programmi kasutatakse.

Alapeatükis 2.1 on defineeritud hulgateoreetilised tehted. Alapeatükis 2.2 on loetletud ülesannete tüübid, mida selle programmiga koostada ja lahendada saab. Alapeatükis 2.3 on toodud näiteid, kuidas neid ülesandetüüpe saab kasutada matemaatilise loogika kursustes.

2.1 Hulgateoreetilised tehted

Hulgateoreetilised valemid, mida programmiga bakalaureusetöö tulemusena teisendada saab, koosnevad järgmistest tehetest:

- 1) täiend,
- 2) ühisosa,
- 3) ühend,
- 4) vahe,
- 5) sümmeetriline vahe.

Järgnevalt tuuakse ära nende tehete definitsioonid [4].

Def. Hulga A täiendiks nimetatakse hulka A' , mille moodustavad kõik elemendid, mis ei kuulu hulka A .

Def. Hulkade A ja B ühisosaks nimetatakse hulka $A \cap B$, mille moodustavad kõik elemendid, mis kuuluvad nii hulka A kui hulka B .

Def. Hulkade A ja B ühendiks nimetatakse hulka $A \cup B$, mille moodustavad kõik elemendid, mis kuuluvad kas hulka A või hulka B .

Def. Hulkade A ja B vaheks nimetatakse hulka $A \setminus B$, mille moodustavad kõik elemendid, mis kuuluvad hulka A , aga ei kuulu hulka B .

Def. Hulkade A ja B sümmeetriliseks vaheks nimetatakse hulka $A\Delta B$, mille moodustavad kõik elemendid, mis kuuluvad kas hulka A või hulka B , aga mitte hulka $A\cap B$.

2.2 Ülesannete tüübid

Programmi eelmises versioonis on kümme lausearvutuse ülesandetüüpi ja kaheksa predikaatarvutuse ülesandetüüpi. Bakalaureusetöona on lisatud viis hulgateooria ülesandetüüpi.

Lausearvutuse ülesandetüübid, mida programmiga on võimalik koostada ja lahendada, on järgmised:

- 1) antud valemi avaldamine konjunktsiooni ja eituse kaudu;
- 2) antud valemi avaldamine disjunktsiooni ja eituse kaudu;
- 3) antud valemi avaldamine implikatsiooni ja eituse kaudu;
- 4) antud valemi viimine disjunktiivsele normaalkujule;
- 5) antud valemi viimine konjunktiivsele normaalkujule;
- 6) antud valemi viimine täielikule disjunktiivsele normaalkujule;
- 7) antud valemi viimine täielikule konjunktiivsele normaalkujule;
- 8) antud valemi viimine eitusega prefikskujule;
- 9) eituste viimine vahetult lausemuutujate ette;
- 10) lausearvutuse vaba teisendus (vt 2.3).

Predikaatarvutuse ülesandetüübid, mida programmiga on võimalik koostada ja lahendada, on järgmised:

- 1) antud valemi avaldamine konjunktsiooni, eituse ja üldisuse kvantori kaudu;
- 2) antud valemi avaldamine konjunktsiooni, eituse ja olemasolu kvantori kaudu;
- 3) antud valemi avaldamine disjunktsiooni, eituse ja üldisuse kvantori kaudu;
- 4) antud valemi avaldamine disjunktsiooni, eituse ja olemasolu kvantori kaudu;
- 5) antud valemi avaldamine implikatsiooni, eituse ja üldisuse kvantori kaudu;
- 6) antud valemi avaldamine implikatsiooni, eituse ja olemasolu kvantori kaudu;
- 7) antud valemi viimine predikaatloogika valemi prefikskujule;
- 8) predikaatloogika vaba teisendus.

Bakalaureusetöona on programmi lisatud järgmised hulgateooria ülesandetüübid:

- 1) antud valemi avaldamine ühisosa ja täiendi kaudu;
- 2) antud valemi avaldamine ühendi ja täiendi kaudu;
- 3) antud valemi viimine disjunktiivsele normaalkujule;

- 4) antud valemi viimine täielikule disjunktiivsele normaalkujule;
- 5) hulgateooria vaba teisendus.

2.3 Programmi rakendamine matemaatilise loogika ainetes

Kõik ülesandetüübid, mis pole *lausearvutuse*, *predikaatloogika* või *hulgateooria vaba teisendus*, on mõeldud kindlat tüüpi ülesannete lahendamiseks. Matemaatilise maailmapildi praktikumiülesannete kogus [5] on mitmeid ülesandeid, mida saab programmi abil koostada ja lahendada, sest programmis leidub sama ülesandetüüp. Näiteks vastab selle ülesannete kogu ülesanne 52 („Teisendage järgmised valemid nii, et nad sisaldaksid ainult eitust ja konjunktsiooni“ [5]) ülesandetübile *Antud valemi avaldamine konjunktsiooni ja eituse kaudu*.

Kasutades *vaba teisenduse* ülesandetüüpe, saab koostada mitmeid teisi ülesandeid, millele täpset ülesandetüüpi programmis pole. Näiteks saab koostada ülesandeid, kus on vaja valemid lihtsustada. Programm kontrollib, kas kõik teisendused on korrektsed, ehk kas igal teisendussammul on algvalem ja lõppvalem samaväärsed. Kontrolli, kas lahendaja valemist piisavalt lihtsustas, peab tegema õppejõud. Lihtsustamiseks pole eraldi ülesandetüüpi, sest mõiste lihtsustamine pole üheselt mõistetav [6] ja seega on kriteeriumite seadmine keeruline.

Vaba ülesandetübina saab samuti koostada ülesandeid, kus peab ühe valemi teisendama teiseks. Ka selle kontrolli, kas valem viidi nõutavale kujule, peab tegema õppejõud.

Matemaatilise maailmapildi ülesannete kogu 127. ülesannet [5] (vt joonis 5) saab samuti koostada ja lahendada kasutades ülesandetüüpi *Hulgateooria vaba teisendus*. Ülesande tekstis tuleks öelda, milliste tehete kaudu peab valemist avaldama. Kontrolli, kas valem on viidud nõutud kujule, peab jällegi tegema õppejõud, kontrolli, kas saadud valem on algse valemiga samaväärne, teeb aga lahendamise ajal programm.

127. Avaldage

- | | |
|--|--|
| a) $A \cup B$ tehete \cap ja Δ abil; | d) $A \setminus B$ tehete \cup ja Δ abil; |
| b) $A \setminus B$ tehete \cap ja Δ abil; | e) $A \cup B$ tehete \setminus ja Δ abil; |
| c) $A \cap B$ tehete \cup ja Δ abil; | f) $A \cap B$ tehete \setminus abil. |

Joonis 5. Matemaatilise maailmapildi ülesannete kogu 127. ülesanne

3. Keskkonna eelmine versioon

Lause- ja predikaatarvutuse valemite teisendamise programmi esimese versiooni programmeeris Vahur Vaiksaar oma bakalaureusetöö raames aastal 2003 ja aastal 2013 täiendas seda programmi Sander Stroom enda bakalaureusetööna. Selles peatükis kirjeldatakse aastal 2013 valminud versiooni, mida käesoleva bakalaureusetööna täiendati.

3.1 Üldine kirjeldus

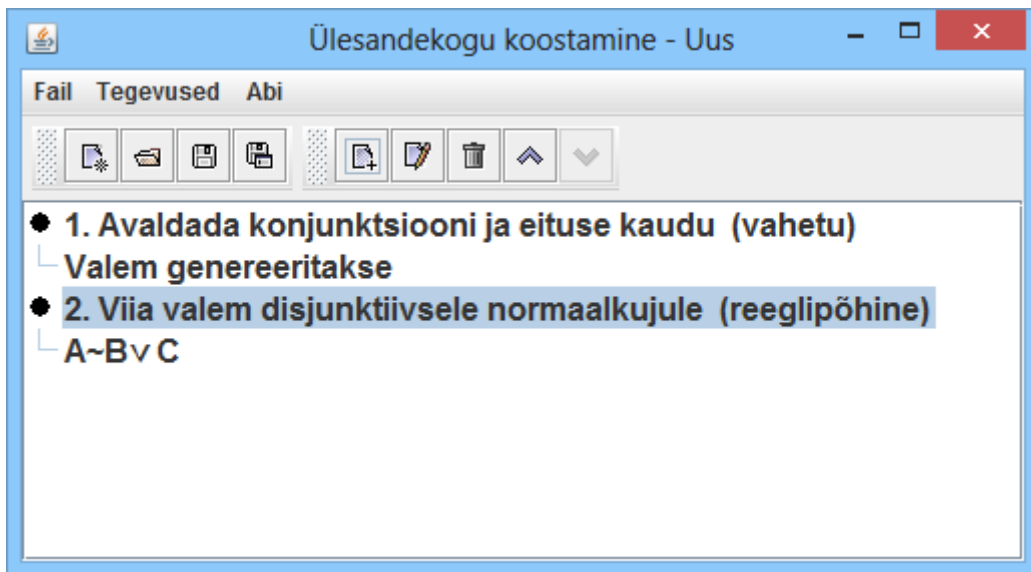
Programmiga saab teisendada nii lausearvutuse kui predikaatarvutuse valemeid. Lausearvutuse valemid sisaldavad lausearvutuse tehteid, milleks on eitus, konjunktsioon, disjunktsioon, implikatsioon ja ekvivalents. Predikaatarvutuse valemid põhinevad lausearvutusel, kuid sisaldavad ka predikaate ja kvantoreid. Teisendamine on võimalik nii sisestus- kui reeglipõhiselt. Sisestuspõhisel ehk vahetul teisendamisel tuleb õpilasel valida osavalem ning sisestuskasti kirjutada valemi osa, millega valitud valem asendatakse, sellise teisendusviisi puhul programm õpilasele teisendusreegleid ei kuva. Programm kontrollib, et soovitud asendus oleks samaväärne valitud valemi osaga ja asenduse tegemisel tehete järjekord valemis ei muutuks. Reeglipõhisel teisendamisel tuleb õpilasel valida osavalem ning seejärel reegel, mida osavalemile rakendada, ja programm teeb vajaliku teisenduse ära, kui valitud osavalem on reegli rakendamiseks vajalikul kujul. Lausearvutuse valemeid saab teisendada mõlemal viisil, kuid predikaatarvutuse puhul on see võimalik vaid reeglipõhiselt, sest programmiga pole võimalik kontrollida predikaatarvutuse valemite samaväärsust.

Programm jaguneb kaheks osaks, õppejõu ja tudengi programmiks. Õppejõu programmiga saab koostada ülesandekogusid, mis võivad sisaldada erinevaid ülesandeid. Tudengi programmiga saab luua lahendusfaili, mis sisaldab ühe ülesandekogu ülesandeid, ning neid lahendada. Lahendused salvestatakse samasse lahendusfaili ning tudengi programmiga on võimalik vaadata lahendusfailis salvestatud tulemusi, mis sisaldavad statistikat ja lahenduskäike (teisenduse logi). Teisenduse logi sisaldab iga lahenduskorra samme, teisendusi, juhiste küsimisi ja tehtud vigu.

Mõlemas programmis on võimalik vahetada keelt, kui on olemas vastavad keelefailid, ning lugeda programmi kasutusjuhendit. Programmiga koos on olemas nii eesti- kui ingliskeelne keelefail. Kasutusjuhend annab ülevaate sellest, mida ja kuidas on programmiga võimalik teha, mida erinevad nupud teevad ning milliste klahvidega saab sisestada erinevaid tehtmärke.

3.2 Õppejõu programm

Õppejõu programmi avades luuakse automaatselt uus tühi ülesandekogu. Samuti on kasutajal võimalik avada juba olemasolev ülesandekogu fail või lisada uude ülesandekogusse ülesandeid. Pärast ülesande või ülesannete lisamist on võimalik ülesandeid kustutada ja nende järjekorda muuta, uus ülesandekogu salvestada või avada uus tühi kogu. Joonisel 6 on õppejõu programm uue ülesandekoguga, kuhu on lisatud kaks ülesannet. Teine ülesanne on valitud ning seda on võimalik liigutada ülespoole. Ülesandekogud salvestatakse *.ylk* failina.



Joonis 6. Õppejõu programm

Uue ülesande aknas (vt joonis 7) saab sätestada ülesande seaded. Ülemises osas (ülesande tüüp) saab valida ülesande tüüpi lause- ja predikaatülesannete hulgast. Ülesande kirjelduse osas saab soovi korral muuta ülesande teksti. Igal ülesandetüübil on ka vaikimisi ülesande tekst. Joonisel valitud ülesande vaikimisi tekst on „Avaldada konjunktsiooni ja eituse kaudu“. Valemis osas saab valida, kas valem sisestatakse ülesande koostaja poolt või genereerib selle programm. Teisendamise osas valitakse, kas teisendamine on vahetu või reeglipõhine ja lahenduste pakkumise osas, kas programm annab lahendamisel juhiseid või mitte. Maksimaalset lubatud vigade arvu saab määrata iga veatüübi kohta eraldi. Kui tudeng teeb lahendamisel mingit viga rohkem, kui lubatud, siis programm ülesannet lahendamiseks ei loe.

Uus ülesanne ✕

Ülesande tüüp

Lausearvutus	Predikaatarvutus
<input checked="" type="radio"/> {&,¬}	<input type="radio"/> TDNK
<input type="radio"/> {∨,¬}	<input type="radio"/> TKNK
<input type="radio"/> {⊃,¬}	<input type="radio"/> eitPrefiks
<input type="radio"/> DNK	<input type="radio"/> eitSees
<input type="radio"/> KNK	<input type="radio"/> Vaba teisendus
	<input type="radio"/> {&,¬,∀}
	<input type="radio"/> {⊃,¬,∀}
	<input type="radio"/> {&,¬,∃}
	<input type="radio"/> {∨,¬,∀}
	<input type="radio"/> {∨,¬,∃}
	<input type="radio"/> kvanPrefiks
	<input type="radio"/> Vaba teisendus

Ülesande kirjeldus

Valem

Sisestatav Genereeritav

Teisendamine

Vahetu Reeglipõhine

Lahenduse pakkumine

Aitab Ei aita

Maksimaalne lubatud vigade arv

Sünt. vigu	<input type="text" value="0"/>	Teis. vigu	<input type="text" value="0"/>
Jrk. vigu	<input type="text" value="0"/>	Vast. vigu	<input type="text" value="0"/>

Joonis 7. Uue ülesande loomise dialog

Uus valem ✕

¬ - F1 & - F2 ∨ - F3 ⊃ - F4 ~ - F5 ∀ - F6 ∃ - F7

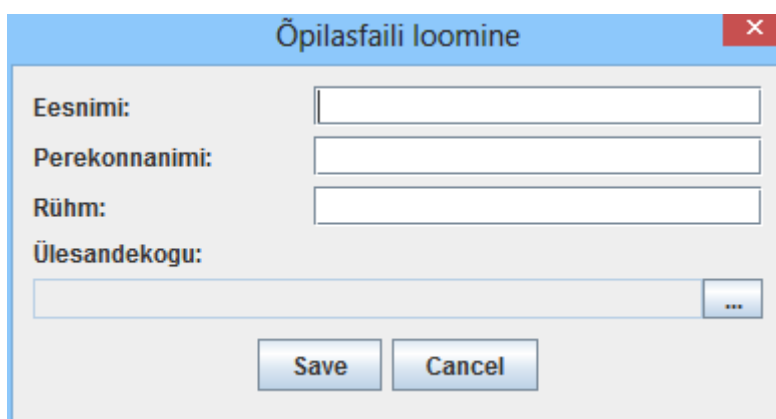
Joonis 8. Predikaatarvutuse valemi sisestamise aken

Kui lisatav ülesanne on ülesande koostaja poolt sisestatava valemiga, ilmub nupule „Ok“ vajutades aken, kus saab valemi sisestada (vt joonis 8). Lausearvutuse ülesande korral saab

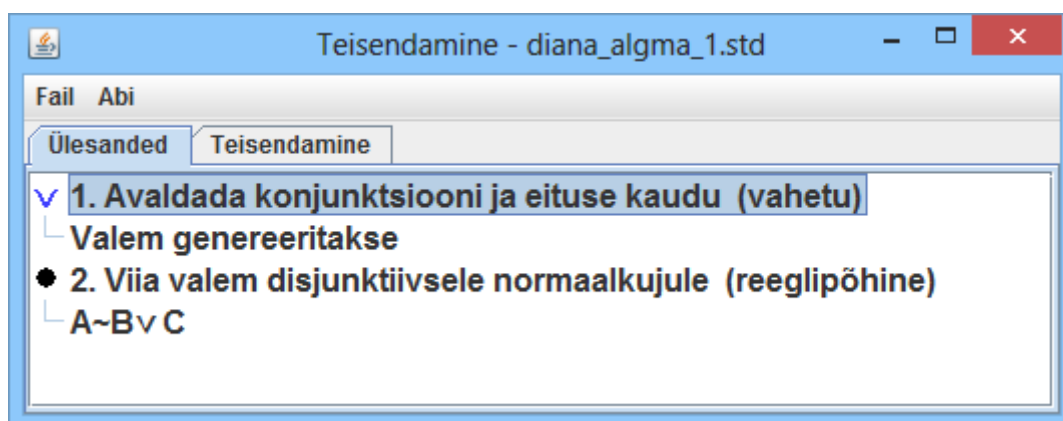
valemisse sisestada eituse, konjunktsiooni, disjunktsiooni, implikatsiooni ja ekvivalentsi märke, suurtähelisi muutujaid A-st kuni O-ni ning ümarsulge. Predikaatarvutuse valemi sisestamisel saab lisaks lausearvutuses lubatud sümbolitele kasutada ka üldisuse ja olemasolu kvantoreid, kantsulge, komasid ning nii väike- kui suurtähti vahemikus A-st Z-ni. Sisestatud valem peab olema süntaktiliselt korrektne, vastasel korral annab programm veateate ning laseb ülesande koostajal valemit parandada.

3.3 Tudengi programm

Tudengi programmis saab avada olemasoleva või luua uue õpilasfaili. Uue õpilasfaili loomisel tuleb sisestada õpilase ees- ja perekonnanimi ning rühm ja valida ülesandekogu (vt joonis 9). Seejärel tuleb õpilasfail salvestada laiendiga *.std*.




Joonis 9. Uue õpilasfaili loomise aken



Joonis 10. Tudengi programmi ülesande põhiaken

Tudengi programm koosneb kahest põhiaknast, „Ülesanded“ ja „Teisendamine“ (vt joonis 10). Kui tudengiprogrammis on avatud õpilasfail, on ülesannete põhiaknas võimalik näha

ülesannete nimekirja ja iga ülesande juures seda, kas ülesanne on lahendatud või mitte, lahendatud ülesannet märgib linnuke, joonisel on esimene ülesanne lahendatud. Ülesande valimisel avaneb teisendamise põhiaken, kus saab valemit teisendada. Teisendamisel on näha tehtud teisendussammud ning viimase teisendussammuga saadud valemist on võimalik valida osavalem, mida teisendada (vt joonis 11). Valitud osavalemid on näidatud sinisega. Reeglipõhisel teisendusel on kasutamiseks reeglid, vahetu teisendamise korral valemi sisestuse väli. Reeglite komplekt ja valemis sisestuseks vajalikud nupud sõltuvad sellest, kas ülesanne on lause- või predikaatarvutuse ülesanne. Lausearvutuse normaalkujudele ning kindlate tehetega väljendamise ülesannete puhul on võimalik küsida juhiseid selle kohta, mida järgmiseks teha. Juhiste küsimiseks tuleb vajutada nuppu . Kui teisendus on lõpule viidud, saab lahenduse esitada, kui see on nõutud kujul.



Joonis 11. Reeglipõhine lausearvutuse valemi teisendamine

Tudengi programmis on võimalik vaadata tulemusi avatud õpilasfaili kohta. Tulemuste aknas (vt joonis 12) on tabel, kus igal real on üks ülesanne ning erinevad andmed lahenduskatsete kohta, sealhulgas edukate katsete arv ning erinevate vigade hulk. Topeltklõps ühel

real avab akna, kus on võimalik näha valitud ülesande igat lahenduskäiku. Lahenduskäikudega koos on teave näiteks selle kohta, milliseid osavalemeid valiti, kas küsiti juhiseid, milliseid reegleid kasutati ja milliseid vigu tehti.

Nr.	Tulemus	Katseid	Edukalt	Valesti	Katkestati	Sünt. vigu	Jrk. vigu	Teis. vigu	Sulud lisa...
1.	Ok	2	1	0	1	0	0	0	0
2.	Not ok	0	0	0	0	0	0	0	0
3.	Not ok	0	0	0	0	0	0	0	0
4.	Ok	1	1	0	0	0	0	0	0
5.	Not ok	0	0	0	0	0	0	0	0
6.	Not ok	0	0	0	0	0	0	0	0
7.	Not ok	0	0	0	0	0	0	0	0
8.	Not ok	0	0	0	0	0	0	0	0

Joonis 12. Tulemuste aken

Teisendusprogrammis eraldi salvestamismuppu ei ole, kuna õpilasfaili uuendatakse jooksvalt teisendusprogrammi kasutamise ajal.

4. Tehtud muudatused ja täiendused programmis

Tähtsaima täiendusena lisati programmi hulgateooria ülesannete lahendamise võimalus. Hulgateooria ülesannetes on võimalik teisendada hulgateooria valemeid, kus tehemärki- deks on täiend, ühisosa, ühend, vahe ja sümmeetriline vahe. Programmiga ei saa teisendada valemeid, kus on nii hulgateooria kui lausearvutuse tehteid. Hulgateoreetiliste ülesannete lisamiseks pidi programmi mitmeid osi (nt valemi generaator ja parser) täiendama ja samuti tehti väikseid parandusi, mis pole otseselt seotud hulgateoreetiliste ülesannete lisamisega. Mõned muudatused puudutavad nii õppejõu- kui tudengi programmi, aga enamus muudatu- sed käivad vaid tudengi programmi kohta. Ainult õppejõu programmi kohta käivaid muu- datusi on vähe ning need on kirjeldatud alapeatükis 4.2.

4.1 Abstraktne valemi esitus ja parser

Esimesena tuli lisada abstraktsed valemi klassid iga hulgateoreetilise tehte kohta, mida see programm kasutab. Nendeks on täiend, ühisosa, ühend, vahe ja sümmeetriline vahe. Abstraktsed valemi klassid sisaldavad tehte prioriteeti, tõeväärtust ning valemi esitust sõne kujul. Kuigi hulgateoreetilisel tehtel pole tõeväärtust, nõuab seda programmi valemi klass. Hulgateoreetiliste tehete tõeväärtus on realiseeritud lausearvutuse eeskujul, see tähendab, et kõi- gile hulga tähistele on võimalik anda väärtustus, kas tõene või väär, ja tehte tõeväärtus sõltub tähiste väärtustustest. Valemi A' tõeväärtus on tõene parajasti siis, kui A tõeväärtus on väär. Vahe tõeväärtus on tõene parajasti siis, kui vasaku argumendi tõeväärtus on tõene ja parema argumendi tõeväärtus on väär. Samuti pidi valemi klassi lisama lausearvutuse tehete mär- kide juurde hulgateoreetiliste tehete märgid. Ühisosa, ühendi ja sümmeetrilise vahe märki- dena kasutatakse Unicode sümboleid vastavalt `\u2229`, `\u222A` ja `\u2206`. Vahe märgina kasutatakse langkriipsu (`\`) ja täiendi märgina ülakoma (`'`).

Hulgateoreetiliste valemitega töötamiseks oli vaja täiendada olemasolevat parserit, mis töö- tles valemit otsides kõigepealt madalaima prioriteediga, siis järjest kõrgema prioriteediga tehteid. Hulgateooria tehted lisati parserisse võimalikult analoogiliselt lausearvutuse tehe- tega, kuid täielikult polnud see võimalik, sest tehete prioriteedid lausearvutuses ja hulga- teoorias on erinevad ning erinevalt eituse märgist on täiendi märk alamvalemi taga, mitte ees.

Lausearvutuse tehete hulgas pole sama prioriteediga tehteid, mis teeb sellise parseri töö lihtsaks, igal sammul kontrollitakse ühte tehet. Programm järgib hulgateooria puhul prioriteedijärjekorda, mida kasutatakse Tartu Ülikooli aines Sissejuhatus Matemaatilisse loogikasse ja mille järgi on kõrgeima prioriteediga tehe täiend, sellest järgmine ühisosa ning madalaima prioriteediga on ühend, vahe ning sümmeetriline vahe [7]. Kuna viimased kolm on võrdse prioriteediga, tuleb parseris ühel sammul kontrollida mitut erinevat tehet, seda tehakse samas sammus, kus otsitakse disjunktsiooni.

Täiendit ei saa otsida analoogiliselt eitusega, kuna täiendi märk asub muutujast või alamvalemist paremal, eituse märk aga vasakul. Seetõttu kontrollitakse parseri töö viimastel samudel, kas leitud muutujast või sulgudes olevast alamvalemist võetakse täiend.

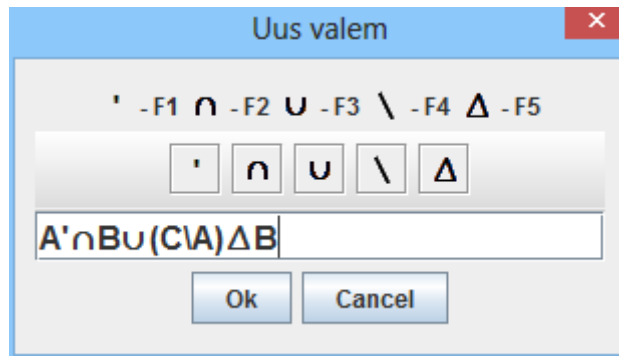
Valemite sisestamisel kontrollib sisestatud teksti tekstifilter, mis lubab sisestada vaid muutujaid, lubatud tehtmärke, erinevaid sulge ning predikaatarvutuse jaoks komasid. Selleks, et valemi sisestamise väljadesse oleks võimalik sisestada hulgateooria tehteid, pidi täiendama tekstifiltrit, et nende tehete sümbolid oleksid hulgateoreetilise valemi sisestamisel lubatud.

4.2 Õppejõu programm

Õppejõu programmi lisati lisaks lausearvutusele ja predikaatarvutusele komplekt hulgateooria ülesandetüüpe, mis on loetletud peatükis 2.1. Kõiki hulgateooria ülesandeid saab lahendada nii vahetult kui reeglipõhiselt.

Hulgateooria avaldiste genereerimiseks loodi lausearvutuse valemite generaatoriga analoogiline generaator. Valides valemi genereerimise, saab kasutaja valida iga tehtmärgi esinemiste arvu ning mitu erinevat muutujat valemis esineb. Kui valitakse valemi sisestamine, peab sisestatud valem olema süntaktiliselt korrektne ning valem ei tohi olla tühi. Vastasel juhul annab programm veateate ning laseb kasutajal valemit parandada. Kui valem on korrektne, salvestatakse ülesanne.

Hulgateooria valemite sisestamiseks lisati uued nupud, mida kasutatakse ka vahetu teisen-damise puhul. Uued nupud on näha joonisel 13 kujutatud valemi sisestamise aknas.



Joonis 13. Valemi sisestamise aken

4.3 Reeglite komplekt hulgateooria jaoks

Hulgateooria ülesannete reeglipõhise lahendamise jaoks lisati uus komplekt hulgateooria samaväärsusi. Komplekti koostamisel võeti aluseks olemasolev lausearvutuse reeglite komplekt ning kasutati Timmu Ründali [8] bakalaureusetööd, kus oli välja toodud lisaks lausearvutuse teisendusreeglitele ka hulgateooria teisendusreegleid.

Joonisel 14 on näha programmi eelmises versioonis olev lausearvutuse reeglite komplekt ning joonisel 15 lisatud hulgateooria reeglite komplekt nii, nagu nad on näha programmis reeglipõhise teisenduse aknas. Edasi on kirjeldatud, kuidas on saadud hulgateooria reeglite komplekt.

$(X) \Rightarrow \Leftarrow X$	$X \supset Y \Rightarrow \Leftarrow \neg(X \& \neg Y)$	$X \& (Y \vee Z) \Rightarrow \Leftarrow X \& Y \vee X \& Z$
$\neg \neg X \Rightarrow \Leftarrow X$	$\neg(X \supset Y) \Rightarrow \Leftarrow X \& \neg Y$	$X \vee Y \& Z \Rightarrow \Leftarrow (X \vee Y) \& (X \vee Z)$
$X \& Y \Rightarrow \Leftarrow \neg(\neg X \vee \neg Y)$	$X \supset Y \Rightarrow \Leftarrow \neg X \vee Y$	$\neg X \& X \vee Y \Rightarrow Y$
$\neg(X \& Y) \Rightarrow \Leftarrow \neg X \vee \neg Y$	$X \sim Y \Rightarrow \Leftarrow X \& Y \vee \neg X \& \neg Y$	$(\neg X \vee X) \& Y \Rightarrow Y$
$X \vee Y \Rightarrow \Leftarrow \neg(\neg X \& \neg Y)$	$\neg(X \sim Y) \Rightarrow \Leftarrow X \& \neg Y \vee \neg X \& Y$	$X \vee X \& Y \Rightarrow X$
$\neg(X \vee Y) \Rightarrow \Leftarrow \neg X \& \neg Y$	$X \sim Y \Rightarrow \Leftarrow (X \supset Y) \& (Y \supset X)$	$X \& (X \vee Y) \Rightarrow X$
$X \& Y \Rightarrow \Leftarrow \neg(X \supset \neg Y)$	$\neg(X \sim Y) \Rightarrow \Leftarrow \neg(X \supset Y) \vee \neg(Y \supset X)$	$X \oplus Y \Rightarrow Y \oplus X$
$\neg(X \& Y) \Rightarrow \Leftarrow X \supset \neg Y$	$X \Rightarrow \Leftarrow X \& Y \vee X \& \neg Y$	$X \oplus X \Rightarrow X$
$X \vee Y \Rightarrow \Leftarrow \neg X \supset Y$	$X \Rightarrow \Leftarrow (X \vee Y) \& (X \vee \neg Y)$	$X \oplus (Y \oplus Z) \Rightarrow (X \oplus Y) \oplus Z$

Joonis 14. Lausearvutuse reeglite komplekt

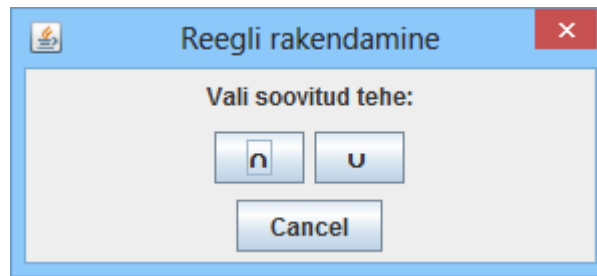
(X)	\rightarrow	\leftarrow	X	$X\Delta Y$	\rightarrow	\leftarrow	$X\cup Y X\cap Y$	$X'\cap X\cup Y$	\rightarrow	Y	
X''	\rightarrow	\leftarrow	X	$X\Delta Y$	\rightarrow	\leftarrow	$X\cap Y\cup (YX)$	$(X'\cup X)\cap Y$	\rightarrow	Y	
$X\cap Y$	\rightarrow	\leftarrow	$(X'\cup Y)'$	X	\rightarrow	\leftarrow	$X\cap Y\cup X\cap Y'$	$X\cup X\cap Y$	\rightarrow	X	
$(X\cap Y)'$	\rightarrow	\leftarrow	$X'\cup Y'$	X	\rightarrow	\leftarrow	$(X\cup Y)\cap (X\cup Y')$	$X\cap (X\cup Y)$	\rightarrow	X	
$X\cup Y$	\rightarrow	\leftarrow	$(X'\cap Y)'$	$X\cap (Y\cup Z)$	\rightarrow	\leftarrow	$X\cap Y\cup X\cap Z$	$X\oplus Y$	\rightarrow	$Y\oplus X$	
$(X\cup Y)'$	\rightarrow	\leftarrow	$X'\cap Y'$	$X\cup Y\cap Z$	\rightarrow	\leftarrow	$(X\cup Y)\cap (X\cup Z)$	$X\oplus X$	\rightarrow	\leftarrow	X
$X\cap Y'$	\rightarrow	\leftarrow	$X\cap Y'$	$X\cap (Y\Delta Z)$	\rightarrow	\leftarrow	$X\cap Y\Delta X\cap Z$	$X\oplus (Y\oplus Z)$	\rightarrow	\leftarrow	$(X\oplus Y)\oplus Z$
$X\Delta Y$	\rightarrow	\leftarrow	$X\cap Y'\cup X'\cap Y$								

Joonis 15. Hulgateooria reeglite komplekt

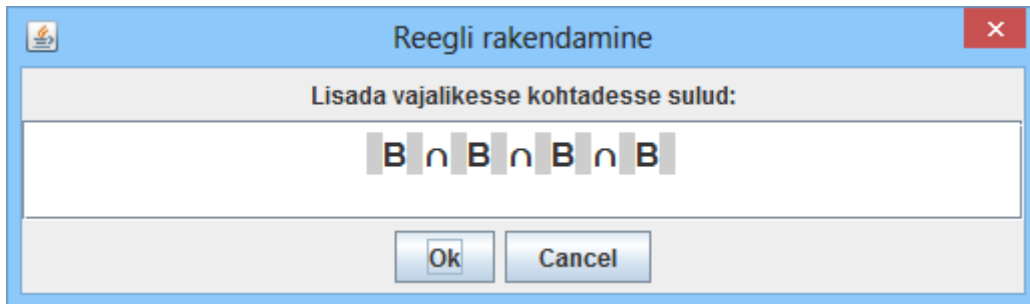
Esimesed kuus reeglit on tehtud lausearvutuse eeskujul. Esimene reegel on jäätud samaks, teised viis on analoogilised, ainult eituste, konjunktsioonide ja disjunktsioonide asemel on vastavalt täiendid, ühisosad ja ühendid. Timmu Ründali bakalaureusetöö prototüübilt on võetud vahe ning kaks sümmeetrilise vahe reeglit (komplektis vastavalt reeglid 7.–9. ehk esimese tulba viimased kaks ja teise tulba esimene). Veel on lisatud kolmas sümmeetrilise vahe reegel, mis teisendab sümmeetrilise vahe vahede ühendiks (teise tulba teine reegel).

Teise tulba 3.–6. reegel on analoogilised lausearvutuse teise tulba kahe viimase ja kolmanda tulba kahe esimese reeglga, kus jällegi eituste, konjunktsioonide ja disjunktsioonide asemel on vastavalt täiendid, ühisosad ja ühendid. Teise tulba viimane reegel on saadud Timmu Ründali prototüübist. Kolmanda tulba esimesed neli reeglit on analoogilised lausearvutuse kolmanda tulba 3.–6. reeglga juba kirjeldatud viisil.

Viimased kolm reeglit on samad lausearvutuse viimase kolme reeglga, kuid hulgateoorias tähistab märk \oplus kas ühisosa või ühendit. Viimased kaks reeglit on tehtud kahesuunaliseks. Esimese reegli puhul nendest küsib programm, millist tehtemärki soovitakse \oplus asemel kasutada (vt joonis 16), kui reeglit rakendatakse paremalt vasakule. Teise reegli rakendamisel pole tegelikult tähtis, kas rakendatakse reeglit vasakult või paremalt, kuna vastav reegel teeb teisenduse ära mõlemat pidi ning isegi siis, kui osaavaldis on keerulisem ning erinevaid sulgude ümberpaigutamise viise on mitu. Sellisel juhul küsib programm, kuhu kasutaja soovib sulge panna (vt joonis 17). Mõlemat pidi rakendamise nupp lisati vaid selleks, et tudeng ei hakkaks enne selle reegli kasutamist asjatult kasutama kommutatiivsuse reeglit (parema tulba viies). Selline olukord tekib näiteks siis, kui osavalem on kujul $(X\cup Y)\cup Z$.



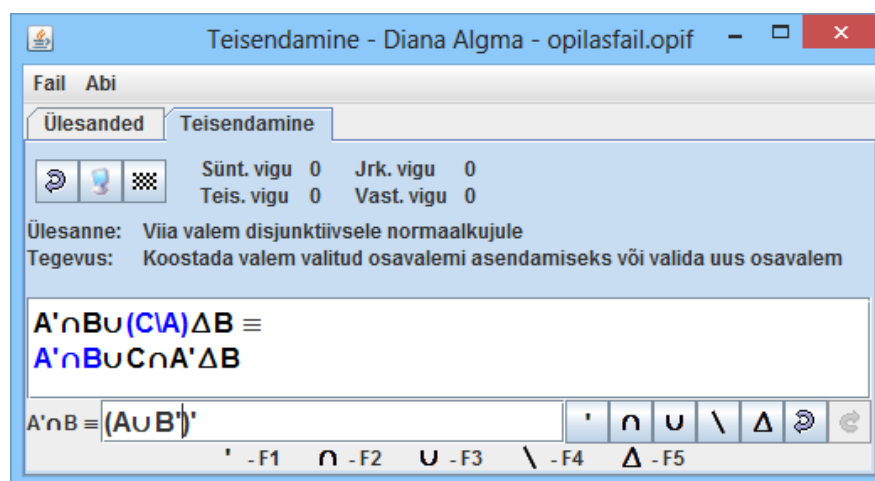
Joonis 16. Soovitud tehemärgi küsimine



Joonis 17. Sulgude lisamise dialoog

4.4 Vahetu teisendamine hulgateooria jaoks

Lausearvutuse ja predikaatarvutuse ülesannete vahetu teisendamise jaoks on kaks vahetu teisendamise akna tüüpi. Akende erinevus seisneb selles, et predikaatarvutuse ülesannete vahetuse teisendamise aknas on võimalik sisestada lisaks lausearvutuse sümbolitele (lausearvutuse tehemärgid ja suurtähed) ka predikaatarvutuse sümboleid (väiketähed, kvantorid, nurksulud ja komad). Hulgateooria ülesannete vahetu teisendamise jaoks oli vaja lisada vastav vahetu teisendamise akna tüüp, kus on nupud hulgateooria tehemärkide sisestamiseks (vt joonis 18).



Joonis 18. Vahetu teisendamise aken hulgateooria ülesannete jaoks

Tehtemärke saab sisestada nii nuppudega kui ka klahvidega F1-F5. Samuti saab täiendi ja vahe märki sisestada klaviatuurilt vastavate klahvidega ' ja \. Selline sisestusviis on samuti õppejõu programmis valemi sisestamisel.

4.5 Automaatlahendaja

S. Stroomi loodud automaatlahendaja oskas lahendada lausearvutuse ülesandeid, kus pidi valemi viima kas konjunktsiooni ja eituse, disjunktsiooni ja eituse või implikatsiooni ja eitusega kujule. Automaatlahendaja oskas lahendada ka täieliku disjunktiivse normaalkuju ülesannet. Ülejäänud ülesandetüüpide puhul saab ülesande koostamisel õppejõu programmiga küll lubada juhiste andmine, kuid teisendamise käigus tudengi programmis juhiste küsimisel annab programm teate, et juhiste ei saa anda. Eelmises versioonis programm sel puhul teadet ei andnud.

Käesoleva bakalaureusetöö jooksul täiendati automaatlahendajat nii, et see suudaks lahendada ka hulgateooria ülesandeid, kus on vaja valem viia kas ühisosa ja täiendi või ühendi ja täiendiga kujule. Lisaks oskab täiendatud automaatlahendaja lahendada hulgateooria täieliku disjunktiivse normaalkuju ülesannet.

Nende ülesannete lahendamisel on kasutajal võimalus küsida soovitusi, kui ülesande koostamisel seda lubati. Soovitused ütlevad, millise sammu peaks tegema järgmisena. Näiteks hulgateooria ülesandeid lahendades, oleks esimene samm vahede eemaldamine. Sümmeetriliste vahede eemaldamine on järgmine samm, sest kui enne eemaldataks sümmeetrilised vahed, võib vahede arv valemis suurened, mis suurendaks vajalike lahendussammude arvu, sest kõik vahed tuleb eemaldada eraldi. Seetõttu on kavalam enne eemaldada vahed ja alles seejärel sümmeetrilised vahed. Sellise ülesande lahendamisel juhul, kui valemis on mõni vahe, annab automaatlahendaja soovitus „Asenda vahed ühisosade ja täienditega“. Automaatlahendaja annab sobivaid soovitusi kuni ülesanne on lahendatud. Juhul, kui valem on juba vajalikul kujul aga kasutaja küsib soovitus, on soovitus „Valem on juba õigel kujul“.

4.6 Väiksemad muudatused

Vanas versioonis oli lausearvutuse muutujatena võimalik kasutada vaid tähti A-O. Kuna pole põhjust, miks programm ei võiks lubada ka tähti P-Z, on uues versioonis nii lausemuutujate, predikaatide kui hulkade tähistuseks võimalik kasutada tähti A-Z.

Lisaks eelmistes alapeatükkides mainitud muutustele pidi jooksvalt täiendama olemasolevaid ja lisama uusi klasse, et näiteks alamvalemi leidmine ja samasuse kontroll töötaks samuti hulgateooria puhul. Samuti tuli kontrollida, et igas meetodis, kus määrati osavalemit indeksite järgi sõnes, oleksid indeksid korrektsed, kuna mitmes kohas oli vaja indeksite määramist muuta täiendi ja eituse asendite erinevuse tõttu alamvalemi ümber.

Nii õppejõu kui tudengi programmi kasutusjuhendit täiendati, et oleks märgitud ka hulgateooria avaldiste teisendamise võimalus, vajalikud tehtemärgid ja nende sisestamise klahvid ning muud võimalikud sümbolid valemi sisestamisel. Tudengi programmi kasutusjuhendisse lisati kirje juhiste küsimise nupule.

Kuna seni kasutatud õpilasfaili faililaiend *.std* on kasutusel OpenOffice.org poolt [9], muudeti õpilasfaili faililaiendit, uueks laiendiks on *.opif*, mis tähistab õpilasfaili.

Tudengi programmi ja tulemuste akna tiitliribadele lisati lisaks failinimele ka lahendaja nimi, kui õpilasfail on avatud. Avatud õpilasfaili korral on tudengi programmi tiitliribal „Teisendamine - <õpilase ees- ja perekonnanimi> - <faili nimi>“, näiteks „Teisendamine - Mari Maasikas - proov.opif“. Kui õpilasfaili avatud pole, on tiitliribal „Teisendamine“. Tulemuste akna tiitliribal on „Tulemused - <õpilase ees- ja perekonnanimi> - <faili nimi>“.

Teisenduste logis on näha eduka katse lõpus vigade arv. Üheks veatüübiks oli vanas versioonis „Ebapedagoogiline viga“, mis aga ei anna edasi seda, milles täpselt viga oli. Selle vea tekst asendati tekstiga „Samaväärne asendus aga järjekorraviga“. Lisaks muudeti teisenduste logi akent kõrgemaks.

Õppejõu programmis uue ülesande loomise aknas asendati „Ülesande kirjeldus“ sobivama tekstiga „Ülesande tekst“.

Nii õppejõu kui tudengi programmile lisati tiitelleht (vt joonis 19), mis avaneb programmi akna peal ning mille sulgemiseks tuleb vajutada nupule „Alusta“. Tiitellehel on Tartu Ülikooli arvutiteaduse instituudi logo, programmi nimi, versiooni loomisaasta ja autorite nimed.



TARTU ÜLIKOOL
arvutiteaduse instituut

Matemaatilise loogika ja hulgateooria avaldiste teisendamise keskkond

Õpilase programm

Alusta

2016

Vahur Vaiksaar
Sander Stroom
Diana Algma

Joonis 19. Õpilase programmi tiitelleht

Kokkuvõte

Aastal 2003 programmeeris Vahur Vaiksaar oma bakalaureusetööna Lausearvutuse ja predikaatarvutuse valemite teisendamise õpiprogrammi, mida kasutatakse Tartu Ülikooli matemaatilise loogika sissejuhatavates kursustes. Aastal 2013 täiendas seda programmi Sander Stroom oma bakalaureusetööna lisades programmi automaatlahendaja, mis annab kasutajale juhiseid, täiendades lahendusfaile, et lahenduskäikude kohta salvestataks ja oleks võimalik vaadata täiendavat informatsiooni, ning lisades võimaluse programmi kasutajaliidest teistesse keeltesse tõlkida.

Antud bakalaureusetööna lisati aastal 2013 valminud versioonile võimalus koostada ja lahendada hulgateooria avaldiste teisendamise ülesandeid. Tartu Ülikooli matemaatilise loogika kursustes teisendatakse samuti hulgateooria avaldiseid ning seda pidi tegema varem kas paberil või mõne tekstiredaktori abiga, kuna programm, mille abil seda oleks mugavam teha, Tartu Ülikoolil puudus. Selle bakalaureusetöö eesmärgiks on täiendada olemasolevat programmi nii, et sellega oleks võimalik õpetada ja õppida ka hulgateoreetiliste avaldiste teisendamist, lihtsustades ja kiirendades sellega hulgateooria teisendamise õppimist ja õpetamist.

Programmi eelmises versioonis on kümme lausearvutuse ja kaheksa predikaatarvutuse ülesandetüüpi. Antud bakalaureusetööna lisati programmi järgmised hulgateooria ülesandetüübid:

- 1) antud valemi avaldamine ühisosa ja täiendi kaudu;
- 2) antud valemi avaldamine ühendi ja täiendi kaudu;
- 3) antud valemi viimine täielikule disjunktiiivsele normaalkujule;
- 4) antud valemi viimine disjunktiiivsele normaalkujule;
- 5) hulgateooria vaba teisendus.

Programm oskab anda lahendajale ka juhiseid nendest esimese kolme ülesandetüübi lahendamisel.

Summary

In 2003 Vahur Vaiksaar created a program Algebraic Manipulation Assistant for Propositional Logic and Predicate Calculus, which is used in the University of Tartu for teaching courses of mathematical logic. In 2013 Sander Stroom improved the program as his bachelor thesis adding an automatic task solver that would give instructions to the user on what steps to take next. Stroom also improved the transformation log and added the possibility to translate the program's user interface.

This bachelor thesis extends the 2013 version by adding the opportunity to create and solve set theory expression conversion exercises. In courses of mathematical logic in the University of Tartu expression manipulation in the algebra of sets is also done but due to the lack of a program to make it easier the expression manipulation is done on paper or with various text editors. The purpose of this bachelor thesis is to add set theory exercises to the existing program to make teaching and learning set theoretical expression manipulation easier and faster in the University of Tartu.

The new version of the program allows to create and solve set theoretical expression conversion exercises in addition to algebraic manipulation exercises in propositional logic and predicate calculus. Exercises added with this bachelor thesis are the following:

- 1) expressing the given formula by intersection and complement;
- 2) expressing the given formula by union and complement;
- 3) finding the disjunctive normal form of the given formula;
- 4) finding the full disjunctive normal form of the given formula;
- 5) free transformation of set theory calculus.

Chapter 1 gives an overview of a similar program SetSails!, that enables proving equations of set algebra and Boolean algebra. Chapter 2 describes where the program that was originally created by V. Vaiksaar is used and lists the exercises that are possible to create and solve with the program. Chapter 3 describes the 2013 version of the program and how it can be used. Chapter 4 consists of detailed descriptions of the improvements made to the program.

Kasutatud materjalid

- [1] Vaiksaar, V. (2003). Lausearvutuse ja predikaatloogika valemite teisendamise õpiprogramm, bakalaureusetöö.
- [2] Stroom, S. (2013). Lause- ja predikaatarvutuse valemite teisendamise õpiprogrammi täiendamine, bakalaureusetöö.
- [3] Semiautomatische Analyse individueller Lernprozesse in der Mathematik, <http://sail-m.de/> (vaadatud 11.05.16).
- [4] Oja, P. (2006). Hulgateooria, Tartu.
- [5] Praktikumiuülesannete kogu 2015. sügissemester, Tartu Ülikooli kursus Matemaatiline maailmapilt (MTMM.00.342).
- [6] Carette, J. (2004, juuli). Understanding expression simplification, Proceedings of the 2004 international symposium on Symbolic and algebraic computation, lk 72-79, ACM.
- [7] Prank, R. Hulgateooria konspekt, Moodle'i kursus Sissejuhatus matemaatilisse loogikasse (MTAT.05.109).
- [8] Ründal, T. (2012). Hulgateoreetiliste avaldiste teisendamisega seotud ülesannete tüübid diskreetse matemaatika veebikeskkonna jaoks, bakalaureusetöö.
- [9] FILExt, A free online resource by Uniblue, <http://filext.com/file-extension/STD> (vaadatud 01.05.16).

Lisa 1. Kokkupakitud fail programmi, keelefailide ja näiteülesandekoguga

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks

Mina, **Diana Algma**,

(autori nimi)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

Hulgateooria ülesannete lisamine matemaatilise loogika avaldiste teisendamise keskkonnale,

(lõputöö pealkiri)

mille juhendaja on Rein Prank,

(juhendaja nimi)

1.1.reprodutseerimiseks säilitamise ja üldsusele kättesaadavaks tegemise eesmärgil, sealhulgas digitaalarhiivi DSpace-is lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2.üldsusele kättesaadavaks tegemiseks Tartu Ülikooli veebikeskkonna kaudu, sealhulgas digitaalarhiivi DSpace´i kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest tulenevaid õigusi.

Tartus, **12.05.2016**