

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKA TEADUSKOND
Arvutiteaduse instituut
Infotehnoloogia eriala

Marten Hennoch

**Venni diagrammidega seotud ülesannete tüübid
diskreetse matemaatika veebikeskkonna jaoks**

Bakalaureusetöö (6 EAP)

Juhendaja: dots. R. Prank
. Margus Niitsoo

Autor: "....." juuni 2012

Juhendaja: "....." juuni 2012

Juhendaja: "....." juuni 2012

Lubada kaitsmisele

Professor "....." juuni 2012

Tartu 2012

Sisukord

Sissejuhatus.....	4
1. Probleemi taust.....	5
1.1. Ainest	5
1.2. Venni diagrammid - nende kasutamine ja probleemid.....	5
1.2.1. Hulgateooria põhitehted Venni diagrammidega	6
1.2.2. Harjutamine.....	8
1.3. Olemasolev süsteem.....	9
1.4. Lisatav funktsionaalsus	10
2. Rakenduse kirjeldus	11
2.1. Ülesannete tüübid ja Venni diagrammid rakenduses	11
2.2. Ülesannetekogu ja ülesande valimine	11
2.3. Valemi leidmine diagrammi järgi	13
2.4. Diagrammi leidmine valemi järgi	15
2.5. Võrdsuse kontroll	17
2.6. Alalhulgaks olemise kontroll	19
2.7. Ülesannete lisamine.....	20
3. Realisatsioon	23
3.1. Kasutatud tehnoloogiad.....	23
3.1.1. Veebipõhine süsteem	23
3.1.2. <i>HTML5</i>	23
3.1.3. Canvas.....	23
3.1.4. JavaScript.....	23
3.1.5. <i>JQuery</i>	24
3.1.6. <i>PHP</i>	24
3.2. Realisatsioon	24
3.2.1. Diagramm	24
3.2.2. Hulkade leidmine	25
3.2.3. Joonise osade värvimine	26
3.2.4. Olemasolevasse süsteemi lisamine	26

Kokkuvõte.....	27
Summary	28
Viited.....	29
Lisad.....	30
Lisa 1: CD plaat koos programmi lähtekoodi ja paigaldamisõpetusega	30

Sissejuhatus

MTAT.05.110 Diskreetse matemaatika elemendid on Tartu Ülikoolis õpetatav aine, mis on mõeldud 1. aasta infotehnoloogia, informaatika, arvutitehnika ja matemaatika tudengitele. Aine võtab osa palju tudengeid ja tegemist on esimese aasta ühe keeruliseima ainega. Seetõttu on aine praktikumid jagatud tahvli- ja arvutipraktikumideks, kus viimastes kasutatakse õppeprotsessi lihtsustamiseks erinevaid õpiprogramme. Programmide kasutamine võimaldab anda tudengitele iga nädal teatud hulga ülesandeid, mille edukat läbimist saab hiljem kontrollida. Nii saavad õpilased kiiresti rakenduse käest tagasisidet ja õppejõududel on võimalik hoida kokku palju aega, mis muidu kuluks ülesannete parandamiseks.

Hetkel on kasutusel kaks programmi: üle veebi töötav programm tõeväärtustabeli ülesannete lahendamiseks ja Java-programm teisendusülesannete jaoks. Hulgateooria ülesannete lahendamiseks võimalused puuduvad. Bakalaureusetöös kirjeldatakse, kuidas olemasolevale veebis töötavale süsteemile lisati neli Venni diagrammidega lahendatavate ülesannete tüüpi: valemi leidmine diagrammi järgi, diagrammi leidmine valemi järgi, hulkade võrdsuse kontroll ja alamhulgaks olemise kontroll. Rakendusega on Venni diagrammide joonistamine kiire ja mugav, mis omakorda peaks aitama õpilasel materjali edukamalt omandada.

Esimeses peatükis tutvustatakse pikemalt aine olemust, Venni diagrammide kasutamist ja nendega seonduvaid probleeme, olemasolevat süsteemi ja vaadeldakse põgusalt lisatavat funktsionaalsust. Teises peatükis kirjeldatakse iga ülesandetüüpi eraldi ja tuuakse lahenduskäigu ja programmi kohta näiteid. Kolmandas peatükis räägitakse kasutatud tehnoloogiatest ja rakenduse realiseerimisest. Bakalaureusetöö lisas on kaasas CD-l olev rakendus ja selle installeerimisjuhend.

1. Probleemi taust

1.1. Aineist

MTAT.05.110 Diskreetse matemaatika elemendid on Tartu Ülikoolis õpetatav aine, mis on mõeldud 1. aasta infotehnoloogia, informaatika, arvutitehnika ja matemaatika tudengitele. Aine eesmärgiks on, et tudeng omandaks põhiteadmised informaatikas ette tulevatest peamistest teoreetilistest mõistetest ning arendaks nende kasutamisoskust. Aine jaguneb üldiselt kaheks suuremaks teemaks: lausearvutuseks ja hulgateooriaks. Lausearvutuse osas vaadeldakse lausearvutuse põhitehteid, lausearvutuse valemite tõeväärtuste leidmist, lausearvutuse samaväärsusi ja valemite teisendamist. Hulgateooria osas vaadeldakse hulgateooria põhimõisteid ja tehteid hulkadega. Aine auditoorne töö on jagatud loenguteks ja praktikumideks. Osa aine praktikume on arvutipraktikumid, kus ülesannete lahendamiseks kasutatakse erinevad õpiprogramme. Arvutipraktikumides kasutatakse praegu kahte rakendust: üle veebi töötav programm tõeväärtustabeli ülesannete lahendamiseks ja Java-programm teisendusülesannete jaoks. Praegu ei ole kasutusel programme hulgateooria ülesannete jaoks. Bakalaureusetöös lisatakse olemasolevale veebis töötavale süsteemile Venni diagrammidega lahendatavate ülesannete baastüübid: valemi leidmine diagrammi järgi, diagrammi leidmine valemi järgi. Baastüüpide kaudu saab realiseerida ka kaks hulcade võrdlemisel põhinevat ülesandetüüpi: võrduse kontroll ja alamhulgaks olemise kontroll.

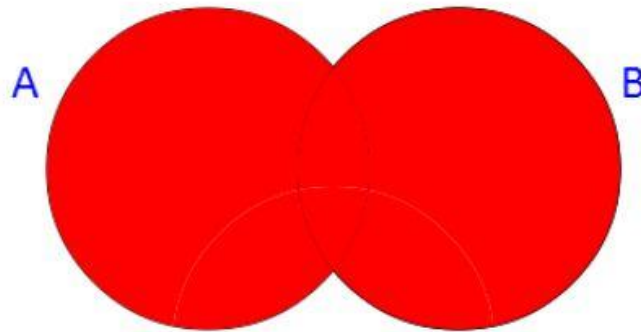
1.2. Venni diagrammid - nende kasutamine ja probleemid

Venni diagramme kasutatakse hulgateoreetiliste avaldiste abil moodustatud hulkade graafiliseks esitamiseks. Venni diagrammid koosnevad enamasti osaliselt kattuvatest ringidest, millega kujutatakse hulkasid ning nende omavahelisi suhteid tasandil. Hulgale vastava ringi sisemus kujutab sümboolselt hulga elemente ja ringist välja jääv ala elemente, mis hulka ei kuulu. Selline esitusviis on väga selge ja võimaldab kergesti piltlikustada tehteid hulkadega. Just selle tõttu osutub hulgateoreetilise ülesannete puhul Venni diagramm tihti esimeseks õpilaste poolt valitud lahendusmeetodiks.

1.2.1. Hulgateooria põhitehted Venni diagrammidega

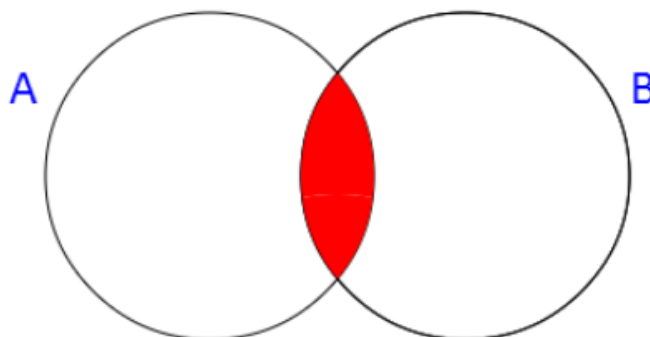
Kuna rakenduse peamine funktsionaalsus põhineb tehetel hulkadega siis defineerin siin ära peamised hulgateooria tehted: ühend, ühisosa, vahe, sümmeetriline vahe ja täiend. Sealjuures näitan ka, kuidas need hulgad Venni diagrammil on kujutatud. [3]

Def. Hulkade A ja B ühendiks e. summaks nimetatakse hulka $A \cup B$, mille moodustavad kõik elemendid, mis kuuluvad vähemalt ühte hulkadest A ja B. Vt. Joonis 1.



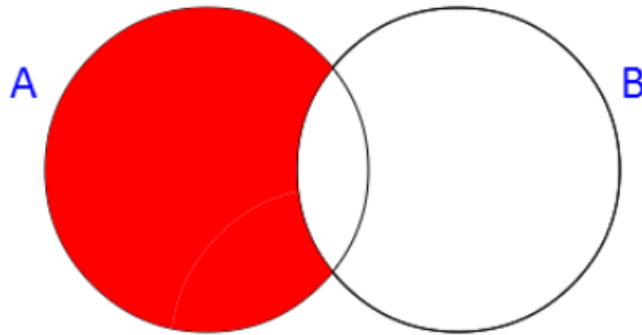
Joonis 1. Hulkade A ja B ühend

Def. Hulkade A ja B ühisosaks e. lõikeks nimetatakse hulka $A \cap B$, mille moodustavad kõik elemendid, mis kuuluvad nii hulka A kui hulka B. Vt. Joonis 2.



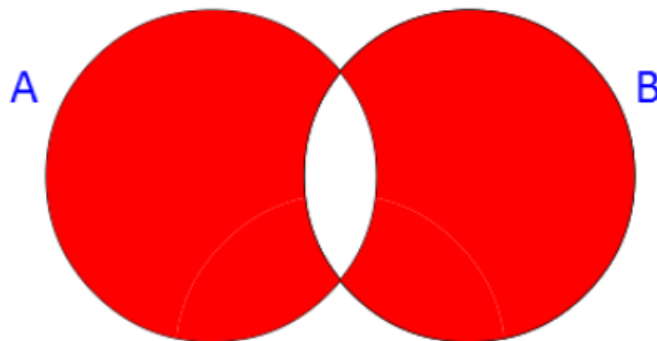
Joonis 2. Hulkade A ja B ühisosa

Def. Hulkade A ja B vaheks nimetatakse hulka $A \setminus B$, mille moodustavad kõik elemendid, mis kuuluvad hulka A, aga ei kuulu hulka B. Vt. Joonis 3.



Joonis 3. Hulkade A ja B vahe

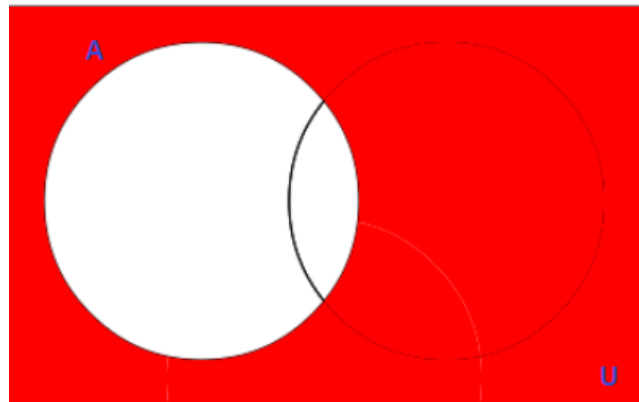
Def. Hulkade A ja B sümmeetriliseks vaheks nimetatakse hulka $A \Delta B$, mille moodustavad kõik elemendid, mis kuuluvad parajasti ühte kahest hulgast A ja B. Vt. Joonis 4.



Joonis 4. Hulkade A ja B sümmeetriline vahe

Paljudes matemaatika valdkondades on uuritavateks objektideks peamiselt teatud ühe hulga elemendid. Näiteks arvuteoorias on selliseks hulgaks naturaalarvude hulk, matemaatilises analüüsis reaalarvude hulk jne. Hulgateoorias nimetatakse niisugust kõiki vaadeldavaid elemente sisaldavat hulka universaalseks hulgaks. Tähistagu universaalset hulka täht U . Hulga A täiendi A' moodustavad kõik need universaalse hulga elemendid, mis ei kuulu hulka A .

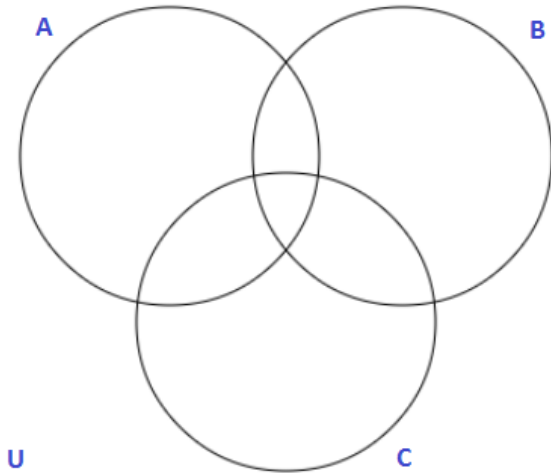
Kui uuritavad avaldised sisaldavad ka täiendi tehet, siis Venni diagrammil kujutatakse universaalset hulka ristkülikuna, mille sees on avaldises esinevaid hulki kujutavad ringid. Täiendi diagramm on kujutatud Joonisel 5.



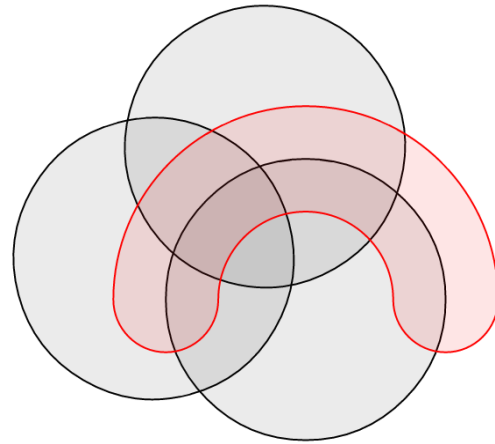
Joonis 5. Hulga A täiend

1.2.2. Harjutamine

Kui ülesandega seotud hulgad välja joonistada, on lihtsam probleemist aru saada ja lahenduse peale mõelda. Venni diagrammid sobivad eelkõige väheste hulkade arvuga ülesannete lahendamiseks, sest juba nelja hulgaga Venni diagrammi joonistamine on suhteliselt keeruline (Joonis 2). Aine raames rohkem kui 3 hulgaga (Joonis 6) ülesandeid õpilastele lahendada ei anta, nii et liigse keerulisusega probleeme ei teki.



Joonis 6. Kolme hulgaga Venni diagramm



Joonis 7. Nelja hulgaga Venni diagramm[1]

Harjutamisel, kus lahendatakse palju ülesandeid ja vigade tekkimise tõenäosus on kõrge, võib kogu aeg uue diagrammi joonistamine tülikaks osuda. Paranduste tegemine juba olemasolevale joonisele teeb pildi väga kirjuks ja raskendab arusaamist. Kuna Venni diagrammi tugevaks küljeks ongi just selle selgus, siis paberi peal paljude seda tüüpi ülesannete lahendamine ei pruugi olla väga selge ja see võib raskendada ülesandest arusaamist. Lisaks kulutab see paberiruumi ja võtab ka palju aega. Probleemi lahenduseks ongi programmi koostamine, mille abil on Venni diagramme võimalik joonistada ja nendega seonduvaid ülesannete tüüpe lahendada. Programm on peamiselt mõeldud harjutamiseks, kuigi seda saab ka kontrolltöodes kasutada õpilaste teadmiste kontrollimiseks. Automaatne ülesannete parandamine ja kohene tagasiside hoiavad kokku ka õppejõudude aega.

1.3. Olemasolev süsteem

Hetkel on kasutusel üle veebi töötav süsteem tõeväärtustabeli ülesannete lahendamiseks. See valmis 2010 a. Alari Luku bakalaureusetöö raames [2]. Süsteemis on realiseeritud 6 lausearvutusega seotud ülesandetüüpi: tõeväärtustabeli arvutamine, samaselt tõesuse kontroll, kehtestatavuse kontroll, samaväärsuse kontroll, järelдумise kontroll ja valemi leidmine veeru järgi. Lisaks ülesannetele on süsteemis olemas ka kasutajaliides õppejõu ja õpilase jaoks. Õpilane saab süsteemi siseneda talle õppejõu poolt antud kasutajaga. Kasutajal on võimalik

avada ülesannetekogusid, mis sisaldavad erinevaid õppejõu poolt valitud ülesandeid. Iga ülesande juures on ära näidatud võimalik saadav punktide arv ja mitu korda seda on lahendatud. Poolelijäänud katseid on võimalik jätkata. Õppejõu liides võimaldab teha erinevaid adminisatoorseid toiminguid: õpilaste haldamine, ülesannetekogude haldamine, uute ülesannete lisamine jne. Olemasolevat süsteemi uurides selgus, et sinna on võimalik lisada käesolevas bakalaureusetöös kirjeldatavad ülesandetüübid. Uusi tüüpe saab hallata ja lisada sarnaselt juba olemasolevatega. Olemasoleva süsteemi kasutamisel on mitmeid eeliseid. Esiteks ei pea õppejõud õppima kasutama mõnda uut rakendust, vaid saab juba tuttavas keskkonnas tegutseda. Tudeng ei pea ka mitme programmi vahel vahetama, et erinevaid ülesandetüüpe lahendada.

1.4. Lisatav funktsionaalsus

Programm joonistab vajaliku diagrammi ja õpilasel jääb ainult üle valida õiged hulgad või sisestada vastav avaldis. Rakendus oskab vastust kontrollida ja vigade korral ka tagasisidet anda. Realiseeritud on 4 ülesandetüüpi: valemi leidmine diagrammi järgi, diagrammi leidmine valemi järgi, hulkade võrdsuse kontroll ja alamhulgaks olemise kontroll. Programmi joonistatud diagramm jaguneb osadeks. Näiteks kolme hulgaga Venni diagramm sisaldab maksimaalselt 8 osa (Joonis 6). Esimese ülesande tüübi korral antakse ette avaldis ja selle järgi tuleb jooniselt õiged piirkonnad märkida. Diagrammile hulkade märkimiseks tuleb õpilasel hiirega vajutada valitud osal, mille peale programm selle ära värvib. Antud osa peale uuesti vajutades taastatakse esialgne seis. Nii viisi osi valides saab õpilane kergesti ja kiiresti enda jaoks läbi proovida erinevaid võimalikke lahendusi. Lõpuks, kui õpilane arvab, et tema valitud piirkonnad vastavad avaldises nõutule, saab diagrammi esitada kontrollimiseks, mille peale programm ütleb, kas see vastab ülesandes nõutule. Teise ülesande tüübi korral joonistab rakendus ette diagrammi ja õpilane peab sellele vastava avaldise sisestama. Sarnaselt eelmisele üleseandetüübile kontrollib rakendus vastust ja annab tagasisidet. Mõlemad hulkade võrdlemisel põhinevad ülesanded kasutavad ära esimese tüübi Venni diagrammi joonistamise funktsionaalsust. Kasutajale kuvatakse kaks tühja diagrammi ja kaks avaldist. Avaldiste järgi tuleb korrektselt täita diagrammid ja seejärel võtta vastu otsus. Hulkade võrdsuse kontrollimisel siis tuleb otsustada kas hulgad on võrdsed ja alamhulgaks kuulumise tüübi puhul, kas üks hulk on teise hulga alamhulk. Rakendusel on realiseeritud kõik peamised hulgateooria põhitehted: ühend, ühisosa, vahe, sümmeetriline vahe ja täiend. Ülesandeid saab teha kuni kolme hulgaga.

2. Rakenduse kirjeldus

2.1. Ülesannete tüübid ja Venni diagrammid rakenduses

Rakendusel on realiseeritud 2 peamist ülesandetüüpi: valemi leidmine diagrammi järgi ja diagrammi leidmine valemi järgi. Lisaks neile on olemas 2 hulcade võrdlemisel põhinevat tüüpi: võrdsuse kontroll ja alamhulgaks olemise kontroll. Olemasolevas süsteemis on võimalik tõeväärtustabeleid kasutades lahendada erinevaid ülesandetüüpe. Kirjutades välja kahe valemi tõeväärtusveerud, saab nende kohta võtta vastu otsuseid. Näiteks, kas valemid on samaselt tõesed või kas üks valem järeldeb teisest. Sarnaselt tõeväärtustabelile saab niiviisi ära kasutada ka Venni diagramme. Andes ette kaks diagrammi saab õpilane need ära täita ja võtta vastu otsuse. Hulkade võrdsuse kontrollimisel, kas hulgad on võrdsed või ei ole ja alamhulgaks olemise puhul, kas hulk on teise hulga alamhulk või ei ole. Toon järgnevalt ära programmi üldkirjelduse ülesandetüüpide kaupa ja tutvustan ka natuke olemasoleva süsteemi kasutajaliidest.

2.2. Ülesannetekogu ja ülesande valimine

Õppejõud saab luua erinevaid ülesannetekogusid. See võimaldab teha näiteks eraldi kogu lausearvutuse ja hulgateooria osa jaoks. Rakendusse saab siseneda õppejõult saadud kasutajatunnuse ja salasõna abil. Seejärel on õpilasel võimalik näha ülesannetekogusid ja neist lahendamiseks mõni valida. Seda protsessi on kujutatud joonisel 8.

Ülesannete kogud



Hulgateooria



Lausearvutus

Joonis 8. Ülesannetekogu valimine

Peale ülesannetekogu valmist kuvatakse kasutajale kogus olevad ülesanded. Lisaks on näha ülesannete kohta erinevat informatsiooni: “Ok”, “Punkte”, “Õigesti”, “Valesti” ja “Katkestatud” (Joonis 9). Väli “Ok” näitab, kas ülesannet on edukalt lõpetatud ja uuesti tegemist ei vaja, seda tähistatakse rohelise ikooniga vt. joonis 10. “Õigesti” ja “Valesti” väljad näitavad mitu korda on lahendamisel edukas oldud ja mitu korda mitte. Katkestatud ülesandeid saab jätkata, kui vajutada “Katkestatud” lingi peale. Selle peale kuvatakse kõik pooleli olevad katsed vt. joonis 11.

Tõeväärtustabeli ülesannete lahendamise keskkond						Sisse logitud: Marten Hennoch
						Loqi välja
Ülesannete kogu: Hulgateooria						Tagasi ülesannete kogude juurde
Nr.	Ülesanne	Ok	Punkte	Õigesti	Valesti	Katkestatud
1.	Venni diagrammidega valemite võrdsuse kontroll valem: $A \cup B = B \cup A$		- / 2	-	-	6
2.	Alamhulgaks olemise kontroll valem: $A \cap B \subset A \cup B$		- / 2	-	-	1
3.	Valemi järqi diagramm valem: $A \cup B$		- / 2	-	-	-
4.	Diagrammi järqi valem		- / 2	-	-	-
Punkte kokku:			0 / 8			

Joonis 9. Ülesanded

3.	Venni diagrammidega valemite võrdsuse kontroll valem: $A \cup B = B \cup A$		2 / 2	1	-	1
----	--	---	-------	---	---	----------

Joonis 10. Läbitud ülesanne

Katkestatud lahendamised:

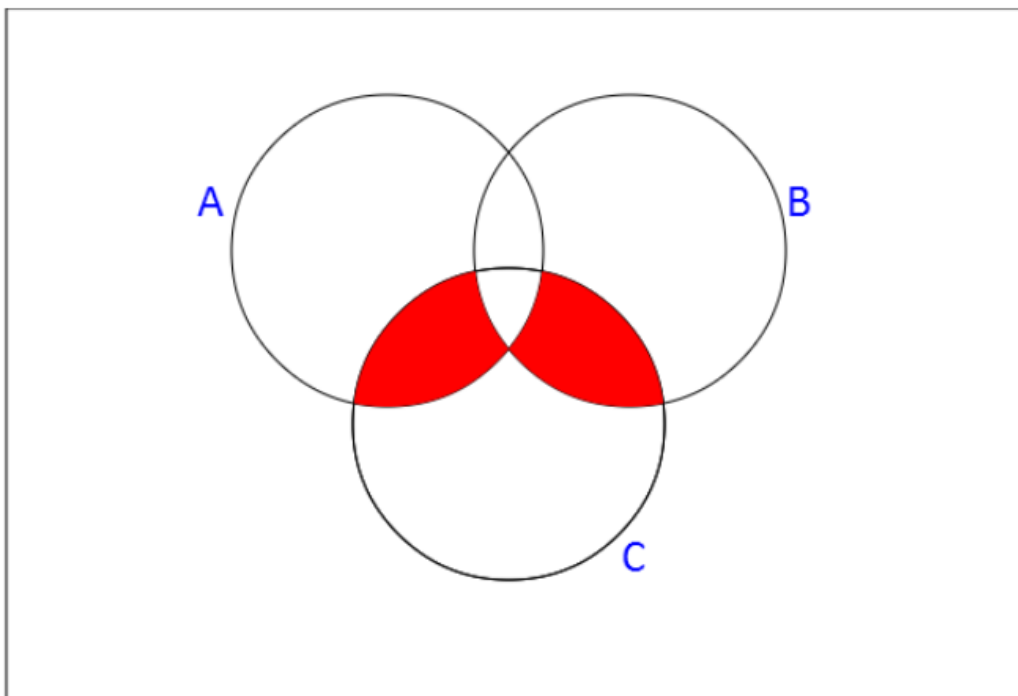
Nr.	Alustatud	Vigu kokku	Järjekorra vigu	Väärtuste vigu	Süntaksi vigu	Vastamise vigu	
1	2012-05-13 17:55:23	-	-	-	-	-	Jätkan
2	2012-05-13 17:56:24	-	-	-	-	-	Jätkan
3	2012-05-13 20:52:27	-	-	-	-	-	Jätkan

Joonis 11. Pooleli olevad katsed

2.3. Valemi leidmine diagrammi järgi

Selle tüüpi puhul joonistatab programm diagrammi ja kasutaja peab vastuseks sisestama joonisele vastava avaldise. Õigeks vastuseks loetakse igat sellist avaldist, millele vastab etteantud diagramm. Korrektseid vastuseid võib olla mitu. Võimalik on teha kahte sorti vigu: samaväärsusvigu ja süntaksivigu. Vigade üle peetakse arvet ja vigade ülempiiri ületanud ülesanne tuleb punktide saamiseks uuesti teha. Lisaks joonisele on olemas tehtmärkide tabel, kus nupuvajutuse või kopeerimise abil on võimalik märke sisestada. Vastuse sisestamiseks on lahter ja esitamiseks nupp. Kasutajale kuvatav pilt on kujutatud joonisel 12.

Sisestada antud diagrammile vastav avaldis



Tehtmärgid

Klahv:	1	2	3	4	5
Kopeeri:	∪	∩	Δ	\	'

Vastus:

Joonis 12. Valemi leidmine diagrammi järgi

Lisaks on kasutaja abistamiseks lehe paremas servas väike infokast (Joonis 13), kus näeb lubatud vigade arvu, tehtud vigu ja mõningast abiinfot.

Vead		
	Lubatud	Tehtud
Vigade arv kokku:	2	0
Süntaksivead:	1	0
Samaväärsusvead:	1	0

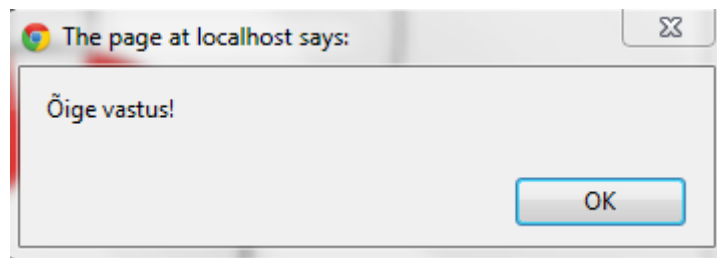
Valikud

[F10] Abiinfo

[Esc] Välju ülesandest

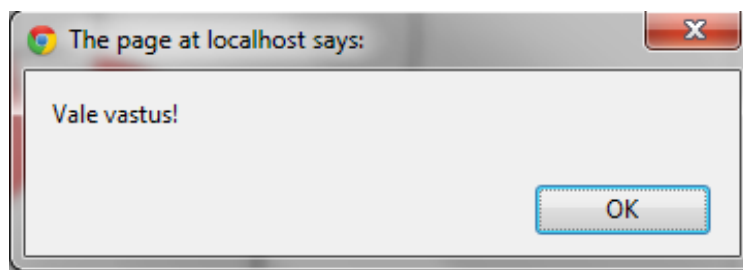
Joonis 13.

Peamine suhtlus rakenduse ja kasutaja vahel toimub dialoogiaknaid kasutades. Õige vastuse korral kuvatakse joonisel 14 näha olev teade ja peale "OK" nupu vajutamist suunatakse kasutaja tagasi ülesande valimise aknasse.



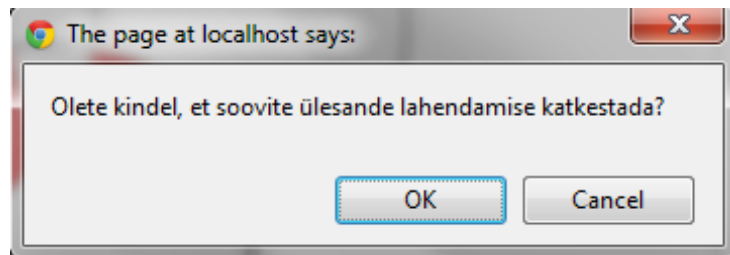
Joonis 14. Õige vastuse teade

Vale vastuse sisestamisel muudetakse tehtud vigade arvu ja kuvatakse sellekohane teade vt. joonis 11.



Joonis 15. Vale vastuse teade

“OK” nuppu vajutades saab kasutaja jätkata ülesande lahendamist või kui lahendamist ei soovi jätkata siis on võimalik “ESC” nuppu vajutades ülesandest väljuda vt joonis 16.



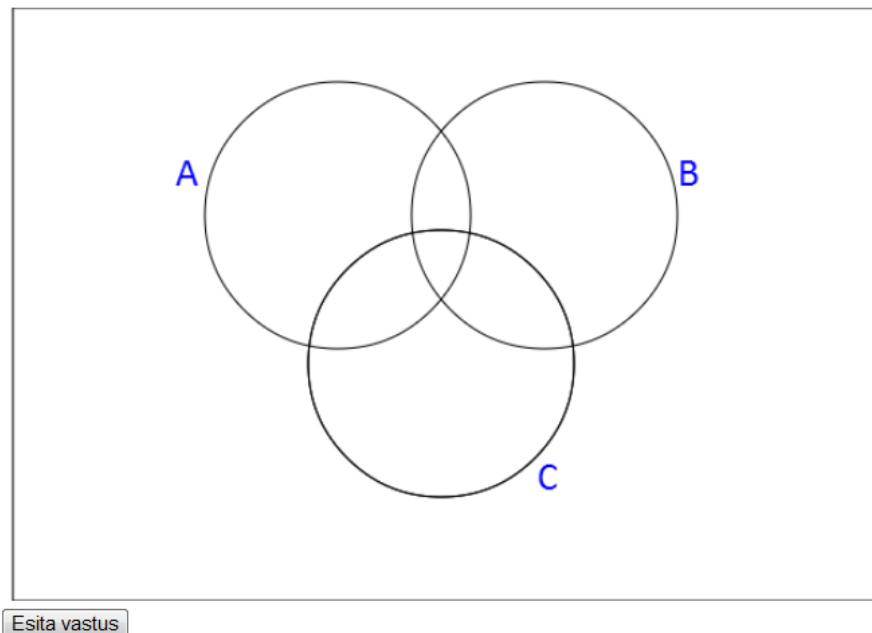
Joonis 16. Ülesande katkestamine

2.4. Diagrammi leidmine valemi järgi

Selle tüübi puhul antakse kasutajale ette avaldis ja joonistatakse ka tühi diagramm. Õpilane peab ülesande lahendamiseks joonisele kandma etteantud avaldisele vastava hulga Venni diagrammi. Joonise eri osade värvimiseks piisab hiirevajutusest osa peal. Enne vastuse esitamist saab joonist muuta kasutaja poolt tahetud arv kordi, programmi poolt ühtegi piirangut selle jaoks ette seatud pole. Kasutajale esialgselt kuvatav tühi diagramm on kujutatud joonisel 17.

Märkida piirkonnad, mis kuuluvad hulka:

Piirkondade valimiseks vajutada hiirega diagrammi eri osadel.



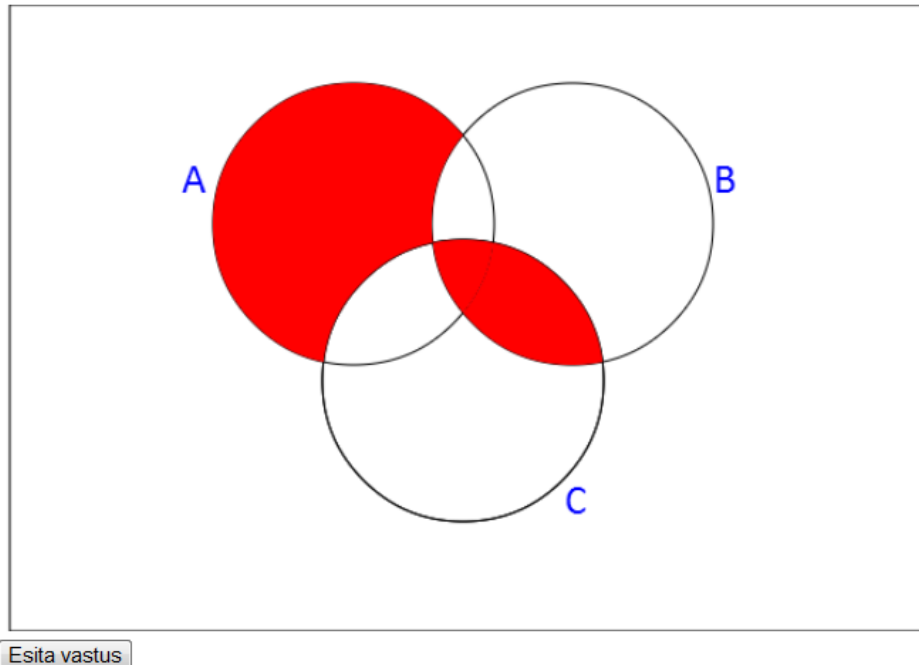
Esita vastus

Joonis 17. Tühi diagramm

Värvitud piirkondadega diagramm on näha joonisel 18.

Märkida piirkonnad, mis kuuluvad hulka:

Piirkondade valimiseks vajutada hiirega diagrammi eri osadel.



Joonis 18. Osaliselt täidetud diagramm

Avaldisele mitte vastava piirkonna vastuseks esitamise korral kuvatakse veateade ja peale "OK" nupu vajutamist on võimalik poolelioleva joonise muutmist jätkata. Veateadet on kujutatud joonisel 15. Õige diagrammi esitamise ja ülesande katkestamise korral kasutatakse sarnaselt eelmisele tüübile joonised 14 ja 16.

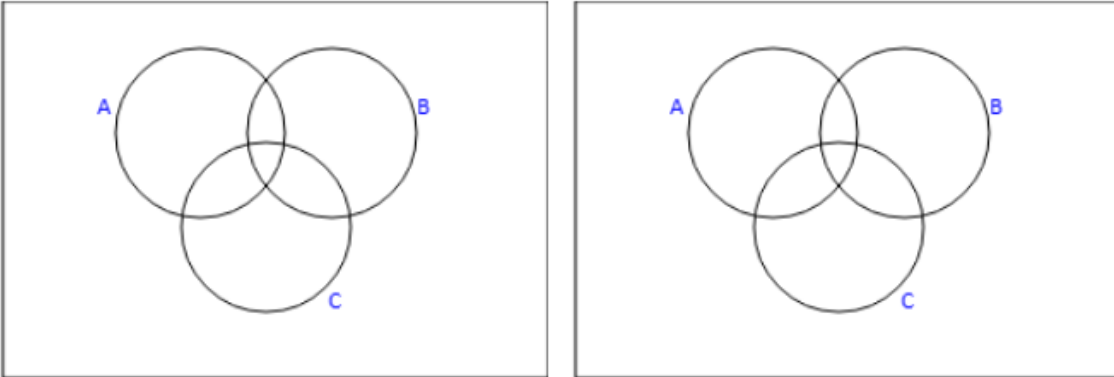
2.5. Võrdsuse kontroll

Hulkade võrdsuse kontroll on esimene kahest hulkade võrdlemisel põhinevast ülesandetüübist. Peamine erinevus kahest põhitüübist on see, et ühe Venni diagrammi asemel tuleb täita kaks. Lisaks tuleb ka jooniste põhjal võtta vastu otsus, kas hulgad on võrdsed või ei ole. Kasutajale kuvatakse kaks avaldist ja kaks diagrammi. Ülesande algseisu on kujutatud joonisel 19.

Kontrollida, kas antud avaldised on võrdsed.
Selleks tuleb teha mõlema valemi kohta Venni diagramm ja valida õige vastus.
Mõlemad diagrammid peavad olema korrektselt täidetud

1. avaldis

2. avaldis

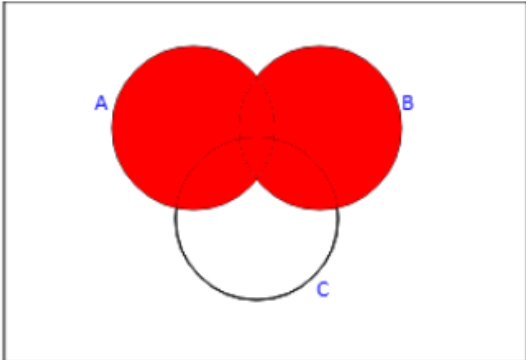


Joonis 19. Hulkade võrdlemise ül. algseis

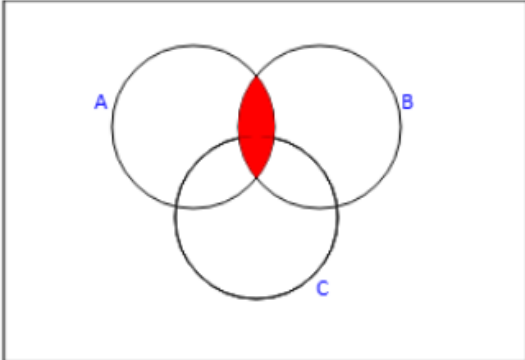
Selle ülesandetüübiga lisandub ka teine veatüüp - vastuse viga. Vastuse viga saab tekkida siis, kui mõlemad joonised on korrektselt täidetud, aga vastu võetud otsus hulkade kohta on vale. Hulkade võrdsuse kontrollimisel on sellise vea tekkimine väga ebatõenäoline, aga siiski võimalik. Tihedamini võib seda esineda alamhulgaks olemise kontrollimise tüübi puhul. Samaväärsusvea tekkimiseks piisab, kui üks esitatud diagrammidest on vale. Vea esinemisel teatatakse kasutajale ka, millised joonised on valed, kas esimene, teine või mõlemad. Joonisel 20 on ära toodud olukord, kus esimene diagramm on korrektne, aga teine on vale. Joonisel 21 on näidatud ka sellise olukorra kohta kuvatav teade ja joonisel 22 uue veatüübiga täiendatud abikast.

Kontrollida, kas antud avaldised on võrdsed.
 Selleks tuleb teha mõlema valemi kohta Venni diagramm ja valida õige vastus.
 Mõlemad diagrammid peavad olema

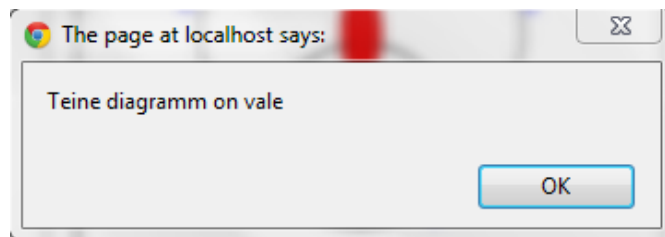
1. avaldis



2. avaldis



Joonis 20. Ebakorrektne teine diagramm



Joonis 21. Vale diagrammi teade

Vead		
	Lubatud	Tehtud
Vigade arv kokku:	2	0
Samaväärsusvead:	1	0
Vastamisevead:	1	0
Valikud		
[F10]	Abiinfo	
[Esc]	Välju ülesandest	

Joonis 22. Abikast

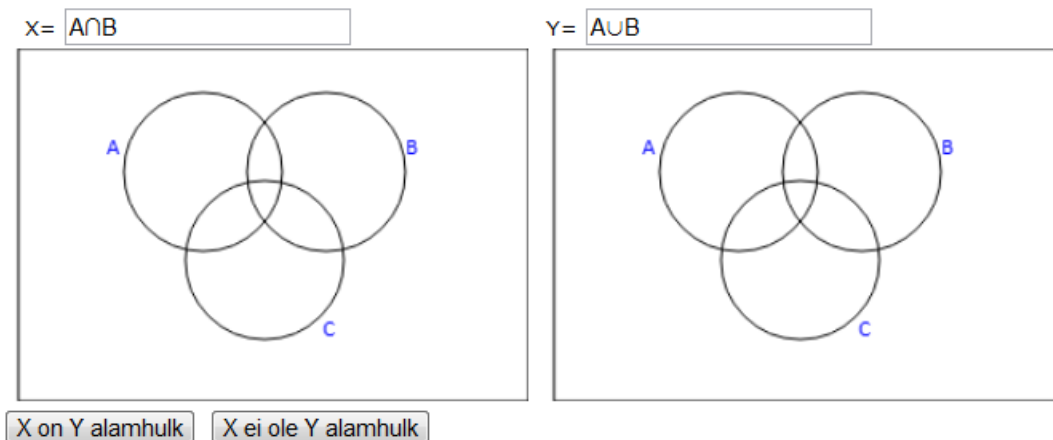
Järjekordselt saab "OK" nuppu vajutades jätkata olemasolevate joonistega töötamist eelnevast seisust. Korrektselt vastuse korral suunatakse kasutaja ülesande valimise lehele.

2.6. Alamhulgaks olemise kontroll

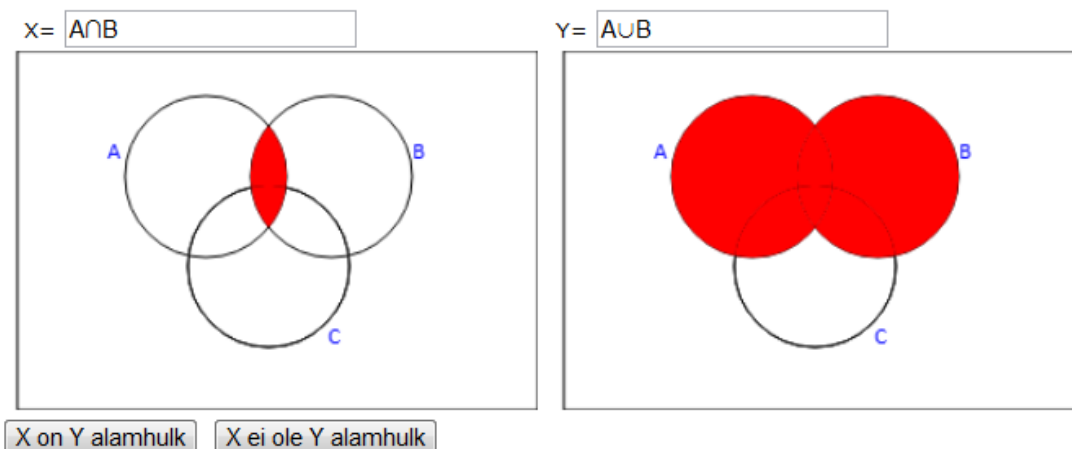
Alamhulgaks olemise kontroll on sarnane hulkade võrdsuse kontrollile. Tegemist on mitterange kuuluvusega, seega võivad hulgad olla ka võrdsed. Sarnaselt eelneva tüübiga kuvatakse kasutajale kaks avaldist ja joonist. Õpilane peab seejärel visandama õiged diagrammid ja nupule vajutades vastu võtma korrektse otsuse. Siin ülesandes on ka võimalik teha vastuse vigu. Dialoog on sarnane eelneva tüübiga, kui mõni joonistest on vale siis loetakse see samaväärsusveaks ja kuvatakse kasutajale sellekohane teade. Ülesande algseis on kujutatud joonisel 23 ja korrektsele vastusele vastav seis ja dialoogiaken Joonisel 24 ja 25.

Kontrollida, hulk X on hulk Y-i alamhulk ehk $X \subseteq Y$.

Selleks täita mõlema avaldise kohta Venni diagramm ja nende järgi võtta vastu otsus.



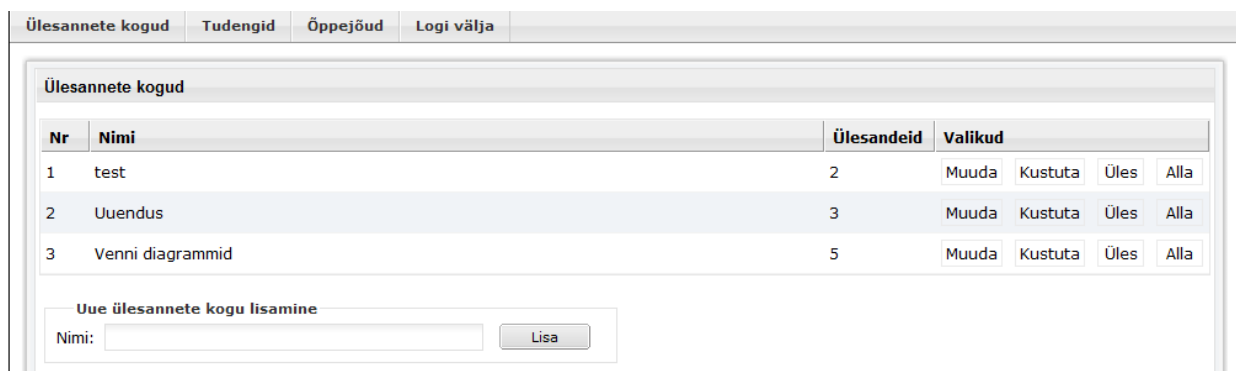
Joonis 23. Alamhulga ül. algseis



Joonis 24. Korrektsele täidetud diagrammid

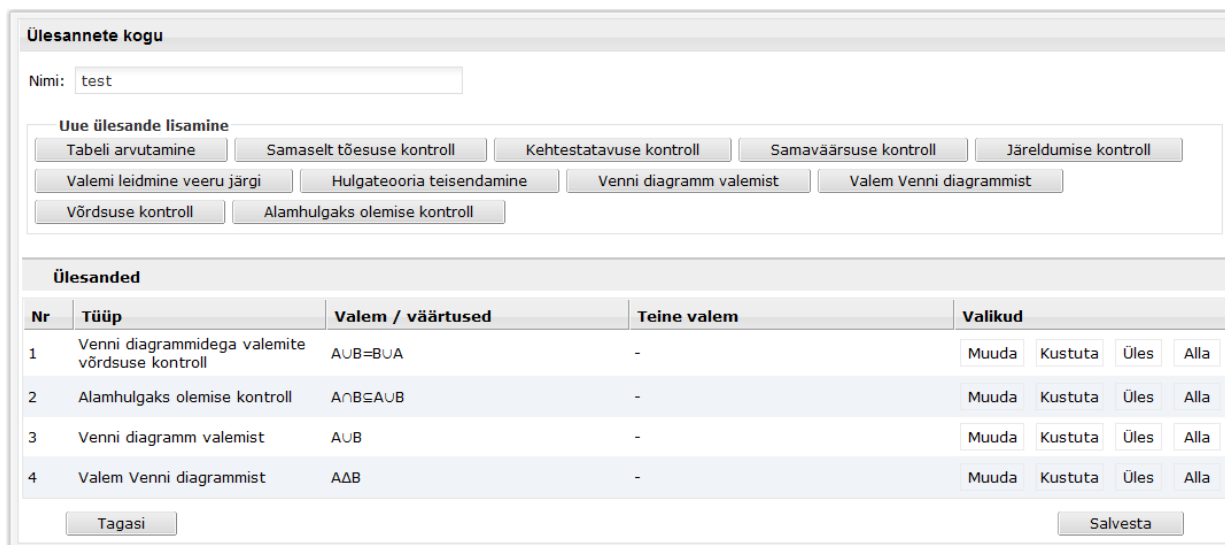
2.7. Ülesannete lisamine

Õppejõul on adminisatoorsete toimingute jaoks enda kaustajaliides. Seal saab ülesannetekogusid hallata, uusi ülesandeid lisada, õpilasi lisada ja õppejõude lisada. Sisse logides kuvatakse olemasolevad ülesannete kogud ja nupud nende muutmiseks või kustutamiseks.(Joonis 25).



Joonis 25. Ülesannetekogud

Nupule “Muuda“ vajutades kuvatakse ülesannete lisamise aken (Joonis 26).



Joonis 26. Ülesannete lisamise aken

Siin aknas on kuvatud iga ülesandetüübi kohta nupp selle loomiseks ja nimekiri juba olemasolevatest ülesannetest. Olemasolevate ülesannete muutmiseks on olemas nupud “Muuda” ja “Kustuta”. Vajutades mõnele ülesande loomise nupule kuvatakse sellele ülesandetüübile vastav aken. Üldiselt on kõik Venni diagrammi tüüpide lisamise aknad sarnased. Väike erinevus

on kahe põhitüübi ja kahe kontrollimisel põhineva tüübi vahel. Esimese kahe põhitüübi puhul tuleb sisestada ainult üks avaldis. Kontrolltüüpide korral tuleb kahe avalidse vahel kasutada, kas võrdusmärki või kuulumise märki. Näitan ära kummagi grupi kohta ühe näite. Kahe põhitüübi sisestamine on kujutatud joonisel 27 ja kontrolltüüpide sisestamine joonisel 28.

Uus ülesanne

Üldine

Ülesannete kogu: test

Ülesande tüüp: Valem Venni diagrammist

Ülesande tekst:

Valem

- 1 - 2 - 3 - 4 - 5 - 6

Lubatud vead	Punktid
Lubatud vigade arv kokku: <input type="text" value="2"/>	Maksimaalne punktide arv: <input type="text" value="2"/>
järjekorra vead: <input type="text" value="0"/>	järjekorra vea trahv: <input type="text" value="0"/>
Samaväärtus vead: <input type="text" value="2"/>	Samaväärtus vea trahv: <input type="text" value="0.2"/>
süntaksi vead: <input type="text" value="0"/>	süntaksi vea trahv: <input type="text" value="0"/>
vastamise vead: <input type="text" value="0"/>	vastamise vea trahv: <input type="text" value="0"/>

Joonis 27. Põhitüüpide lisamise aken

Uus ülesanne

Tagasi

Üldine

Ülesannete kogu: test

Ülesande tüüp: Alamhulgaks olemise kontroll

Ülesande tekst: Alamhulgaks olemise kontroll

Valem

$A \cap B \subseteq A \cup B$

' - 1 \cap - 2 \cup - 3 \setminus - 4 Δ - 5 \subseteq - 6

Lubatud vead

Lubatud vigade arv kokku: 3

järjekorra vead: 0

Samaväärtus vead: 2

süntaksi vead: 0

vastamise vead: 1

Punktid

Maksimaalne punktide arv: 2

järjekorra vea trahv: 0

Samaväärtus vea trahv: 0.2

süntaksi vea trahv: 0

vastamise vea trahv: 0.4

Joonis 28. Kontrolltüüpide lisamise aken

3. Realisatsioon

3.1. Kasutatud tehnoloogiad

3.1.1. Veebipõhine süsteem

Olemasolev süsteem on realiseeritud veebipõhisena seetõttu tuli ka juurde lisatav osa luua erinevaid veebitehnoloogiaid kasutades. Diagrammide joonistamiseks kasutatakse *HTML5 canvas* elementi. Kasutaja poolel on veel kasutusel scriptimiskeelt *JavaScript* ja selle teek *JQuery*. Olemasolev ülesannete lahendamise keskkond on programmeeritud kasutades serveripoolset scriptimiskeelt *PHP*. Seetõttu tuli Venni diagrammi programmi integreerimiseks ka seda kasutada.

3.1.2. HTML5

HTML on kirjelduskeel, mida kasutatakse veebi sisu struktureerimiseks ja näitamiseks. *HTML5* on keele uusim versioon, mille eesmärgiks on ühildada sellele eelnenud erinevaid standardeid, et tekiks ühtne ja kergesti mõistetav keel. Arenduse peamine eesmärk on olnud keele täiustamine uusima multimeedia toega samal ajal hoides seda inimestele hästi loetavana ja tagades platformide vahelise ühildumise. Lisaks tutvustati mitmeid uusi elemente, näiteks `<video>`, `<audio>` ja `<canvas>`. Loodud rakenduse kontekstis on eriti tähtis canvase element, sest seda kasutatakse diagrammide joonistamiseks.

3.1.3. Canvas

Canvas[5] on *HTML5* element, millega on võimalik 2d kujundite ja piltide dünaamiline ja skriptitav esitamine. *Canvast* tutvustati algselt Apple poolt aastal 2004, kui seda kasutati mitmetes toodetes, sealhulgas ka Safari veebibrauseris. Tasapisi jõudis see ka teistesse brauseritesse ja tänapäeval on canvase tugi olemas kõigis enimlevinud veebilehitsejates.

3.1.4. JavaScript

JavaScript[4] on Netscape'i poolt loodud objektorienteeritud programmeerimiskeel, mida kasutatakse peamiselt veebilehtede skriptimiseks. *JavaScripti* kasutatakse peamiselt kliendipoolel veebilehele interaktiivuse lisamiseks. On laialt levinud ja töötab kõikides brauserites.

3.1.5. *JQuery*

JQuery[6] on JavaScripti teek, mis lihtsustab dokumendil navigeerimist, DOM-elementide[7] valimist, animatsioonide loomist, sündmuste haldamist ja *AJAXi*[8] kasutamist. *JQuery* on laialt kasutuses tema lihtsuse ja loetavuse tõttu. Venni diagrammi programmis kasutati *JQueryt* ka selle tõttu, et ühtlustada programmi käitumist erinevates brauserites. Veebibrauserid käsitlevad teatud sündmusi erinevalt. Kasutades *JQuery* poolt realiseeritud meetodeid vähendatakse selle tõttu tekkivate vigade esinemist.

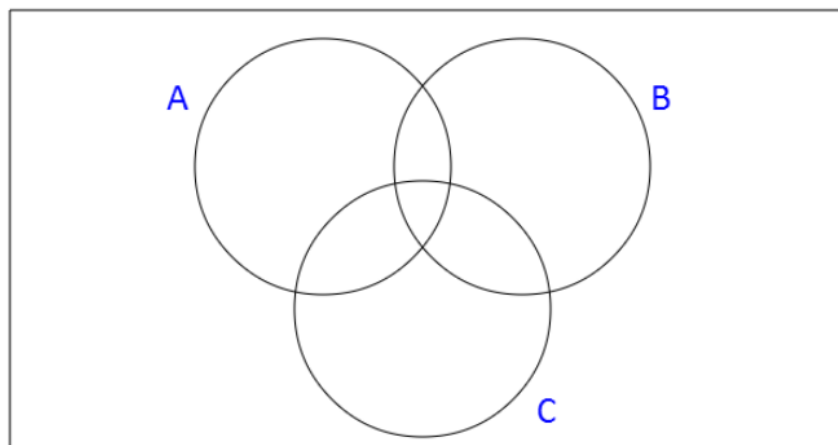
3.1.6. *PHP*

PHP[9] on laialdaselt kasutatav üldotstarbeline skriptimiskeel, mis on eriti sobiv veebirakenduste loomisel. Seda saab otse lisada *HTML* dokumentidesse. *PHP* tugevaks küljeks on see, et seda on kerge õppida ja, et see on laialt levinud. Kood interpreteeritakse veebiserveri poolt. *PHP* on tasuta ja töötab peaaegu kõigi operatsioonisüsteemi ja platvormi peal.

3.2. Realisatsioon

3.2.1. Diagramm

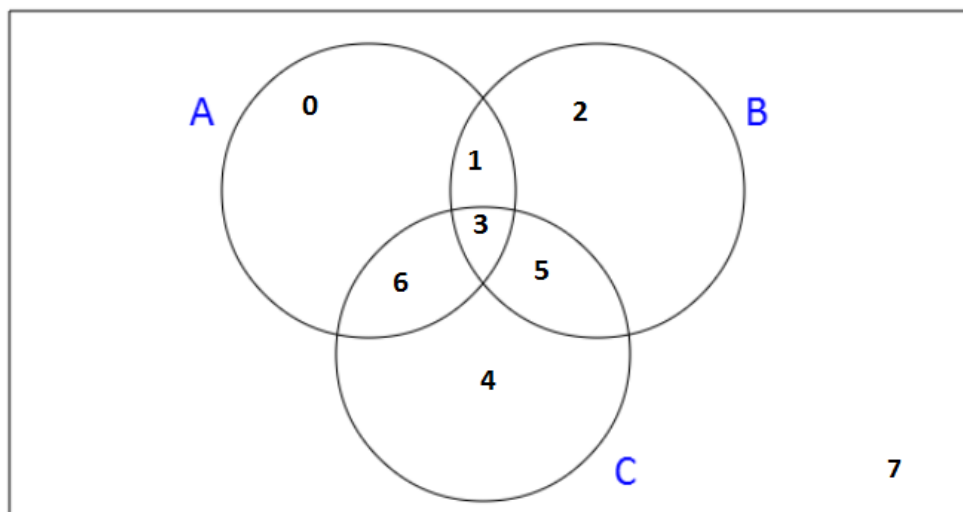
Rakendus võimaldab teha tehteid maksimaalselt 3 hulgaga. Hulkade tähisteks on A, B ja C. Lisaks on täiendit sisaldavate ülesannete jaoks toodud sisse universaalne hulk U, mida joonisel tähistatakse ristkülikuga. Kokku on diagrammil 8 võimalikku osa. Tühi diagramm on kujutatud joonisel 29.



Joonis 29. Tühi diagramm

Diagrammi seis hoitakse massiivis, kus iga 8 osa kohta on üks element. Elementidel võib olla väärtustena, kas “true” või “false” vastavalt sellele, kas elemendile vastav ala diagrammil peab olema värvitud või mitte. Algele diagrammile, kus ükski osa pole värvitud, vastab näiteks massiiv: `diagrammiSeis = [false,false,false,false,false,false,false,false]`

Igale osale vastab enda number ja kõikide kolme hulga korral on ära määratud, mis osad sinna kuuluvad. Lisaks on veel ka universaalne hulk U, kuhu kuuluvad kõik 8 elementi. Hulkadele vastavad massiivid: $A = [0,1,3,6]$, $B = [1,2,3,5]$, $C = [3,4,5,6]$, $U = [0,1,2,3,4,5,6,7]$. Joonisel 30 on ära näidatud iga osa numbriline vaste.



Joonis 30. Osad ja neid tähistavad numbrid

3.2.2. Hulkade leidmine

Programmil on realiseeritud meetodid iga tehte jaoks. Meetod võtab sissendiks kaks massiivi ja siis vastavalt tehte definitsioonile tagastab ühe massiivi. Näiteks antakse ühendi leidmise meetodile sisendiks hulkadele A ja B vastavad massiivid. Vastavalt definitsioonile tagastatakse massiiv, kuhu kuuluvad kõik sellised elemendid, mis kuuluvad vähemalt ühte hulkadest A ja B. Ühendi puhul oleks selleks siis hulk $A \cup B = [0,1,2,3,5,6]$.

Programmi jaoks on kõik tehted peale hulga täiendi on sama prioriteediga. Tehete järjekorra määravad sulud. Sulgude puudumisel käib lahendamine vasakult paremale. Täiend on kõige kõrgema prioriteediga ja see tehakse esimesena. Rakendus kontrollib kõigepealt, kas valemis on sulge. Kui on, siis leitakse kõige sisemine paar sulge. Seejärel kontrollitakse, kas alavalemis leidub täiend, kui leidub siis leitakse kõik täiendid. Kui kõik täiendid leitud arvutatakse vasakult paremale ülejäänud tehted. Lõpuks asendatakse järgi jäänud hulk esialgsesse valemisse ja hakatakse jälle otsast peale sulgude kontrollimisega. Täiendi leidmisel kasutatakse universaalset hulka U. Täiendi arvutamise meetod võtab sisendiks ühe massiivi. Tagastatav massiiv koosneb kõigist neist elementidest, mis kuuluvad U-sse, aga ei kuulu sisendiks antud massiivi.

3.2.3. Joonise osade värvimine

Hiirega ringi osal vajutades arvutatakse hiire vajutuse kordinaatide kaugused kõigist kolmest ringi keskpunktist ja selle kaudu määratakse ära, millise osa sees vajutust tehti. Iga vajutuse puhul asendatakse diagrammi seisuhoidvas massiivis osale vastav element väärtusega „true“. Joonise värvimisel saab seadistada muutuja „draw“ väärtuseks kas „true“ või „false“ vastavalt selle, kas tahetakse joonist tegelikult ka värvida või tahetakse ainult joonise seisule vastavat massiivi muuta. Väärtust „false“ kasutab programm ise vastuste kontrollimiseks.

3.2.4. Olemasolevasse süsteemi lisamine

Olemasolevasse süsteemi integreerimisega tekkis mõningaid probleeme. Tuli tekitada vajalikud uued ülesandetüübid ja nendega siduda diagrammi joonistamise funktsionaalsus. Probleeme tekkis tehetemärkide ühildamisega. Lisatav programm ja olemasolev süsteem said vahe märgist erinevalt aru ja see põhjustas vigu. Lisaks tuli uutele tüüpidele programmeerida punktide ja vigade arvestamine. Diagrammi joonistamise funktsionaalsust oli raske osade süsteemi funktsioonidega ühildada.

Kokkuvõte

Bakalaureusetöö peamiseks eesmärgiks oli olemasolevale veebis töötavale ülesannete süsteemile lisada uued ülesannete tüübid, mis põhinevad Venni diagrammidel. Töö raames lisati süsteemile juurde neli hulgateooria ülesandetüüpi: valemi leidmine diagrammi järgi, diagrammi leidmine valemi järgi, hulkade võrdsuse kontroll ja alamhulgaks olemise kontroll.

Bakalaureusetöös kirjeldati Diskreetse matemaatika elementide ainet, Venni diagrammidega seotud ülesandeid ning olemasolevat programmi. Näidati ka ära, miks arvutiga Venni diagrammide lahendamine efektiivsem ja mugavam on, kui paberi peal selle tegemine. Selleks, et Venni diagrammide joonistamine oleks mugav ja kiire tehti joonise täitmine väga lihtsaks. Õpilastele joonistatakse ette tühi diagramm ja hiirega vajutades saab selle pealt värvida ära vajalikke osi. Ühe nupuvajutusega saab vastuse esitada. Kohene tagasiside ja vigadest teavitamine aitavad õpilasel materjali paremini omandada.

Uute tüüpide implementeerimisel püüti järgida juba olemasoleva süsteemi funktsionaalsust. Uusi ülesandeid saab lisada sarnaselt vanade tüüpidega. Sarnaselt realiseeriti ka kogu ülejäänud funktsionaalsus. Ka kasutajaliidese disaini poolelt järgiti enamjaolt olemasolevat kujundust. Ühtne kujundus ja funktsionaalsus lihtsustavad programmi kasutamist ja vähendavad süsteemi kasutamaõppimisele kuluvat aega.

Venn diagramm exercise types for Discrete Mathematics web environment

Bachelor's thesis

Marten Hennoch

Summary

MTAT.05.110 Elements of Discrete Mathematics is a course taught in Tartu University to first year computer science and mathematics students. The course is divided roughly into two parts: propositional calculus and set theory. Besides normal practicals there are computer practicals in which learning programs are used. At the moment there are no programs for set theory in use.

One of the programs in use is a web application for solving truth table exercises. The aim of this bachelor's thesis is to extend the existing application with 4 new exercise types based on Venn diagrams to that application. The types are: finding the formula from a given diagram, finding the diagram from a given formula, determining the equality of sets and subset control.

Most of the application's functionality remains the same for new exercise types. The teacher can add new exercises the same way he/she did for previously existing types. Being able to manage all the different exercise types in the same place is also a great convenience and saves time.

Drawing Venn diagrams is made simple. Program draws an empty figure for the user and he/she can just click on the parts that he/she wants to highlight. The application checks the student's answer and gives instant feedback. This saves a lot of time for the teachers as there are many students taking this course. Application also records all mistakes made. The program also saves all student's attempts. The teacher can later check that data and for example see which exercise type causes problems for students.

Viited

Kõik viidatud veebileed on vaadatud seisuga 07.05.2012

[1] "Venn diagram - Wikipedia, the free encyclopedia"

http://en.wikipedia.org/wiki/Venn_diagram

[2] Alari Lukk, Kaasaegse kasutajaliidesega veebikeskkond tõeväärtustabeli ülesannete lahendamiseks, bakalaureusetöö, 2010.

[3] Peeter Oja, Hulgateooria, Tartu, 2006.

[4] "JavaScript - MDN"

<https://developer.mozilla.org/en/JavaScript>

[5] "Canvas element - Wikipedia, the free encyclopedia"

http://en.wikipedia.org/wiki/Canvas_element

[6] "jQuery: The Write Less, Do More, JavaScript Library"

<http://jquery.com/>

[7] "W3C Document Object Model"

<http://www.w3.org/DOM/>

[8] "Ajax (programming) - Wikipedia, the free encyclopedia"

[http://en.wikipedia.org/wiki/Ajax_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))

[9] "PHP: Hypertext Preprocessor"

<http://www.php.net/>

[10] M.Kilp, U. Nummert. Hulgateooria elemendid Tartu 1994

Lisad

Lisa 1: CD plaat koos programmi lähtekoodi ja paigaldamisõpetusega.