



**K A U N O  
TECHNOLOGIJOS  
UNIVERSITETAS**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS  
INFORMATIKOS FAKULTETAS  
SISTEMŲ ANALIZĖS KATEDRA**

Vilma Jakubauskienė

**NETIESINIŲ LYGČIŲ SPRENDIMO  
KOMPIUTERINĖMIS PROGRAMOMIS TAIKYMAS  
MOKYMO PROCESSE**

Magistro darbas

**Recenzentas  
lekt. dr. Ingrida Lagzdinytė  
2008-05-19**

**Vadovas  
doc. dr. Kostas Plukas  
2008-05-19**

**Atliko  
IFT 6 gr. stud.  
V. Jakubauskienė  
2008-05-19**

**KAUNAS, 2008**

# NONLINEAR EQUATIONS SOLVING USING COMPUTER-AIDED PROGRAMS IN EDUCATIONAL PROCESS

## SUMMARY

The most practical, scientific and various applied tasks are solved by using numerical methods. These methods are urgent in teaching aspect, too. The students of various higher schools are solving practical tasks when studying theory of numerical methods, therefore they have to programme algorithms by themselves or use computer programs.

In this work there are analyzed universal computer algebra systems – Matlab and Mathcad – as the one of the most important measures of solving tasks by using numerical methods. Also there is analyzed the use of Excel program for the reason how to solve the nonlinear equation by using numerical methods. In the work there are analyzed methods of nonlinear equations solving, such as simple iterative, bisection, Newton and secant methods and their realization in each referred program..

There are described the very methods when presenting every those themes and compared the possibilities of each computer program for the reason how to solve the tasks by using these methods. There is given description of methods functions, application and examples of solving.

In the work there is presented created computer training appliance for the reason of mastering with nonlinear equation solving by using numerical methods.

## TURINYS

<b>1. ĮVADAS .....</b>	<b>6</b>
<b>2. PROBLEMOS PAGRINDIMAS .....</b>	<b>9</b>
2.1. STUDENTŲ PASIRENGIMO SKAITINIŲ METODŲ STUDIJOMS ANALIZĖ .....	9
2.2. SKAITINIŲ METODŲ MOKYMO ANALIZĖ .....	13
2.3. PROBLEMOS, SUSIJUSIOS SU NETIESINIŲ LYGČIŲ SPRENDIMU: .....	14
2.4. INTERNETO WWW SKAITINIŲ METODŲ SVETAINĖS .....	15
<b>3. KOMPIUTERINIŲ MOKOMŲJŲ PROGRAMŲ APŽVALGA .....</b>	<b>16</b>
<b>4. NETIESINIŲ LYGČIŲ SPRENDIMO METODAI KOMPIUTERINĖSE PROGRAMOSE...</b>	<b>18</b>
4.1. ŠAKNŲ LOKALIZACIJA .....	18
4.2. PUSIAUKIRTOS METODAS .....	21
4.3. PAPRASTŲJŲ ITERACIJŲ METODAS.....	24
4.4. NIUTONO METODAS.....	29
4.5. KIRSTINIŲ METODAS .....	31
4.6. UŽDAVINIŲ SPRENDIMO KOMPIUTERINĖMIS PROGRAMOMIS PALYGINIMAS .....	33
<b>5. MOKOMOSIOS KOMPIUTERINĖS PRIEMONĖS PROJEKTAVIMAS .....</b>	<b>35</b>
5.1. KOMPIUTERINIŲ PRIEMONIŲ VERTINIMAS .....	35
5.2. KOMPIUTERINĖ MOKOMOSIOS PRIEMONĖS PASKIRTIS .....	38
5.3. REIKALAVIMAI APARATINEI IR PROGRAMINEI ĮRANGAI .....	40
5.4. TECHNOLOGIJŲ KOMPIUTERINEI PRIEMONEI PARINKIMAS .....	40
5.5. KOMPIUTERINĖS PRIEMONĖS STRUKTŪRINĖ IR HIPERYŠIŲ SCHEMA.....	42
5.6. KOMPIUTERIZUOTA TEORINĖ MEDŽIAGA .....	43
5.7. TESTAI .....	45
5.8. SAVIKONTROLĖS UŽDUOTYS .....	46
5.9. KONTROLINIS TESTAS .....	50
5.10. INSTRUKCIJA SVETAINĖS NAUDOTOJUI .....	51
<b>6. STUDENTŲ KOMPIUTERINĖS PRIEMONĖS VERTINIMAS.....</b>	<b>53</b>
<b>IŠVADOS .....</b>	<b>55</b>
<b>LITERATŪRA .....</b>	<b>56</b>
<b>1 PRIEDAS. ANKETA .....</b>	<b>58</b>
<b>2 PRIEDAS.DELPHI PROGRAMOS KODAS.....</b>	<b>58</b>
<b>3 PRIEDAS. UŽDUOTYS .....</b>	<b>69</b>
<b>4 PRIEDAS. RUOŠINIAI LYGČIŲ SPRENDIMUI MATLAB APLINKOJE .....</b>	<b>69</b>
<b>5 PRIEDAS. DARBO MATLAB APLINKOJE DEMONSTRACINIS FILMUKAS .....</b>	<b>70</b>

## PAVEIKSLŲ IR LENTELIŲ SĄRAŠAS

### Paveikslai

1 pav. Matematikos mokyklinio egzamino balo pasiskirstymas .....	9
2 pav. Matematikos valstybinio egzamino balo pasiskirstymas .....	10
3 pav. Moksleivių matematikos egzamino laikymo A ir B lygiu pasiskirstymas.....	10
4 pav. Matematikos egzamino vidurkio pasiskirstymas .....	11
5 pav. Studentų pasirengimo dirbti kompiuterinėmis programomis pasiskirstymas .....	11
6 pav. Darbo su skirtingomis kompiuterinėmis programomis pasiskirstymas .....	12
7 pav. Sprendimo pusiaukirtos metodu grafas.....	13
8 pav. Sprendimo papr. iteracijų metodu grafas .....	13
9 pav. Sprendimo Niutono metodu grafas .....	14
10 pav. Sprendimo kirstinių metodu grafas .....	14
11 pav. Šaknų izoliacijos intervalų nustatymas Excel programoje .....	19
12 pav. Šaknų izoliacijos intervalų nustatymas Mathcad programoje.....	19
13 pav. Šaknų izoliacijos intervalų nustatymas Matlab programoje .....	21
14 pav. Funkcijų grafikų susikirtimo intervalų nustatymas.....	22
15 pav. Funkcijos grafikų susikirtimo taško nustatymo intervalas.....	25
16 pav. Funkcijos grafikų susikirtimo taško nustatymo intervalas.....	26
17 pav. Funkcijos grafikas grafiniame Matlab lange.....	29
18 pav. Grafinis sprendimas Niutono metodu .....	29
19 pav. Grafinis sprendimas kirstinių metodu .....	32
20 pav. Mokymui naudojamų kompiuterinių programų klasifikacija .....	35
21 pav. Svetainės vartotojų grupės .....	39
22 pav. Pagrindinio puslapio struktūrinė schema .....	42
23 pav. Puslapio “Metodai” struktūrinė schema.....	42
24 pav. Puslapio “Savikontrolė” struktūrinė schema.....	43
25 pav. Pradinis svetainės langas.....	43
26 pav. Metodų aprašymo langas .....	44
27 pav. Programos kūrimo langas.....	44
28 pav. Animuotas vaizdas .....	45
29 pav. Testų atlikimo langas .....	45
30 pav. Savikontrolės užduočių schema(1) .....	46
31 pav. Savikontrolės užduočių schema(2) .....	47
32 pav. 1 užduotis .....	47
33 pav. 2 užduotis .....	48
34 pav. 3 užduotis .....	48
35 pav. 4 užduotis .....	49
36 pav. 5 užduotis .....	49
37 pav. Pradinis testo langas.....	50
38 pav. Testo vertinimo langas.....	50
39 pav. Pradinis kontrolinės užduoties langas .....	51
40 pav. Testo pasirinkimo langas .....	51
41 pav. Klaidų pranešimo langas.....	51
42 pav. Failo pasirinkimas .....	52
43 lenelė. Studentų vertinimo rezultatai .....	53
44 pav. Studentų vertinimo procentinis pasiskirstymas.....	54

## Lentelės

1 lentelė. Problemos, susijusios su lygčių sprendimu.....	14
2 lentelė. Svetainės .....	15
3 lentelė. Pusiaukirtos metodo realizacija MS Excel pagalba .....	23
4 lentelė. Sprendimas papr. it. metodu MS Excel pagalba .....	27
5 lentelė. Sprendimas Niutono metodu MS Excel programos pagalba .....	30
6 lentelė. Sprendimas kirstinių metodu MS Excel programos pagalba .....	32
7 lentelė. Sprendimų pusiaukirtos metodu kompiuterinėmis programomis palyginimas..	33
8 lentelė. Sprendimų papr. iteracijų metodu kompiuterinėmis programomis palyginimas	34
9 lentelė. Sprendimų Niutono metodu kompiuterinėmis programomis palyginimas .....	34
10 lentelė. Sprendimų Kirstinių metodu kompiuterinėmis programomis palyginimas .....	34

## 1. IVADAS

Įvairiose mokslo, technikos ir gamybos srityse sutinkame matematinius uždavinius, kurių negalime tiksliai išspręsti analitiniais metodais. Todėl įprasta kalbėti apie atskirą matematikos sritį, kuri vadinama skaičiavimo matematika. Joje nagrinėjama, kaip įvairių klasikinės matematikos šakų uždavinius spręsti skaitiniais metodais. Kitaip tariant, skaičiavimo matematikoje nagrinėjama, kokiais metodais ir kaip reikia spręsti klasikinės matematikos uždavinius, norint gauti paprasčiausius skaitinius ar apytikslius jų sprendinius. Daugelis šių sprendimo metodų yra apytiksliai, t.y. jais galima rasti ne tikslų, o apytikslų uždavinio sprendinį.

Skaičiavimo matematikoje yra aiškus tikslas: sistemingai išnagrinėti pagrindinių matematinės analizės ir algebros uždavinių skaitinius sprendimo metodus. Štai šie uždaviniai: netiesinių lygčių ir jų sistemų sprendimas, interpoliavimas, aproksimavimas, diferencijavimas, integravimas., diferencialinių bei integralinių lygčių sprendimas. Visi metodai šiems uždaviniams apytiksliai spręsti paprastai vadinami apytiksliais skaičiavimo metodais, arba tiesiog skaičiavimo metodais.

Norint išspręsti tam tikrus uždavinius skaitiniais metodais, reikia daug laiko ir darbo sąnaudų reikalaujančio programavimo. Tam buvo sukurtos kompiuterinės matematinės sistemos: Matlab, Mathcad, Maple, Mathematica. Skaitiniai metodai yra aktualūs, nes dauguma praktinių ir mokslinių uždavinių yra sprendžiami skaitiniais metodais. Šie metodai aktualūs ir mokymo aspektu. Įvairiuose universitetuose studentai, studijuodami skaitinių metodų teoriją, sprendžia praktines užduotis, taigi tenka arba patiems programuodami uždavinių sprendimo algoritmus, arba naudoti kompiuterines matematikos sistemas.

Šiame darbe nagrinėjami universalių kompiuterinių matematinių sistemų – Matlab ir Mathcad, kaip vieno svarbiausių skaitiniais metodais sprendžiamų uždavinių priemonės. Taip pat nagrinėjamas MS Excel programos panaudojimas spręsti netiesines lygtis skaitiniais metodais. Darbe išnagrinėti netiesinių lygčių sprendimo metodai, tokie kaip paprastųjų iteracijų, pusiaukirtos, Niutono ir kirstinių metodai, jų realizacija kiekvienoje iš nurodytų programų. Pateikiant kiekvieną iš šių temų yra trumpai aprašyti patys metodai, lyginamos kiekvienos kompiuterinės programos galimybės spręsti uždavinius pagal šiuos metodus, jų funkcijų aprašymas, taikymai ir sprendimo pavyzdžiai. Nėra žinoma darbų lietuvių kalba, kur būtų aprašyti ar palyginti šių programų skaitiniai metodai, todėl manau, kad šis darbas bus naudingas praktinių uždavinių sprendimui.

## **Problemos analizė**

Dėstant netiesinių lygčių ir jų sistemų sprendimo metodus Klaipėdos verslo ir technologijų kolegijoje(KVTK), susiduriama su problema: studentams matematika dėstoma pirmo kurso pirmame pusmetyje. Todėl pradėjus skaitinių metodų kursą antrame kurse, jiems trūksta žinių ir įgūdžių pakankamai gerai suprasti ir įsisavinti netiesinių lygčių sprendimo metodus. Skaitinių metodų praktiniams darbams atlikti, t.y. spręsti lygtis skaitiniais metodais buvo naudojama MS Excel programa, nors ši programa nėra ta priemone, su kuria būtų galima spręsti lygtis ir gauti pakankamai tikslius sprendinius. Įsigijus MatLab programos licenziją, keičiasi praktinių darbų atlikimo galimybės. Šiame darbe atliekamas MS Excel, Matlab ir Mathcad programų, sprendžiant tuos pačius uždavinius, sulyginimas. Bandoma išsiaiškinti, kokią geriausiai pasirinkti programą uždaviniams spręsti skaitiniais metodais, kokie šių programų trūkumai ir privalumai.

Paruošta mokymo priemonė būtų skirta Klaipėdos verslo ir technologijų kolegijos besimokantiems studentams.

Studijos Kolegijoje vykdomos dienine ir neakivaizdine forma, planuojama pradėti vykdyti vakarines studijas. Parengtos ir įgyvendintos išlyginamųjų studijų programos aukštesniašias studijas baigusiems absolventams.

Siekiant optimizuoti studijų procesą yra koreguojamos studijų programos - mažinamas kontaktinių ir didinamas studentų savarankiško darbo valandų kiekis.

Dabartinis aukštojo neuniversitetinio mokymo tikslas – ne siaura profesinė specializacija, bet savarankiškos, nuolat besimokančios asmenybės, turinčios konkurencingą kvalifikaciją ugdymas. Baigę kolegiją, studentai turi gebėti pademonstruoti bazines žinias, susijusias su atitinkamu studijų dalyku, kontekste bei fundamentalių žinių mokėjimus, Jie turi gebėti rinkti, apibūdinti ir interpretuoti informaciją studijuojamo dalyko kontekste. Siekiant šio tikslo studentai turi daug dirbti savarankiškai.

Pagal naują aukštųjų neuniversitetinių studijų programą net 35% akademinų valandų skiriama savarankiškam besimokančiųjų darbui. Šiandien naujausios informacinės technologijos suteikia studentams galimybes studijuoti savarankiškai, būti nepriklausomais nuo dėstytojų. Savarankiškame darbe labai svarbi aktyvi studentų veikla.

Savarankiško darbo metu pedagogas organizuoja mokymosi procesą. Pateikia mokymosi tikslus, mokymo medžiagą, o besimokantysis sudaro savo mokymosi ir išmokimo sistemą.

## **Darbo tikslas**

Išanalizuoti kokiomis programinėmis priemonėmis galima spręsti netiesines lygtis skaitiniais metodais. Pasirinkti palankiausią kompiuterinę priemonę netiesinių lygčių sprendimui.

## **Uždaviniai**

- Atlikti netiesinių lygčių sprendimo skaitiniais metodais analizę.
- Išnagrinėti ir aprašyti, kaip šie metodai realizuoti MatLab, Mathcad ir MS Excel programose.
- Palyginti sprendimo šiose programose galimybes pagal jų sudėtingumą, patogumą vartotojui ir funkcijų įvairovę.
- Sukurti mokomąją priemonę, skirtą spręsti netiesines lygtis skaitiniais metodais.
- Atlikti eksperimentinį priemonės testavimą.

## **Tyrimo metodai**

- Anketinė apklausa
- Rezultatų analizė
- Literatūros apžvalga
- Mokomųjų kompiuterinių programų analizė



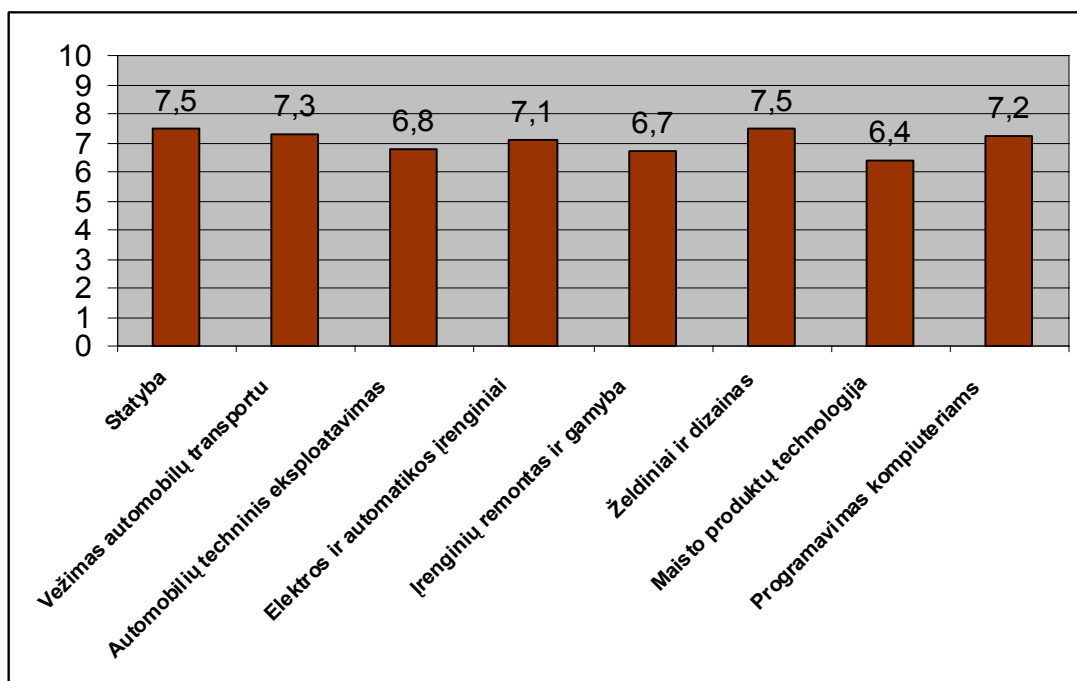
## 2. PROBLEMOS PAGRINDIMAS

### 2.1. Studentų pasirengimo skaitinių metodų studijoms analizė

Skaitinius metodus nagrinėja skaičiuojamoji matematika, kuri yra matematikos dalis. Todėl nuo bendro aukštosios matematikos kurso įsisavinimo priklauso ir skaitinių metodų kurso sėkmingos studijos.

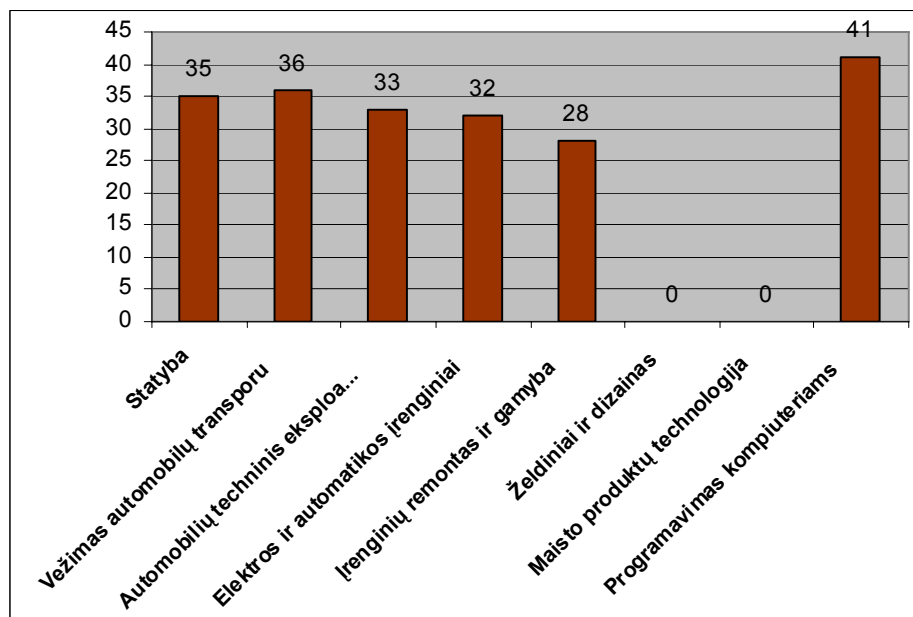
Daugumoje kolegijoje manoma, kad, kad tie abiturientai, kurie laikė valstybinį matematikos egzaminą, sėkmingiau tęsia studijas kolegijose, nei tie, kurie laikė mokyklinį matematikos egzaminą.

Stojant į kolegiją nepriklausomai nuo to, ar buvo laikytas mokyklinis ar valstybinis egzaminas, balai skaičiuojami vienoda tvarka, t.y. egzamino pažymys dauginamas iš svertinio koeficiento. Matematika turi didžiausią iš visų taikomų svertinių koeficientų t. y. 0,5. Atlikus stojančiųjų turėtus matematikos dalyko įvertinimų analizę, taip pat atsižvelgus į tai, ar tai buvo mokyklinis, ar valstybinis egzaminas, galima teigti, kad bendras stojančiųjų matematikos vidurkis yra 7,06. Klaipėdos technologijų fakultetas teikia išsilavinimą pagal 8 studijų programas. Stojančiųjų matematikos vidurkis tarp įvairių specialybių pasiskirstė tokia tvarka:



1 pav. Matematikos mokyklinio egzamino balo pasiskirstymas

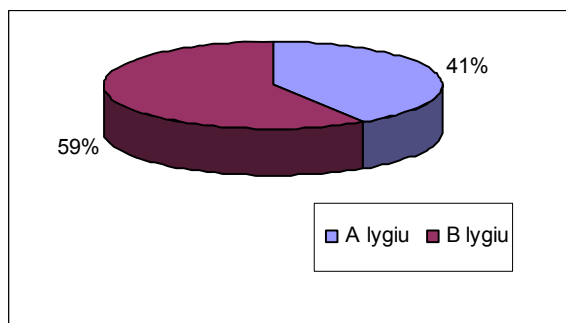
Nemaža dalis abiturientų valstybinio matematikos egzamino nelaikė.



2 pav. Matematikos valstybinio egzamino balo pasiskirstymas

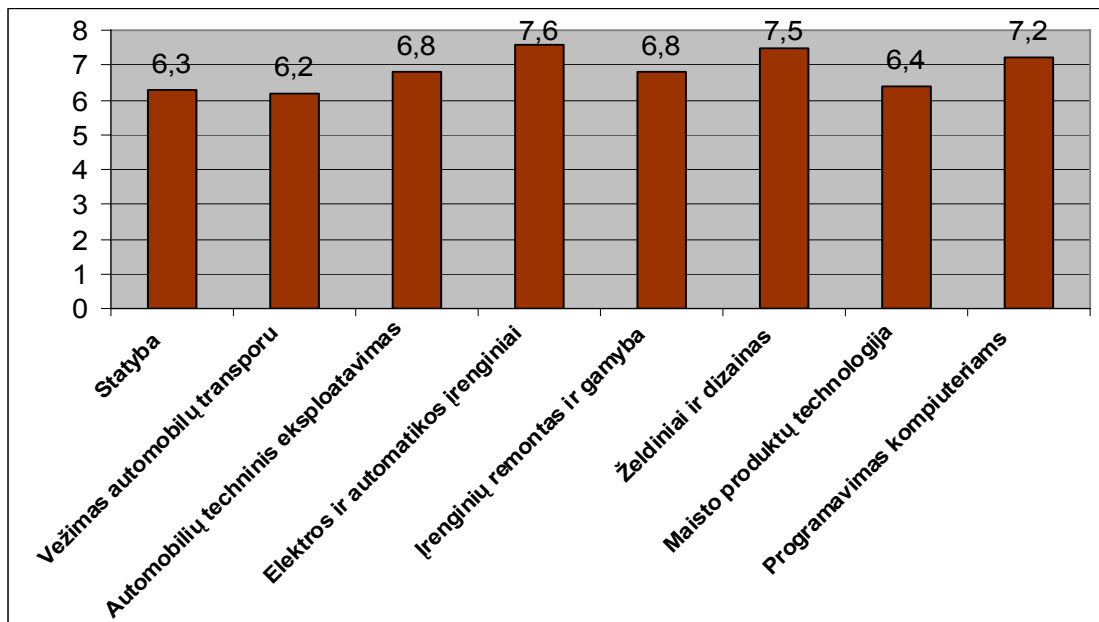
Kolegijoje nepakankami įstojusiųjų studijuoti bendrieji gebėjimai. Studentai įstoja iš įvairių mokyklų. Bendras pasiruošimo lygis nevienodas. Tai be abejo priklauso ir nuo motyvacijos, nuo mokytojų paruošimo, nuo to, ar studentas baigė mokyklą mieste ar rajone.[7]. Matomai gali būti dėl to, kad moksleiviai, besimokantys miesto tipo vidurinėse mokyklose, mokydamiesi labiau linkę naudotis korepetitoriaus paslaugomis, o tai sumažina galimybę įprasti dirbti savarankiškai, mažiau skatina loginį mąstymą, kas vėliau atsiliepia studijoms.

Skirtingi mokinių mokymosi lygiai įtakoja labai įvairią įstojusiųjų į kolegiją mokyklinių žinių kokybę. Atlikus stojančiųjų duomenų analizę, paaiškėjo, kad matematikos egzaminą laikusiųjų A lygiu yra tik maža dalis, t.y. 41%.



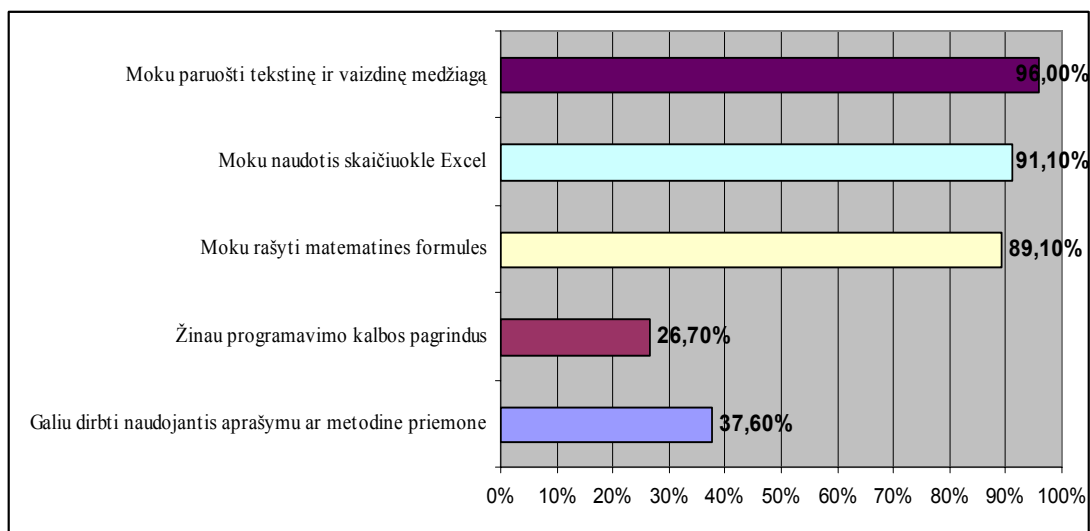
3 pav. Moksleivių matematikos egzamino laikymo A ir B lygiu pasiskirstymas

Aukštosios matematikos egzamino rezultatai prastesni, nei mokykloje laikyto egzamino rezultatai:



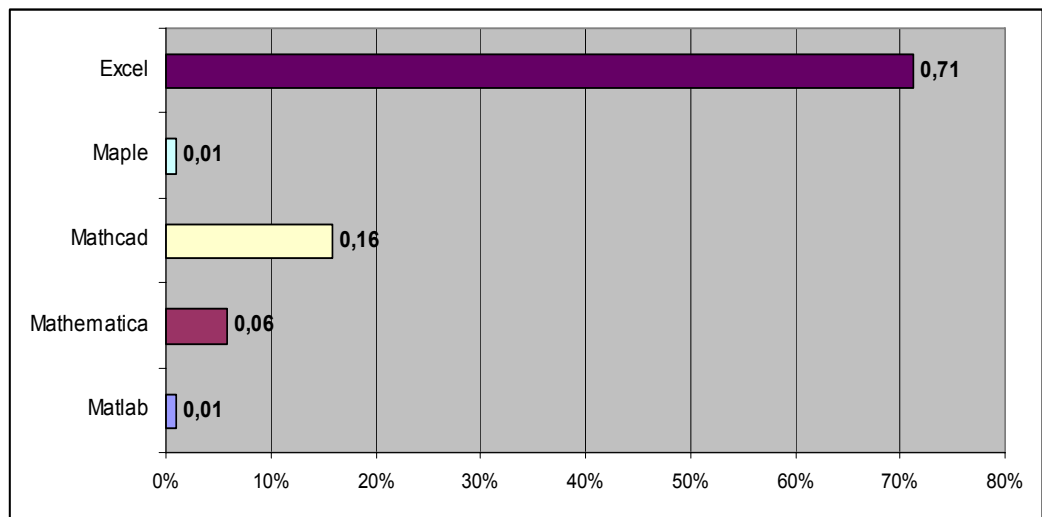
4 pav. Matematikos egzamino vidurkio pasiskirstymas

Tiriant studentų pasirengimą ir norą taikyti kompiuterines programas skaitinių metodų dalyko studijoms buvo apklausti 84 studentai. Tyrimo rezultatai rodo, kad studentai moka dirbti ir naudotis skaičiuokle MS Excel, bet nepakankamai žino programavimo pagrindus.



5 pav. Studentų pasirengimo dirbti kompiuterinėmis programomis pasiskirstymas

Studentų atsakymų į klausimus „Kokias kompiuterines programas, skirtas matematikai žinote“, „Ar matematikos studijoms, mokymuisi teko naudotis (ar esate naudoję) taikomąją kompiuterinę programą“, rezultatų analizė rodo, kad 68,2% tiriamųjų teigia naudojantys kompiuterines programas, tačiau labiausiai taikoma yra skaičiuoklė MS Excel, o specializuoti paketai, skirti matematikai žinomi ir taikomi mažai. Darbą su MS Excel programa studentai vertina pakankamai gerai. Dirbant ja, nereikia programavimo įgūdžių. Vidurinėje mokykloje programavimas nėra privalomas. Kai kurie respondentai yra baigę mokyklą anksčiau, kada iš viso nebuvo informatikos dalyko.

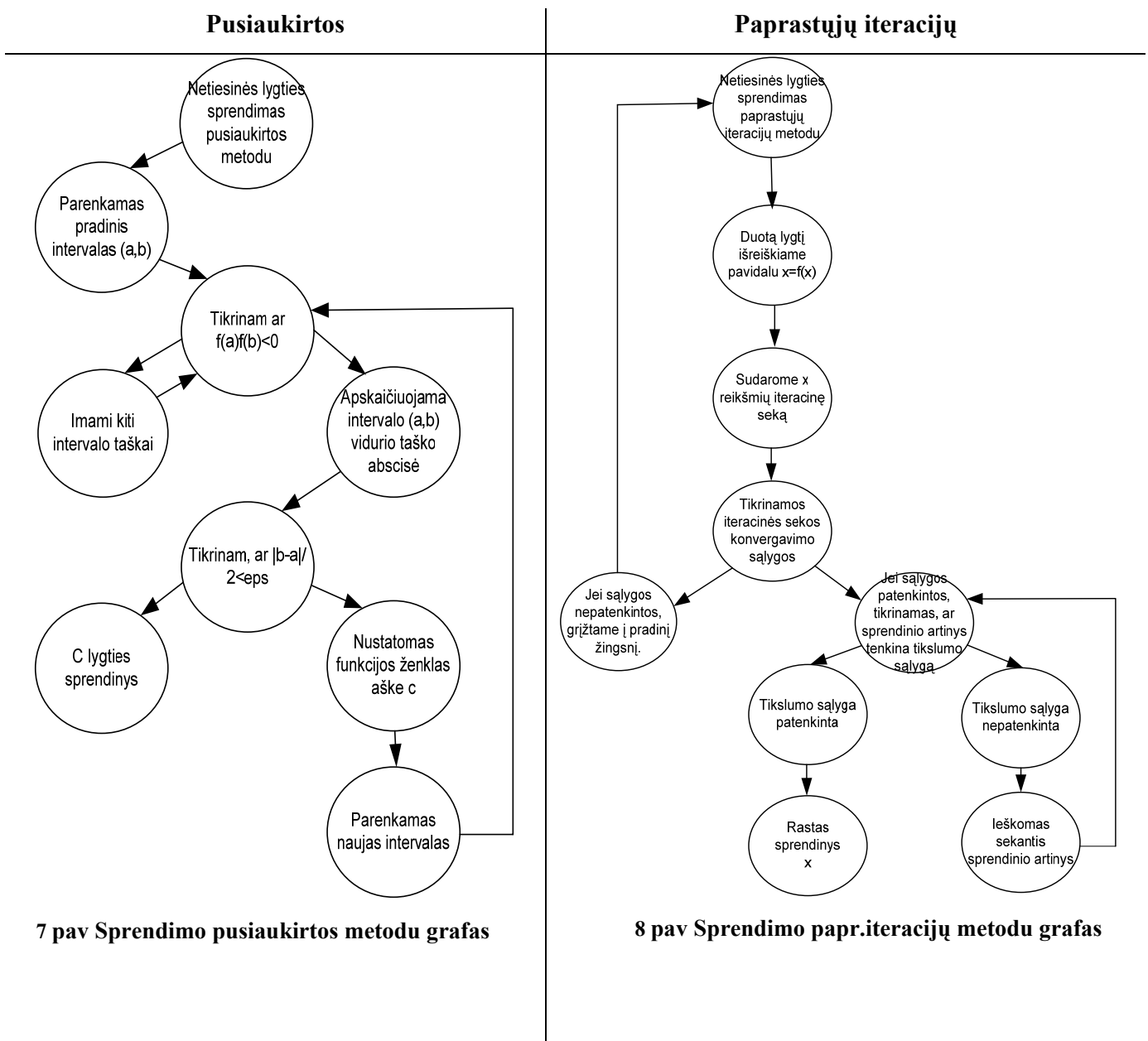


6 pav. Darbo su skirtingomis kompiuterinėmis programomis pasiskirstymas

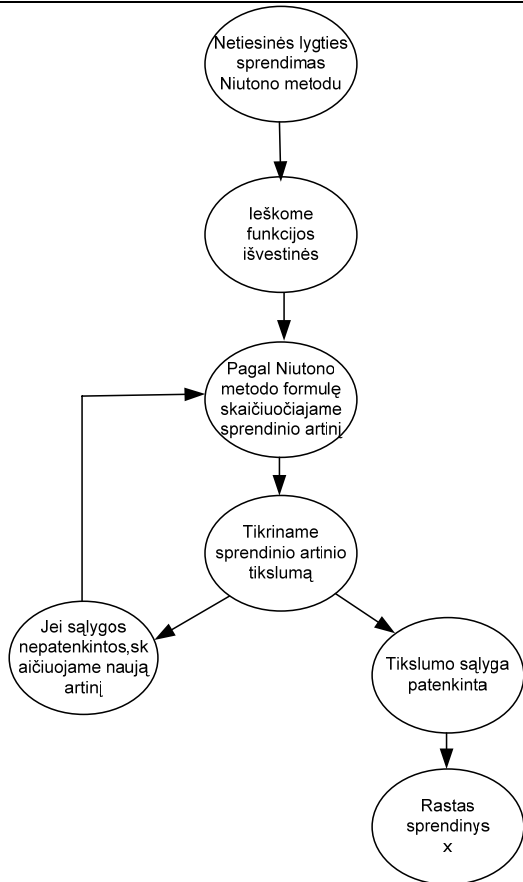
## 2.2. Skaitinių metodų mokymo analizė

Skaitinių metodų modulio temos pavaizduotos Protégé 3.2.1 įrankiu sukurtoje TGVizTab kortelėje

Iš antrus metus dėstomo „Skaitiniai metodai“ modulio praktikos, galima teigti, kad studentams sunkiau sekasi skaitiniais metodais spręsti netiesines lygtis. Netiesinių lygčių sprendimo metodams priklauso šie pagrindiniai metodai:

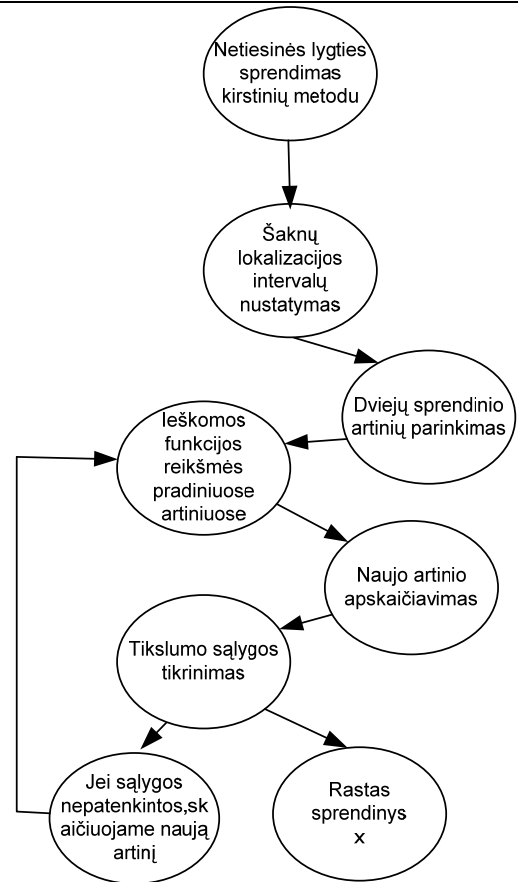


### Niutono



9 pav Sprendimo Niutono metodu grafas

### Kirstinių



10 pav Sprendimo kirstinių metodu grafas

### 2.3. Problemos, susijusios su netiesinių lygčių sprendimu:

1 lentelė. Problemos, susijusios su lygčių sprendimu

<b>Pusiaukirtos</b>	Pradinė lygtis skaidoma į dvi lygtis. Norint rasti lygties sprendinį, pirmiausia braižomi šių funkcijų grafikai. Nustatomas intervalas, kuriame yra sprendinys. Nustatę nors vieną sprendinio buvimo intervalą, dažnai nepatikrina kitų galimų intervalų, kuriuose gali būti sprendiniai. <i>Dažnai neatsižvelgiama į sąlygą <math>f(a) \cdot f(b) &lt; 0</math></i>
<b>Paprastųjų iteracijų</b>	Sprendžiant šiuo metodu lygtis turi būti perrašoma tokiu pavidalu, kad pasirinkus pirmąjį artinį, sekantys sprendinio artiniai artėtų prie norimo gauti tikslumo sprendinio, t.y. artinių seka konverguotų. Studentams ne visada pavyksta teisingai perrašyti duotą lygtį ir dėl to ieškant sprendinio, artiniai tolsta nuo sprendinio
<b>Niutono</b>	Šio metodo algoritmo esmė ta, kad pagal artinio radimo formulę, metodas netiesinių lygčių sprendimui gali būti taikoma tik tada, jei funkcijos išvestinė yra baigtinė ir nelygi nuliui. Studentai susiduria su sudėtinių funkcijų išvestinių ieškojimu.

<b>Kirstinių</b>	Taikant kirstinių metodą, pirmiausia būtina parinkti du sprendinių artinius. Sunkiau sekasi parinkti pradinius artinius.
------------------	--

## 2.4. Interneto WWW skaitinių metodų svetainės

2 lentelė. Svetainės

Skaitiniai metodai	
<a href="http://www.exponenta.ru">www.exponenta.ru</a>	
Matematikos svetainė. Įvadas į skaitinius metodus.	Rusų kalba
<a href="http://www.mathsoft.com">www.mathsoft.com</a>	
Sistemos Mathcad gamintojo svetainė.	Anglų kalba
<a href="http://www.mathworks.com">www.mathworks.com</a>	
Sistemos Matlab gamintojo svetainė.	Anglų kalba
<a href="http://mathworld.wolfram.com">http://mathworld.wolfram.com</a>	
Matematikos svetainė, kurioje pateikiamas platus matematinių temų pasirinkimas, duoti uždavinių sprendimų pavyzdžiai.	Anglų kalba
<a href="http://aleph0.clarku.edu/~djoyce/newton/newton.html">http://aleph0.clarku.edu/~djoyce/newton/newton.html</a>	
Pateikiamas Niutono metodo taikymas sudėtingoms netiesinėms lygtims spręsti.	Anglų kalba
<a href="http://numbers.computation.free.fr/Constants/Algorithms/newton.html">http://numbers.computation.free.fr/Constants/Algorithms/newton.html</a>	
Pateikiamas lygčių sprendimo Niutono metodu algoritmas.	Anglų kalba
<a href="http://numericalmethods.eng.usf.edu/topics/secant_method.html">http://numericalmethods.eng.usf.edu/topics/secant_method.html</a>	
Svetainėje pateikiami įvairių skaitinių metodų aiškinimai.	Anglų kalba

Visos rastos svetainės yra anglų arba rusų kalba. Lietuvių kalba svetainių, skirtų skaitiniams metodams, rasti nepavyko.

Angliškos programinės įrangos vartojimas mokyklose nesudaro sąlygų gauti lygiaverčio išsilavinimo valstybine kalba, labai gausi informacija anglų ir kitomis kalbomis užgožia gimtąją kalbą. Todėl informacinėse technologijose būtina puoselėti lietuvių kalbą ir kultūrą. Mokiniai dėstomąjį dalyką geriausiai perpranta tada, kai kompiuterinės programos parašytos gimtąja kalba. [20].

### **3. KOMPIUTERINIŲ MOKOMŲJŲ PROGRAMŲ APŽVALGA**

Kompiuterinių mokomųjų priemonių, skirtų netiesinių lygčių sprendimui skaitiniais metodais nėra. Svetainės, kuriose apžvelgiami skaitiniai metodai yra užsienio kalba.

Įvairių šalių pedagogų tyrimai parodė, kad technologijos padeda moksleiviams ir studentams geriau ir greičiau apdoroti žinias bei išsiugdyti reikiamų įgūdžių. Iširta, kad kompiuterius naudojančių moksleivių testų rezultatai vidutiniškai yra 10–15% geresni nei nenaudojančių [9]. Mokomųjų kompiuterinių programų kūrimas yra labai sudėtingas ir daug žinių (ne tik dalykinių) reikalaujantis procesas. Mokytojai ieško jau sukurtų kompiuterinių mokomųjų programų. Jas galima rasti internete ir kt. Tačiau Lietuvos moksleiviai ir mokytojai susiduria su daugybe problemų, norėdami mokomajame procese pasitelkti kompiuterines programas. Lietuvos moksleivių ir mokytojų apklausa parodė, kad:

- programos pernelyg sudėtingos mokytojams ir (arba) moksleiviams
- mokomosios programos nesuderintos su ugdymo programa;
- mokytojams ir (arba) moksleiviams sunku surasti reikiamos informacijos;
- programos nepakankamai pritaikytos pamokoms;
- dauguma kompiuterinių programų yra užsienio kalba;
- mokytojams nesmagu, kad kai kurie moksleiviai geriau už juos išmano informacijos ir komunikacijos technologijas.

Informacijos amžiuje studentams dėstytojas ir mokomoji knyga nebėra vieninteliai žinių šaltiniai. Informacinių technologijų teikiamų galimybių išnaudojimas informacijos visuomenėje ypač aktualus. Tinkamai panaudotos kompiuterinės technologijos studijų procese sujungia tiek mokomojo dalyko programas, tiek kompiuterinių technologijų funkcijas. Studijų kokybei gerinti, būtina naudoti modernias technologijas, kompiuterines mokymo ir mokymosi priemones[9]

Kompiuteris supaprastina uždavinių sprendimą, leidžia rinkti, keisti pradinius duomenis, grafiškai stebėti proceso ar objekto priklausomybę nuo duomenų pasikeitimo, tai skatina kūrybiškumą ir savarankiškumą. Skaitinių metodų dalyko studijoms sukurta kompiuterinių programų, kurių taikymas būtinas žinių visuomenėje, nes šiuolaikinio gyvenimo iššūkiai skatina naujai pažvelgti į studijų procesą, ieškoti būdų ir priemonių jį tobulinti. Studentai ir moksleiviai orientuojami ne į atskirus dalykinių žinių fragmentus, bet ir į sprendžiamą problemą. Šiems uždaviniams įgyvendinti talkina kompiuterinės technologijos.



Toliau apžvelgsime kompiuterinių programų – MS Excel, Mathcad ir Matlab galimybes, sprendžiant netiesines lygtis skaitiniais metodais. Palyginimui pateiksime pavyzdžių sprendimo pavyzdžius šiomis programomis..

Matlab – tai efektyvi inžinerinių ir mokslinių skaičiavimų kalba. Šioje sistemoje realizuoti klasikiniai tiesinės algebros uždavinių, apibrėžtinių integralų reikšmių radimo, interpoliacijos, diferencialinių lygčių ir lygčių sistemų sprendimų skaitiniai algoritmai. Patogi sąsaja bei integruota paprasta programavimo kalba, kurios paprastumas kompensuojamas grafinių, skaičiavimo bei serviso funkcijų bibliotekomis, padeda greitai kurti aplikacijas, skirtas uždavinio tyrimui ir sprendimui[13].

Mathcad yra populiari dėl matematinių tekstų redaktoriaus, naudojamos standartinės matematinių reiškinių užrašymo formos. Mathcad – tai skaitmeninis matematinis programų paketas, turintis ir kai kurias simbolinio(analitinio) skaičiavimo funkcijas. Skaitmeniniai matematiniai paketai visus veiksmus atlieka apytiksliai, naudodami skaitmeninius skaičiavimo metodus, todėl ne visada gali pateikti analitines išraiškas. Mathcad išsiskiria nesudėtinga vartotojo sąsaja, standartinių funkcijų ir simbolinio skaičiavimo bei programavimo priemonėmis, galimybe naudotis Windows, puikia grafika ir didele pavyzdžių biblioteka[13].

MS Excel programa galima atlikti įvairius skaičiavimus, braižyti diagramas, analizuoti duomenis, rengti ir spausdinti ataskaitas. tai galingas įrankis, kuriuo galite kurti ir formatuoti lenteles, analizuoti duomenis ir bendrai naudotis sukurtais failais. Tai leidžia kurti informatyvią, aiškiai suvokiamą bei greitai atsinaujinančią aplinką. MS Excel leidžia ne tik atlikti sudėtingus skaičiavimus, grafiškai pavaizduoti duomenis bei tendencijas, atlikti duomenų analizę įvairiais lygiais, bet ir pasinaudoti kitomis galimybėmis darbu su didelėmis lentelėmis.

## **4. NETIESINIŲ LYGČIŲ SPRENDIMO METODAI KOMPIUTERINĖSE PROGRAMOSE**

Skaitiniai metodai yra aktualūs, nes dauguma praktinių mokslinių, įvairių taikymų uždavinių yra sprendžiami skaitiniais metodais. Šie metodai aktualūs ir mokymo aspektu. Studentai, studijuodami skaitinių metodų teoriją, sprendžia praktines užduotis, taigi tenka arba patiems programuodami algoritmus, arba naudoti kompiuterines matematikos sistemas.

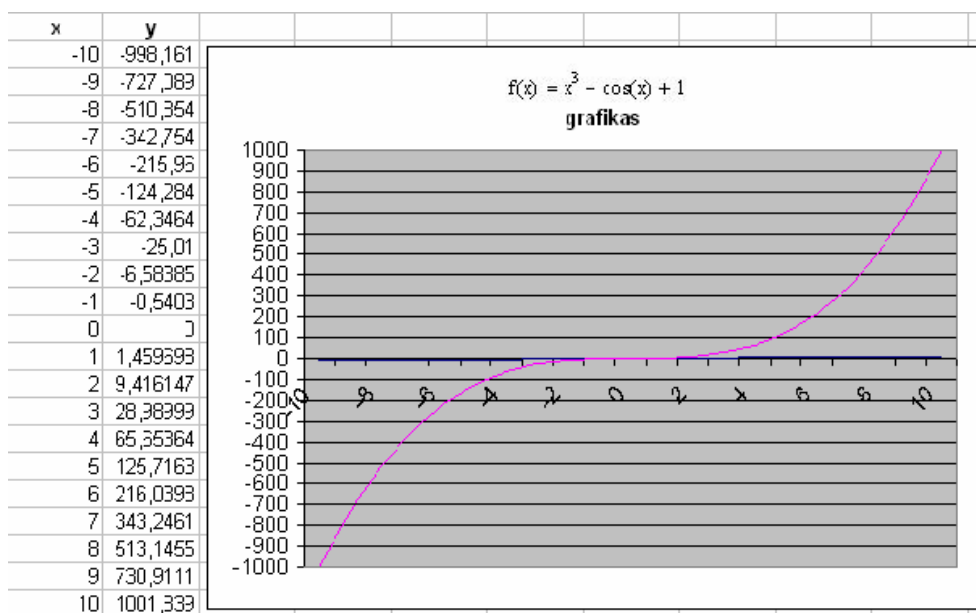
Toliau darbe nagrinėjami trijų universalių kompiuterinių matematinių sistemų – Matlab, Mathcad ir programos MS Excel galimybės, sprendžiant netiesines lygtis aprašytais metodais, lyginamas jų funkcijų aprašymas, taikymai ir sprendimo pavyzdžiai.

### **4.1. Šaknų lokalizacija**

Netiesinė funkcija  $f(x) = 0$  savo apibrėžimo srityje gali turėti baigtinį arba begalinį skaičių nulinių reikšmių, arba neturėti jų iš viso. Dauguma šaknų radimo metodų būtina žinoti intervalus, kuriuose, kuriuose funkcija įgyja nulines reikšmes. Todėl pirmas žingsnis yra intervalų, kuriuose funkcija įgyja nulines reikšmes, nustatymas. Kitaip tariant, pirmiausia sprendžiamas šaknų egzistavimo, intervalo galų nustatymo ir šaknų lokalizacijos uždavinys.

#### **4.1.1. Šaknų lokalizacija Excel programoje**

MS Excel programoje parenkamos  $x$  reikšmės, apskaičiuojamos funkcijos  $y$  reikšmės, ir braižomas grafikas. Iš grafiko 11 pav. matome, kad duota lygtis turi šaknis. Tačiau tiksliai negalime nustatyti kiek jų yra.

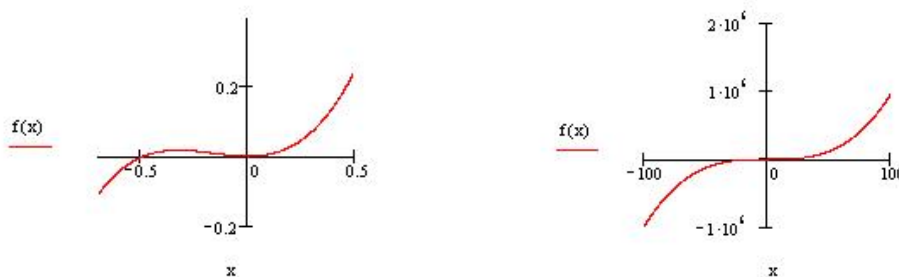


11 pav. Šaknų izoliacijos intervalų nustatymas Excel programoje

Norint tikslinti šaknies buvimo intervalą, reikia pagal grafiką siaurinti x reikšmių intervalą. Tokiu būdu nustatomas šaknų lokalizacijos intervalas. Braižiant grafiką MS Excel programoje kartais negalima nustatyti visų intervalų, kuriuose yra šaknys.

#### 4.1.2. Šaknų lokalizacija Mathcad aplinkoje

Norint nubraižyti funkcijos  $f(x) = x^3 - \cos(x) + 1$  grafiką, reikia naudoti grafiko kūrimo įrankių rinkinį. Pirmiausia pažymimos duomenų lentelės, po to – sritis, kurioje norime matyti grafiką. Į grafiko lange pažymėtus juodus kvadratus įrašomi ašių pavadinimai, ir dar kartą spustelėjus kairįjį pelės klavišą šioje srityje gaunamas funkcijos grafikas. Grafiką galima koreguoti brėžinyje paspaudus du kartus kairįjį pelės klavišą, nustatant ašių režimą (**Crossed Lines**), linijos (**Traces**) storį. Braižome funkcijos  $f(x) = x^3 - \cos(x) + 1$  grafiką.



12 pav. Šaknų izoliacijos intervalų nustatymas Mathcad programoje

### 4.1.3. Šaknų lokalizacija MatLab aplinkoje

Matlab aplinkoje grafikas išvedamas atskirame lange., vadinamame **Figure**.

Grafikų braižymui bei jų charakteristikų nustatymui naudojamos šios funkcijos:

**plot**( $x_1, y_1, s_1, x_2, y_2, s_2, \dots$ ) - išveda kelių funkcijų grafikus viename lange. Dydžiai  $x_i, y_i$  -  $i$  – osios funkcijos argumentų bei funkcijų reikšmių masyvai,  $s_i$ - simbolinis kintamasis (nebūtinai), kuriame gali būti iki 3 specialių simbolių, nurodančių kreivės charakteristikas: linijos tipą, mazgo (taško) tipą bei linijos spalvą.

**plot**( $y, s$ ) – grafike ant x ašies atidedami taškų eilės numeriai.

**grid** įveda arba panaikina papildomą tinklelį. Taip pat galima naudoti **grid on** tinklelio įvedimui ir **grid off** jo panaikinimui.

**title**('pavadinimas') – užrašomas grafiko pavadinimas.

**xlabel**('x\_vardas'), **ylabel**('y\_vardas') – užrašomi ašių pavadinimai.

**legend**('tekstas') – įveda legendą, **legend off** – ją panaikina.

**text**( $x, y, 'tekstas'$ ) – tekstas patalpinamas grafiko taške, kurio koordinatės (x, y).

**inline**(*expr*, *arg1*, *arg2*, ..) įvedama funkcija, kur *expr* – funkcijos išraiška, *arg1*, *arg2* – funkcijos argumentai.

**linspace**(**a**, **b**, **n**) – n elementų vektorius eilutė iš intervalo [a, b].

**[x,y]=ginput**(**grafinio\_lango\_nr**) – nurodo koordinačių x, y, kurioms buvo nustatytas grafinis cursorius lange **Figure**(**grafinio\_lango\_nr**). Jei nebuvo daugiau langų, tai **grafinio\_lango\_nr=1**.

Funkcijos reikšmių skaičiavimui naudojamos paelementės operacijos, t.y. su atitinkamais vektorių elementais, naudojamos operacijos:

**.\*** vieno vektoriaus kiekviena koordinatė dauginama iš kito vektoriaus atitinkamos koordinatės.

**./** vieno vektoriaus kiekviena koordinatė dalinama iš kito vektoriaus atitinkamos koordinatės.

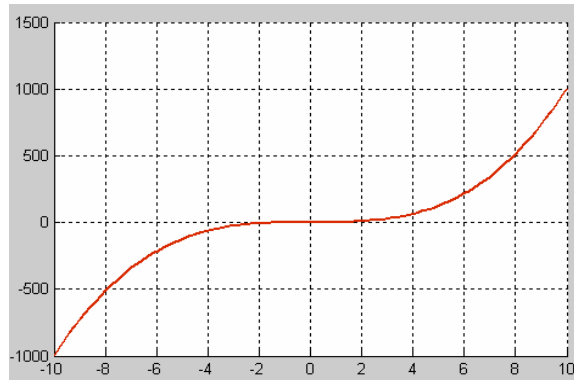
**.^** vieno vektoriaus kiekviena koordinatė kelinama laipsniu, kuri yra kito vektoriaus atitinkama koordinatė.

Grafiko parametrai, tokie kaip, pavadinimas, ašių pavadinimai, legenda, taško koordinatės gali būti įvedami grafiko braižymo lange **Figure**. Todėl rašant programą, jų rašyti nebūtina.

```

% f(x)= x^3 - cos(x) + 1 šaknų lokalizavimas
x = linspace(-10,10,100);
f = x.^3 - cos(x) + 1;
grid on
% Brėžiame grafiką intervale [-10,10]
plot(x, f)
t=ginput(1)
x0=t(1)
feval('f',x)

```



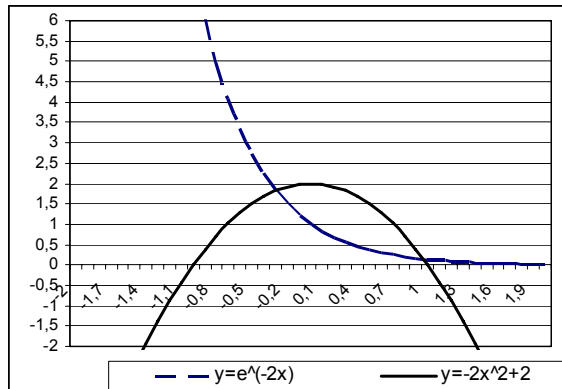
13 pav. Šaknų izoliacijos intervalų nustatymas Matlab programoje

## 4.2. Pusiaukirtos metodas

Sprendimas pusiauikirtos metodu remiasi Bolcano Koši teorema, kuri teigia, kad jei funkcija yra tolydi duotame intervale  $[a, b]$  ir šio intervalo galuose įgyja priešingų ženklų reikšmes, tai tarp duoto intervalo taškų yra toks taškas, kuriame funkcija lygi nuliui. Sprendžiant šiuo metodu duotas intervalas dalinamas pusiau, kol randamas sprendinys norimu tikslumu. Kiekvieną kartą parenkamos tokios intervalo galų reikšmės, kad tenkintų sąlygą  $f(a)f(b) < 0$  [1].

### 4.2.1. Sprendimas pusiauikirtos metodas MS Excel programoje

Tarkime, pusiauikirtos metodu 0,001 tikslumu reikia rasti mažiausią teigiamąjį lygties  $e^{-2x} + 2x^2 - 2 = 0$  sprendinį. Pirmiausia reikia atskirti šios lygties sprendinius: Pasirenkame  $x$  reikšmių intervalą  $[-2, 2]$ ,  $x$  kitimo žingsnį 0,1 ir nubraižome grafikus  $y_1 = y_1 = e^{-2x}$  ir  $y_2 = 2x^2 + 2$



14 pav. Funkcijų grafikų susikirtimo intervalų nustatymas

Iš grafiko matome, kad vienintelis teigiamas lygties sprendinys yra intervale  $[0,1]$ . Tai išeina ir iš Bolcano – Koši teoremos:  $f(0)f(1) = (1-2)e^{-2} < 0$ . Apskaičiuosime šį sprendinį 0,001 tikslumu. Pradinis intervalas yra  $[0,1]$ , jo vidurio taške

$$x_1 = \frac{0+1}{2} = 0,5 \quad \text{funkcijos reikšmė} \quad f(0,5) = e^{-2*0,5} + 2 * 0,5^2 - 2 = -1,132$$

Šis taškas ir yra pirmasis lygties sprendinio artinys; jo paklaida

$$\varepsilon = \frac{1-0}{2} = 0,5, \quad \text{o funkcijos reikšmė šiame taške} \quad f(0,5) = -1,132. \quad \text{Kadangi} \quad f(0,5) < 0, \quad \text{tai}$$

lygties sprendinys yra intervale  $[0,5;1]$ ; jį padaliję pusiau, rasime antrąjį artinį ir apskaičiuosime jo paklaidą bei funkcijos reikšmę tame taške. Lygiai taip pat ieškosime ir kitų artinių kol bus pasiekas norimas tikslumas. Rezultatai surašyti į 4.2.1.1 lentelę. Lentelėje  $n$  - intervalo dalijimų skaičius,  $a_n$  - kairysis intervalo galas,  $b_n$  - dešinysis intervalo galas,  $x_n$  - intervalo vidurio taškas (sprendinio artinys),  $f(x_n)$  - funkcijos reikšmė vidurio taške,  $|b_n - a_n| / 2$  - paklaida[2].

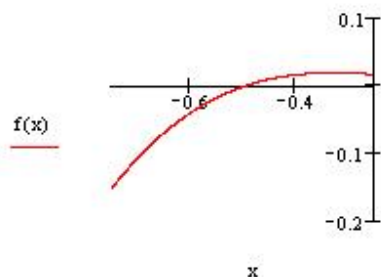
3 lentelė. Pusiaukirtos metodo realizacija MS Excel pagalba

$n$	$a_n(-)$	$b_n(+)$	$x_n = (a_n + b_n)/2$	$f(x_n)$	$ b_n - a_n /2$
0	0,0	1,0	0,5	-1,1321	0,5
1	0,5	1,0	0,75	-0,6519	0,25
2	0,75	1,0	0,875	-0,2950	0,125
3	0,875	1,0	0,9375	-0,0888	0,0625
4	0,9375	1,0	0,9688	0,0210	0,0313
5	0,9375	0,9688	0,9531	-0,0345	0,0156
6	0,9531	0,9688	0,9609	-0,0069	0,0078
7	0,9609	0,9688	0,9648	0,0070	0,0039
8	0,9609	0,9648	0,9629	0,00008	0,00195
9	0,9609	0,9629	<b>0,9619</b>	-0,0034	0,00098

Randame lygties sprendinį:  $x=0,9619$ .

#### 4.2.2. Sprendimas MathCad aplinkoje

$$f(x) := x^3 - \cos(x) + 1$$



Pradinės reikšmės

$$a := -0.6 \quad b := -0.4 \quad \varepsilon := 10^{-10}$$

$$\text{bisec}(f, a, b, \varepsilon) = \begin{pmatrix} -0.4900726384 \\ 30 \end{pmatrix}$$

Šaknis rasta  $\varepsilon := 10^{-10}$  tikslumu atlikus 30 iteracijų 0.4900726384.

```

bisec(f, a, b, ε) :=
| an ← a
| bn ← b
| k ← 0
| while (bn - an) > 2·ε
|   | xn ← (an + bn) / 2
|   | fa ← f(an)
|   | fb ← f(bn)
|   | fxn ← f(xn)
|   | bn ← xn if fa·fxn ≤ 0
|   | an ← xn otherwise
|   | k ← k + 1
| xn ← (an + bn) / 2
| res ← (xn)
|   (k)
| res
    
```

#### 4.2.3. Sprendimas Matlab aplinkoje

Užrašant programą Matlab aplinkoje naudojame standartines funkcija:

**function** – standartinė funkcija, skirta

**while, else, if** – sąlygų tikrinimo funkcijos

```
function psk
% Išspręsti lygtį  $x^3 - \cos(x) + 1=0$ 
% Įvedam funkcija f(x)
f = inline('x.^3 - cos(x) + 1');
root1 = bisek(f, -0.6, -0.4)
root2 = bisek(f, -0.2, 0.2)
% Pusiaukirtos metodas
function c = bisek(f, a, b)
while b - a > eps * 2
c = (a - b) / 2 + b;
if f(c) * f(a) > 0
a = c;
else
b = c;
end
end
root1 = -0.4901
root2 = -1.0537e-008
```

### 4.3. Paprastųjų iteracijų metodas

Paprastųjų iteracijų metodu lygtis  $f(x) = 0$  pakeičiama ekvivalenčia lygtimi  $x = \phi(x)$  ir, pasirinkus bet koki pradinį artinį  $x_0$ , priklausantį sprendinio  $C$  aplinkai  $[c - \delta, c + \delta]$ , pagal taisyklę  $x_{n+1} = \phi(x_n)$  sudaroma iteracinė seka  $\{x_n\}$ . Ji konverguoja į tikslųjį sprendinį  $C$ , jei galioja ši pakankamoji konvergavimo sąlyga:

Funkcija  $\phi(x)$  taško  $C$  aplinkoje  $[c - \delta, c + \delta]$  turi tolydžią išvestinę ir tos išvestinės modulis šioje aplinkoje yra mažesnis už vienetą:

$$|\phi'(x)| \leq q < 1.$$

Galiojant šiai sąlygai,  $(n + 1)$ -ojo artinio paklaida įvertinama taip:

$$|x_{n+1} - c| \leq \frac{q}{1-q} |x_{n+1} - x_n| [2].$$

#### 4.3.1. Sprendimas MS Excel programoje

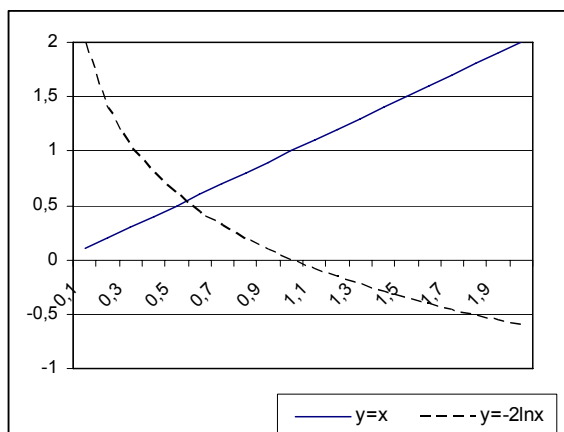
Paprastųjų iteracijų metodu 0,001 tikslumu išspręsimė lygtį  $x + 2 \ln x = 0$ .

Pirmiausia duotą lygtis perrašome tokiu pavidalu:  $x = -2 \ln x$ .

Nubrėžę funkcijų  $y_1 = x$  ir  $y_2 = 2 \ln x$  grafikus, gausime jų susikirtimo tašką, kuris ir bus duotos lygties sprendinys.

Pasirenkame  $x$  reikšmių intervalą  $(0, 2]$ ,  $x$  kitimo žingsnį 0,1 ir nubraižome grafikus.





15 pav. Funkcijų grafikų susikirtimo taško nustatymo intervalas

Nubraižę grafikus, matome, kad duotos lygties sprendinys yra intervale  $(0,1]$ , nes šiame intervale yra funkcijų susikirtimo taškas.

Sąlygoje duotą lygtį pakeičiame ekvivalenčia lygtimi  $x = -2 \ln x$ . Vadinasi mūsų atveju  $\phi(x) = -2 \ln x$

Dabar nustatysime, ar funkcija  $\phi(x)$  tenkina pakankamą konvergavimo sąlygą, t.y. ar  $|\phi'(x)| \leq q < 1$ .

Apskaičiuosime funkcijos  $\phi(x) = -2 \ln x$  išvestinės modulio didžiausią reikšmę intervale  $(0,1]$ .

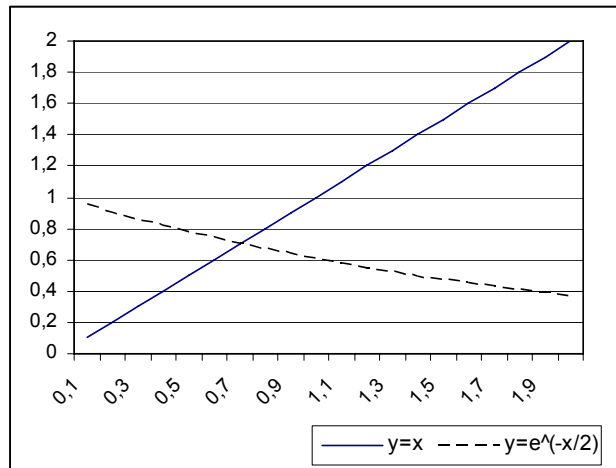
$$|\phi'(x)| = \frac{2}{x}, \quad \max |\phi'(x)| > 1.$$

Iš šios nelygybės matyti, kad iteracinis procesas gali diverguoti, todėl pradinę lygtį perrašykime kitaip:

$$\ln x = -\frac{x}{2}, \quad x = e^{-x/2}.$$

$$\text{Šioje lygtyje } \phi(x) = e^{-x/2}, \quad \phi'(x) = -0,5e^{-x/2}.$$

Dabar nubraižykime lygčių  $y = x$  ir  $y = e^{-x/2}$  grafikus.



16 pav. Funkcijų grafikų susikirtimo taško nustatymo intervalas

Rasime funkcijos  $\phi(x) = e^{-x/2}$  išvestinės modulio didžiausią reikšmę intervale  $[0,1]$ .

$$|\phi'(x)| = 0,5e^{-x/2}.$$

Kadangi ši rodiklinė funkcija yra mažėjanti, tai didžiausią reikšmę ji įgyja su mažiausia  $x$  reikšme, taigi kai  $x = 0$

$$q = \max_{[0,1]} |\phi'(x)| = \max_{[0,1]} 0,5e^{-x/2} = 0,5e^0 = 0,5 < 1.$$

Su šia  $q$  reikšme santykis  $\frac{q}{1-q} = 1$ , ir  $(n+1)$ -ojo artinio paklaida įvertinama tiesiog

dviejų gretimų artinių skirtumu  $|x_{n+1} - x_n|$ .

Tarkime, kad pradinis artinys  $x_0 = 0$ . Tada  $\phi(x) = e^{0/2} = 1$ . Vadinasi sekantis artinys  $x_1 = \phi(x_0) = 1$ . Tikriname tikslumo sąlygą:

$$|x_1 - x_0| = |1 - 0| = 1 > 0,001.$$

Matome, kad tikslumo sąlyga nepatenkinta, nes artinių absoliutinis dydis didesnis už sąlygoje nurodytą tikslumą. Toliau skaičiuojame sekantį artinį

$$x_2 = e^{-1/2} = 0,6065307$$

Artinius patartina skaičiuoti su Excel programa. Skaičiavimo rezultatai pateikti 4 lentelėje.

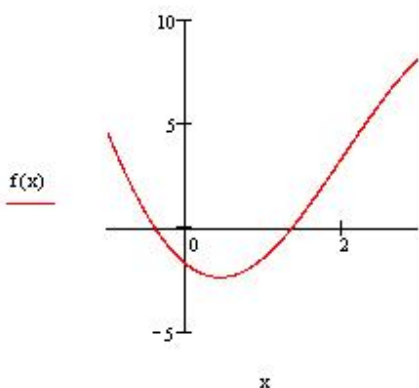
4 lentelė. Sprendimas papr. it. metodu MS Excel pagalba

n	$x_n$	$e^{-x/2}$	$ x_{n+1}-x_n $
0	0	1	1
1	1	0,60653066	0,39346934
2	0,60653066	0,73840315	0,13187249
3	0,73840315	0,69128605	0,0471171
4	0,69128605	0,707765096	0,016479046
5	0,707765096	0,701957409	0,005807688
6	0,701957409	0,703998746	0,002041337
7	0,703998746	0,703280563	0,000718183

Iš lentelės rezultatų matome, kad reikiamo tikslumo yra septintas artinys  $x_7 = 0,70328$

### 4.3.2. Sprendimas Mathcad aplinkoje

$$f(x) := x^2 - 3 \cdot x + 3.25 - 5 \cdot \cos(x)$$



Ieškosime lygybės šaknį, esančią intervale  $[-0.4, 0]$

$$f(-0.4) = 4.695 \times 10^{-3}$$

$$f(0) = -1.75$$

Rasim šaknį funkcijos **root** pagalba

$$x0 := 0, \text{ TOL} := 10^{-10} \text{ root}(f(x0), x0) = -0.3991826759$$

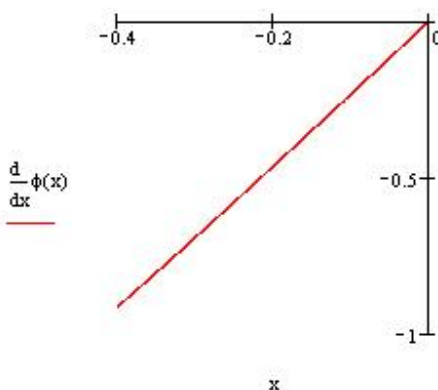
$$\phi(x) := \frac{(x^2 - 5 \cdot \cos(x) + 3.25)}{3}$$

**1 būdas** Parašom lygtį pavidalu  $x = \phi(x)$ , kur

$$\frac{d}{dx} \phi(x) \rightarrow \frac{2}{3} \cdot x + \frac{5}{3} \cdot \sin(x)$$

Tikrinam konvergavimo sąlygą.

Išvestinės grafikas



Maksimalus iteracinės funkcijos išvestinės reikšmės modulis yra kairiame intervalo gale.

$$\phi1(x) := \frac{d}{dx} \phi(x)$$

$$q1 := |\phi1(-0.4)|$$

$$q1 = 0.916$$

$$x0 := -0.4$$

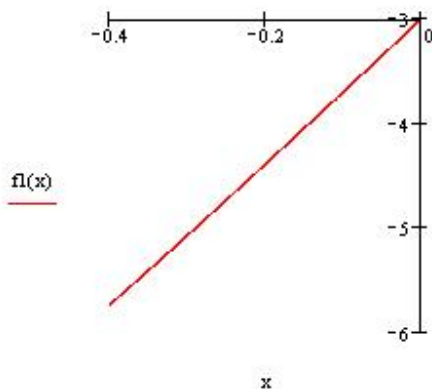
Atliksime tris iteracijas pagal formulę  $x = \phi(x)$

1 iteacija	$x_0 := -0.4$	$y := \phi(x_0)$	$y = -0.39843499$
2 iteacija	$x_0 := -0.39843499$	$y := \phi(x_0)$	$y = -0.3998653686$
3 iteacija	$x_0 := -0.3998653686$	$y := \phi(x_0)$	$y = -0.3985582517$
Gautos reikšmės paklaida:		$ y - (-0.3991826759)  = 6.244 \times 10^{-4}$	

**2 būdas.** Parašysim lygybę pavidalu  $x = x - \alpha f(x)$ , kur iteracinė funkcija  $\phi(x) = x - \alpha f(x)$ ,  $\alpha$ -iteracinis parametras.

$$\frac{d}{dx} f(x) \rightarrow 2 \cdot x - 3 + 5 \cdot \sin(x)$$

Išvestinės grafikas  $f'(x)$



$$f'(x) := 2 \cdot x - 3 + 5 \cdot \sin(x)$$

Maksimalią ir minimalią reikšmę išvestinė įgyja intervalo galuose

$$m := f'(-0.4)$$

$$m = -5.747$$

$$M := f'(0)$$

$$M = -3$$

$$\alpha := \frac{2}{m + M}$$

$$\alpha = -0.229$$

$$q := \left| \frac{M - m}{m + M} \right|$$

$$q = 0.314$$

Atliksime tris iteracijas pagal formulę  $x = \phi(x) = x - \alpha f(x)$

$$\phi(x) := x - \alpha \cdot f(x)$$

1 iteracija	$x_0 := -0.4$	$y := \phi(x_0)$	$y = -0.3989264935$
2 iteracija	$x_0 := -0.3989264935$	$y := \phi(x_0)$	$y = -0.3992627662$
3 iteracija	$x_0 := -0.3992627662$	$y := \phi(x_0)$	$y = -0.399157617$

Gautos reikšmės paklaida:  $|y - (-0.3991826759)| = 2.506 \times 10^{-5}$

### 4.3.3. Sprendimas Matlab aplinkoje

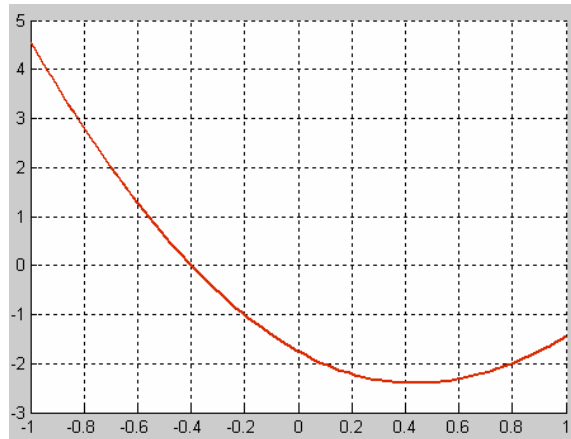
Sprendžiant Matlab aplinkoje naudojamoms anksčiau naudotoms funkcijoms.

```
function papr
% Išspręsti x^2 - 3*x + 3.25 - 5*cos(x) = 0 paprastųjų iteracijų metodu
% Įvedam funkciją f(x)
f = inline('x.^2 - 3*x + 3.25 - 5*cos(x)');
% Brėžiam grafiką intervale [-1,1]
x = linspace(-1,1,100);
% Pažymim ašis
figure('Name', '[-1,1]');
axes('NextPlot', 'Add');
grid on
plot(x, f(x));
% Suvedam lygybę į pavidalą g(x)=x
g = inline('(x.^2 - 5 * cos(x) + 3.25) / 3');
% Pradinis artinys
x0 = -0.4;
% Atliekam 10 metodo žingsnių
for i = 1:100
x0 = g(x0);
```

```

end
% Tikrinam šaknis
f(x0)
>>
ans = 5.7730e-007

```

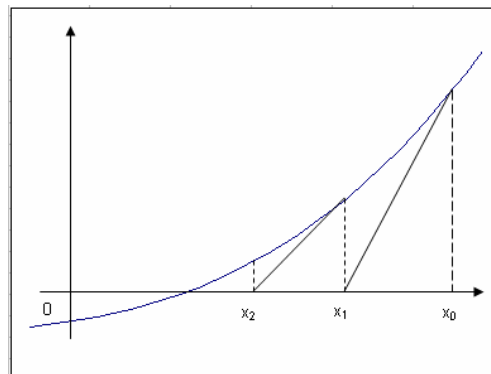


17 pav. Funkcijos grafikas grafiniame Matlab lange

#### 4.4. Niutono metodas

Niutono metodu lygties  $f(x) = 0$  sprendinio artiniai  $x_n$  apskaičiuojami pagal iteracinę formulę  $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$ ,

Kuri geometriškai reiškia funkcijos  $y = f(x)$  grafiko liestinių susikirtimo su OX ašimi taškus.



18 pav. Grafinis sprendimas Niutono metodu

#### 4.4.1. Sprendimas MS Excel programoje

Niutono metodu 0,001 tikslumu reikia rasti lygties  $x - \sin x = 0,25$  sprendinį, kai  $x_0 = 1$ .

$$f(x) = x - \sin x - 0,25$$

$$f'(x) = 1 - \cos x$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} = 1 - \frac{1 - \sin 1 - 0,25}{1 - \cos 1} = 2,085808$$

Tikriname tikslumą:

$$|x_1 - x_0| = |2,085808 - 1| = 2,085808 > 0,001$$

Tikslumo sąlyga nepatenkinta, todėl kintamajam  $x_0$  suteikiame  $x_1$  reikšmę ir ieškome sekančio artinio, kol bus patenkinta tikslumo sąlyga.

Sudarome reikšmių lentelę:

5 lentelė. Sprendimas Niutono metodu MS Excel programos pagalba

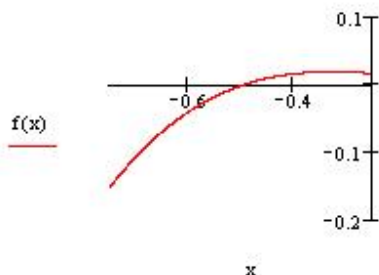
n	0	1	2	3	4
$x_n$	1	1,19898073	1,17179104	1,17122989	1,17122965
$ x_{n+1} - x_n $	0,19898073	0,02718969	0,00056115	0,00000024	0,00000000

Matome, kad ketvirtas artinys yra reikiamo tikslumo  $x_4 = 1,1712937$ .

#### 4.4.2. Sprendimas Mathcad aplinkoje

$$f(x) := x^3 - \cos(x) + 1$$

$$\text{mN\_step}(f, x) := x - \frac{f(x)}{\frac{d}{dx}f(x)}$$



Pradinės reikšmės

$$\varepsilon := 10^{-10} \quad x_0 := -0.4$$

```

mN(f, x0, ε) :=
    xold ← x0
    k ← 1
    xnew ← mN_step(f, xold)
    while |xnew - xold| > ε
        xold ← xnew
        xnew ← mN_step(f, xold)
        k ← k + 1
    res ← (xnew)
           (k - 1)
    res
    
```

$$mN(f, x_0, \varepsilon) = \begin{pmatrix} -0.4900726385 \\ 5 \end{pmatrix}$$

$$x_0 := -0.6$$

$$mN(f, x_0, \varepsilon) = \begin{pmatrix} -0.4900726385 \\ 5 \end{pmatrix}$$

Šaknis, rasta  $\varepsilon := 10^{-10}$  tikslumu atlikus 5 iteracijas -0.4900726385 .

Pasirinkus kitą artinį šaknis rasta atlikus 3 iteracijas tokiu pačiu tikslumu.

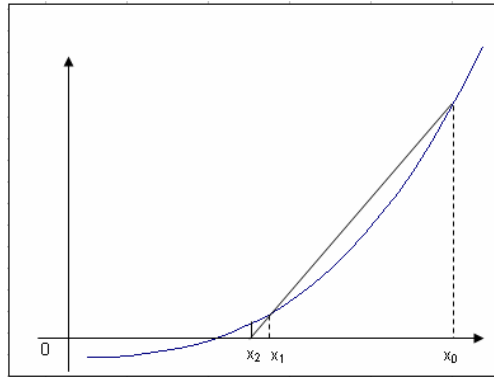
#### 4.4.3. Sprendimas MatLab aplinkoje

Išvestinei apskaičiuoti naudojama funkcija  $df$

```
function niut
% Išspręsti x^3 - cos(x) + 1=0
% Įvedam funkcija f(x)
f = inline('x.^3 - cos(x) + 1');
% Jos išvestinė
df = inline('3*x.^2 + sin(x)');
s1 = newton(f, df, -0.5)
%Tikrinam šaknis
f(s1);
s2 = newton(f, df, -0.1)
% Tikrinam šaknis f(root2)
% Niutono metodas
function sak = newton(f, df, x0)
sak = x0 - f(x0) / df(x0);
ssak = x0;
while abs(ssak - sak) > eps
t = ssak;
ssak = sak;
sak = t - f(t) / df(t);
end
s1 =-0.4905; s2 = -0.0428
```

#### 4.5. *Kirstinių metodas*

Vartojant Niutono metodą, kiekvienoje iteracijoje reikia skaičiuoti funkcijos  $f(x)$  ir išvestinės  $f'(x)$  reikšmes. Šie veiksmai padidina skaičiavimų apimtį ir, be to, kai kada yra sudėtinga analitiškai rasti funkcijos išvestinę. Lengviau yra sudaryti funkcijos grafiko kirstinės lygtį ir rasti jos susikirtimo su Ox ašimi taškų koordinatas. Kirstinių metodas tuo ir grindžiamas – brėžiamos kreivės  $y = f(x)$  kirstinės, kurių susikirtimo su Ox ašimi taškai yra lygties  $f(x) = 0$  sprendinio artiniai.



19 pav. Grafinis sprendimas kirstinių metodu

Artiniai skaičiuojami pagal formulę:

$$x_{n+2} = x_{n+1} - f(x_{n+1}) \frac{x_{n+1} - x_n}{f(x_{n+1}) - f(x_n)}$$

#### 4.5.1. Sprendimas MS Excel programoje

Kirstinių metodu 0,00001 tikslumu rasti lygties  $x^3 - x - 0,231 = 0$  sprendinį

Šios lygties tikslus sprendinys yra  $c = 1,1$ . Todėl paklaidą apskaičiuosime formule

$\varepsilon_n = |x_n - c|$ . Artinius skaičiuosime tol, kol bus nepatenkinta ši tikslumo sąlyga.

Pradinius artinius imkime  $x_0 = 4, x_1 = 3$ .

Apskaičiuojame  $x_2$  artinį:

$$x_2 = 3 - (3^3 - 3 - 0,231) \frac{3 - 2}{(3^3 - 3 - 0,231) - (4^3 - 4 - 0,231)} = 2,339750$$

$$\varepsilon_2 = |x_2 - c| = 1,23975$$

MS Excel programos pagalba apskaičiuojame kitus artinius ir sudarome artinių ir paklaidų reikšmių lentelę:

6 lentelė. Sprendimas kirstinių metodu MS Excel programos pagalba

n	0	1	2	3	4	5	6	7	8	9
$x_n$	4	3	2,339750	1,840179	1,498238	1,277084	1,157067	1,11036	1,100697	1,100009
$\varepsilon_n$	2,9	1,9	1,23975	0,740179	0,398238	0,177084	0,057067	0,01036	0,000697	0,000009

Sprendinys norimu tikslumu  $x=1,100009$



### 4.5.2. Sprendimas Mathcad aplinkoje

```

x_r(n) := | i ← 1
          | x1 ← x_guess1
          | x2 ← x_guess2
          | while i ≤ n
          |   | x_root ← x2 -  $\frac{f(x2)(x1 - x2)}{f(x1) - f(x2)}$ 
          |   | x1 ← x2
          |   | x2 ← x_root
          |   | i ← i + 1
          | x_root
    
```

Įvedame iteracijų skaitliuką.

Pasirenkame du sprendinių artinius

Ieškome naujo artinio n kartų

Sekančio artinio apskaičiavimui naudojame jau apskaičiuotą artinį.

### 4.5.3. Sprendimas Matlab aplinkoje




```

clear;
x0=2.1;
x1=2.3;
while abs(x0-x1)>0.001
    x1=x0-(x0^3-9*x0+4)*(x0-x1)/((x1^3-9*x1+4)-(x0^3-9*x0+4));
x0=x1;
% disp(['u ' num2str(u)]);
end;
format short;
disp(x1);
    
```

## 4.6. Uždavinių sprendimo kompiuterinėmis programomis palyginimas




### 4.6.1. Pusiaukirtos metodu

7 lentelė. Sprendimų pusiauikirtos metodu kompiuterinėmis programomis palyginimas

 <b>Excel</b>	Sprendžiant lygtį pusiauikirtos metodu, sprendinių radimui reikia sudaryti reikšmių lentelę. Parenkant intervalo galų koordinatas, reikia tikrinti funkcijos reikšmių ženklus.
 <b>Mathcad</b>	Ieškant sprendinių šioje aplinkoje nereikia turėti gerų programavimo įgūdžių. Objektai, su kuriais reikia atlikti veiksmus, pirmiausia aprašomi, o tik po to atliekamos operacijos.
 <b>Matlab</b>	Sprendžiant šioje aplinkoje, reikia turėti minimalių algoritmavimo žinių.




#### 4.6.2. Paprastųjų iteracijų metodu

8 lentelė. Sprendimų papr. iteracijų metodu kompiuterinėmis programomis palyginimas

 <b>Excel</b>	<p>Funkcija pertvarkoma į tokį pavidalą, kad artinys konverguotų į sprendinį. Sudarinėjama reikšmių lentelė</p>
 <b>Mathcad</b>	<p>Sprendžiant šioje aplinkoje pirmiausia reikia parašyti lygtį tokiu pavidalu, kad būtų tenkinama konvergavimo sąlyga. Kaip matome iš pateikto pavyzdžio, sprendimo užrašymo eiga gana sudėtinga.</p>
 <b>Matlab</b>	<p>Pertvarkant lygtį taip pat turi būti tenkinama konvergavimo sąlyga.. Sprendimo algoritmas užrašomas, panaudojant standartines Matlab funkcijas ir komandas.</p>




#### 4.6.3. Niutono metodu

9 lentelė. Sprendimų Niutono metodu kompiuterinėmis programomis palyginimas

 <b>Excel</b>	<p>Funkcijos išvestinę reikia rasti pačiam ir įrašyti į Niutono formulę.</p>
 <b>Mathcad</b>	<p>Išvestinei apskaičiuoti yra komanda pagrindinių įrankių rinkinyje.</p>
 <b>Matlab</b>	<p>MatLab aplinkoje išvestinei skaičiuoti naudojama funkcija <i>dif</i>. Jos užrašymo sintaksė paprastesnė, nei Mathcab aplinkoje. Funkcijos išvestinė naudojama apskaičiuojant funkcijos reikšmes.</p>

#### 4.6.4. Kirstinių metodu

10 lentelė. Sprendimų Kirstinių metodu kompiuterinėmis programomis palyginimas

 <b>Excel</b>	<p>Artinio užrašymo formulė sudėtinga. Todėl reikia atskirai skaičiuoti funkcijos reikšmes, o tik po to jas panaudoti formulėse.</p>
 <b>Mathcad</b>	<p>Sprendimo algoritmo užrašymo forma nėra sudėtinga, reikia naudotis įrankių palete. Specialių funkcijų spręsti lygtis šiuo metodu nėra</p>
 <b>Matlab</b>	<p>Užrašant sprendimo algoritmą reikia žinoti tik kelias MatLab programavimo kalbos komandas</p>

## 5. MOKOMOSIOS KOMPIUTERINĖS PRIEMONĖS PROJEKTAVIMAS

### 5.1. Kompiuterinių priemonių vertinimas

Kompiuterinės mokomosios programos– tai programos, skirtos mokymui ir tiesiogiai jam taikomos. Šios programos specialiai tam sukurtos. Kompiuterinių mokomųjų programų yra labai įvairių: nuo elementarių programų mokyti dviejų skaičių sudėties iki sudėtingų astronomijos enciklopedijų.

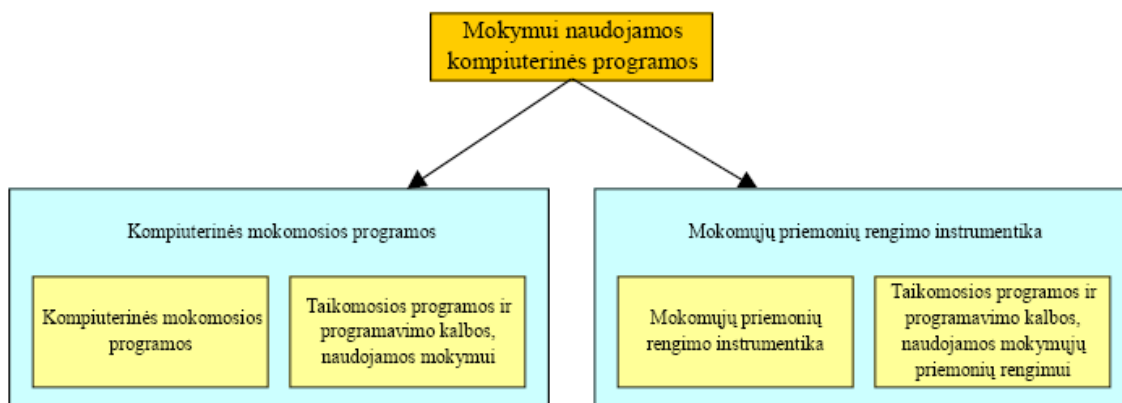
Mokymui skirta programinė įranga galima suskirstyti į dvi stambias dalis: kompiuterines mokomasias programas ir mokomųjų priemonių rengimo instrumentiką (žr. 20 pav.).

Kompiuterines mokomosios programos- tai programos, skirtos mokymui ir tiesiogiai jam taikomos. Šios programos specialiai tam sukurtos. Kompiuteriniu mokomųjų programų yra

labai įvairiu: nuo elementarių programų mokyti dviejų skaičių sudėties iki sudėtingų astronomijos enciklopedijų.

Mokymui sėkmingai gali būti naudojami programavimo kalbų paketai bei taikomoji programinė įranga (tekstu rengimo sistemos, skaičiuoklės, grafikos ir muzikos redaktoriai, statistikos paketai, telekomunikacijos programos ir pan.), kuri nėra tiesiogiai tam skirta.

Taikomosios programos dažniausiai būna pagalbinės priemonės pamokose ar mokantis savarankiškai. Pavyzdžiui, skaičiuoklės, matematikos bei statistikos paketai naudojami per matematikos, chemijos ar fizikos pamokas sudėtingiems skaičiavimams atlikti, tekstu rengimo ir telekomunikacijos programos- per užsienio kalbos pamokas laiškas rašyti, informacijai gauti bei siusti.



20 pav. Mokymui naudojamų kompiuterinių programų klasifikacija

Mokomųjų priemonių rengimo instrumentika - tai programos, skirtos mokomosioms priemonėms sudaryti. Jas pasitelkus galima kurti įvairias kompiuterines mokymo programas: demonstracinę medžiagą, kompiuterines kontrolines užduotis ir pan. Naudojantis mokomųjų priemonių rengimo instrumentika dažnai rengiama ir įprastinė mokomoji medžiaga: kompiuteriu sudaromos ir išspausdinamos kontrolinių darbų užduotys, įvairi padalomoji medžiaga pamokoms. Pagrindinė mokomųjų priemonių kūrimo instrumentikos savybė - paprastumas: nereikia būti programuotoju, norint paruošti priemonę naujai temai dėstyti ar moksleivių žinioms patikrinti.

Kiekvienos kompiuterinės mokomosios priemonės tipinės dalys: teorinė dalis, praktinė dalis ir testavimas. Išnaudojant kompiuterio plačias vaizdinės, tekstinės ir garsinės informacijos saugojimo bei pateikimo galimybes kompiuterinėje priemonėje turėtų būti galimybė pereiti nuo gana pasyvaus įprastinio grupinio mokymo į aktyvų ir efektyvų individualizuotą mokymą. Kompiuteris leidžia diferencijuoti mokymą, prisiderinant prie konkretaus besimokančiojo galimybių. Tuo pačiu kompiuteriu sėkmingai gali naudotis kaip gabūs, taip ir silpnai pasirengę vaikai. Ypač teigiamai reikia vertinti kompiuterinio mokymo galimybę nuosekliai ir nenuilstant formuoti besimokančiojo mokėjimus ir įgūdžius, įvertinant jo klaidas ir mokymosi istoriją.

Todėl svarbiausias kompiuterinio produkto kūrimo tikslas turi būti ne informaciją perduodančios technologijos kūrimas, o supratimo, kaip individai mąsto sprendami įvairius klausimus, ugdymas. Vis dėlto nėra lengva atsakyti į klausimą, kokia mokomoji programa yra gera[9].

Iki šiol mokomosios programos vertinimo kriterijai buvo tokie:

- *efektyvumas*— programa leidžia vartotojui dirbti labai produktyviai;
- *lengvas išmokimas* - vartotojas gali greitai pereiti nuo sistemos nežinojimo iki jos minimalaus naudojimo;
- *lengvas įsimenamumas* - po pertraukos nereikia mokytis sąsajos iš naujo;
- *lengvas klaidų ištaisymas*- vartotojui padarius klaidą, padariniai lengvai pašalinami;
- *malonumas naudotis*- vartotojui patinka naudotis šia sąsaja, informacija programoje

pateikiama patraukliai[9].

Šie kriterijai labai subjektyvūs. Pavyzdžiui, tokie vertinimo kriterijai, kaip mokymosi lengvumas ir lengvas įsimenamumas, priklauso nuo subjektyvaus požiūrio ne tik į mokomąjį dalyką, bet ir į jo pateikimą kompiuteryje. Toks vertinimas kritikuotinas, nes atskiria mokymosi objektą nuo mokymosi konteksto.

Todėl pagrindinė mokymuisi skirtos programos kūrimo varomoji jėga turi būti supratimas, kaip žmonės mąsto sprenddami skirtingas užduotis.

Kognityvusis požiūris į mokymąsi akcentuoja kokybinį darbo su mokomąja programa vertinimą, kuris siejamas su mokymosi proceso tyrimu – kodėl ir kaip mokinys naudojami konkrečia informacija.

Kokybinio vertinimo kriterijai yra šie:

- *mokymosi tikslas*: ar vartotojui pavyksta perprasti autoriaus siekius? Kokiam mokymuisi skirta konkreti mokomoji programa - faktams įsiminti ar įgūdžiams ugdyti (kelti hipotezes, ieškoti alternatyvių sprendimų, konceptualiai mąstyti);

- *mokomojo dalyko vertinimas*: ar išsami mokomoji medžiaga? Ar ji yra patikima?

- *individualus informacijos peržiūrėjimo būdas*: ar yra sąlygos pasirinkti vienokį ar kitokį informacijos peržiūrėjimo būdą? (Informacijos peržiūrai įtakos turi pradinis išsilavinimas, domėjimosi objektas, užduotis, suformuluota darbo pradžioje, ir kiti veiksniai);

- *hipertekstinės sąsajos ypatybės*: ar yra sąlygos informacijos paieškai? Ar informacijos srautas suskaidytas atskirais vienetais? Ar naudojamos grafinės programos įrangos galimybės? Ar yra sąlygos nutraukti mokymąsi?

- *informacijos pateikimo supratimas*: ar lengva orientuotis informacijos erdvėje? Koks informacijos sandaros poveikis mokomojo dalyko supratimui?

Kiekvieno produkto kokybė priklauso nuo jo kūrimo tikslo ir vartojimo būdų. Todėl mokomoji programa turi būti vertinama pagal tai, kokias pažintines mokinių savybes ji ugdo.

Mokymosi kokybė priklauso ne vien tik nuo technologijos priemonių panaudojimo, bet ir nuo daugybės tarpusavyje susijusių veiksnių – mokymosi tikslų, mokymosi veiksmų, mokytojo vaidmens, klasės kultūros. Visi šie tarpusavyje susiję veiksmai turi įtakos ir mokomosios priemonės vertinimui. Todėl objektyvus, atskirtas nuo mokymosi konteksto mokomosios programos vertinimas yra neįmanomas.

Kadangi mokomųjų kompiuterinių priemonių yra daug ir įvairių: nuo elementarių iki pakankamai sudėtingų, todėl jas galima klasifikuoti pagal jų turinį ir paskirtį. Kiekvienas mokomųjų programų tipas pasižymi tam tikromis, tik jam būdingomis savybėmis. Dažnai ta pati mokomoji priemonė pasižymi įvairiomis savybėmis ir taikoma įvairaus pobūdžio mokomajai veiklai: demonstravimui, eksperimentavimui, modeliavimui, įgūdžių formavimui, savarankiškam mokymuisi ir t.t.

Kompiuterinės mokymo priemonės galima vertinti keliais aspektais.

Jan Wibe ir Charles Jennings [19] didžiausią dėmesį skiria mokymo turinio analizei, o ne priemonėms, kuriomis visa tai praktiškai realizuojama. M.Vivet [18] be visų šitų dalykų, vertindamas kompiuterinę priemonę, didelį dėmesį skiria ne tik mokymo metodų vertinimui, bet ir priemonės dizainui, ryšiams su vartotoju, programos lankstumui, valdymui, mokytojo darbo palengvinimui, kontrolės priemonėms, sandarai ir t.t.

Pagal M.Vivet patyrimą sudaryta vertinamųjų objektų seka. [18]

1. Kokių tikslų siekiama, naudojant mokymo priemonę.
2. Naudojamos technologijas (kiek jos yra naujos).
3. Mokymo priemonės vertinimas.
4. Mokymo turinio vertinimas.
5. Mokymo su mokomąja programa vertinimas.
6. Mokymosi su mokomąja programa vertinimas.

Atsižvelgiant į šiuos vertinimo kriterijus, visapusiška kompiuterinės priemonės vertinimo schema galėtų būti tokia:

- Tikslai, kurių siekiama naudojant mokymo priemonę.
- Mokomosios medžiagos turinio ir jo pateikimo vertinimas.
- Technologijų vertinimas.
- Terminologija.
- Mokinio darbas ir grįžtamasis ryšys (kontrolė).
- Navigacija (Sąsajos su vartotoju ypatybės).
- Ekraninis vaizdas (Informacijos pateikimas vartotojui).

Vertinant kompiuterines priemones, reikia surinkti įvairių žmonių požiūrį į priemonę.

Tai gali būti: mokiniai, kurie studijavo šį dalyką; mokytojai, kurie gerai išmano dalyko turinį, mokinių poreikius; ekspertai, kurie yra dalyko žinovai. Šie žmonės galėtų išsamiai išanalizuoti turinį ir pateikti vertingų kritinių pastabų. Todėl reikia parengti klausimų sąrašą arba lentelę ir pateikti ją priemonę naudojantiems asmenims. Atsakymus ir kritines pastabas išanalizuoti.

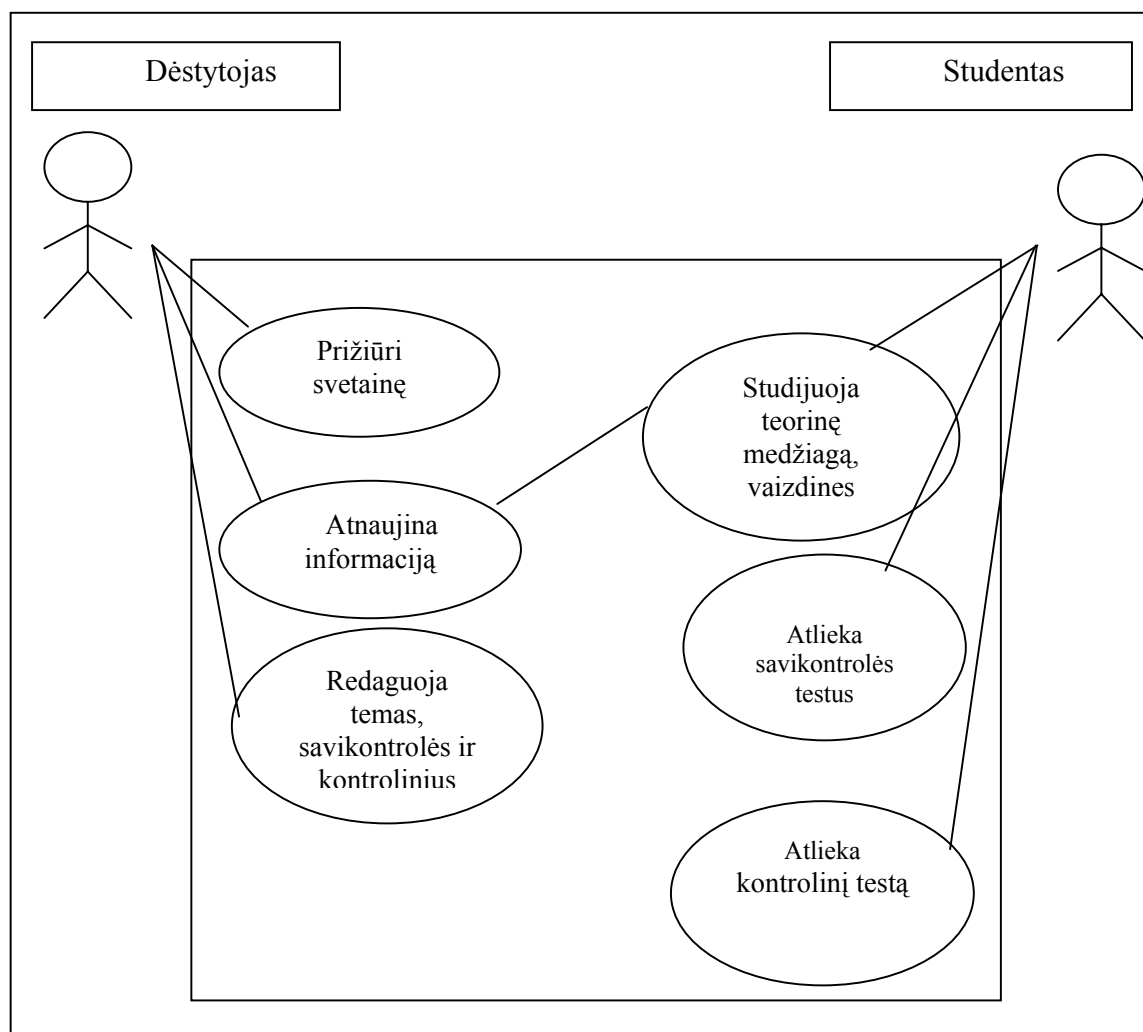
## ***5.2. Kompiuterinė mokomosios priemonės paskirtis***

Klaipėdos verslo ir technologijų kolegijoje studentai studijuoja dieniniu ir neakivaizdiniu būdu. Ir vieni ir kiti studentai mokosi pagal tą pačią studijų programą, praeina tuos pačius dalykus, jiems dėsto tie patys dėstytojai, todėl ir dėstymo metodai mažai kuo skiriasi. Vienintelis

skirtumas yra tai, kad smarkiai skiriasi kontaktinių valandų skaičius. Neakivaizdinių studijų studentai turi labai nedaug kontaktinių valandų, todėl jie daug ką turi išsiaiškinti savarankiškai. Pastaruoju metu vis labiau vystomas ir plečiamas individualus bei savarankiškas studentų darbas. Jis padeda įgyti žinių minimumą, ugdyti loginį mąstymą. Žinoma, tam būtina metodinė pagalba – studentai turi būti aprūpinti reikiama vadovėliais, uždavinynais, mokymo priemonėmis.

Kita vertus studentų savarankiškas darbas yra neatskiriamas nuo žinių įsisavinimo kontrolės – vieno iš svarbiausių mokymo proceso elementų. Nuo savarankiško darbo ir jo kontrolės tinkamo organizavimo priklauso studentų aktyvumas paskaitų, praktikumų metu, jų požiūris į studijuojamą medžiagą bei studijas apskritai. Manau, sukurta mokomoji kompiuterinė priemonė bus kaip papildoma medžiaga, studijuojant atskiras dalyko temas.

Kompiuterinės priemonės vartotojai: studentai, dėstytojai(21 pav.)



21 pav. Svetainės vartotojų grupės

### 5.3. Reikalavimai aparatinei ir programinei įrangai

<u>Minimalūs reikalavimai aparatinei įrangai</u>	
Procesorius	233 MHz
Operatyvioji atmintis	16 MB
Vieta kietajame diske	24 MB
Kiti būtini kompiuterio įrenginiai	Vaizduoklis, klaviatūra, pelė
<u>Minimalūs reikalavimai programinei įrangai</u>	
Operacinė sistema	Microsoft Windows XP
Interneto naršyklė	Internet Explorer 6

### 5.4. Technologijų kompiuterinei priemonei parinkimas

Mokomajai kompiuterinei priemonei sukurti pasirinkau svetainių kūrimo įrankį FrontPage 2003, Java Script programavimo kalbą, vaizdų įrašymo programą BB FlashBack, animacijos kūrimo programėlę Animated GIF Producer, Delphi programavimo sistemą.

„MS FrontPage 2003“, svetainių kūrimo įrankis apima priemonės, lankstumą ir funkcionalumą, kuriais naudodamiesi galite sukurti svetainę. Įtraukti turinį galima įvedant ir formatuojant tekstą. Darbo su tekstu principai tokie patys, kaip programoje „Microsoft Word“. Žymėti ir formatuoti tekstą galite taip pat, kaip minėtoje programoje.

Jeigu jau turite „Word“ dokumente įvestą tekstą, galite jį kopijuoti ir įklijuoti savo puslapyje arba visą failą tiesiog vilkti ir numesti tiesiai puslapyje.

„FrontPage“ tema suteikia galimybę visai svetainei pritaikyti jau sukurtą profesionalų ir koordinuotą formatą. Jeigu naudojama tema, antraštes galima formatuoti naudojant antraščių stilius. Tada tema parinks stilius ir juos atitinkamai formatuos.

**Java Script** – tai scenarijų kalba, skirta interaktyvioms svetainėms projektuoti, ji gali sąveikauti su *HTML* kalba parašytais tinklalapiais ir suteikti jiems dinamiškumo. Interaktyvios mokomosios svetainės palengvina mokytojo darbą mokykloje ar dėstytojo universitete. Mokiniais ir studentams yra daug paprasčiau ir lengviau įsiminti pateiktą informaciją.

**Animated GIF Producer** yra greita, paprasto naudojimo programa, skirta įvairių tipų animacijoms. Su ja galima kurti animacijas, mygtukus ir banerius, kurie internetinį puslapį pavers gražesniu, labiau traukiančiu akį. Jums nereikės daugiau valandų valandas mokytis, kaip su tam tikra programa sukurti paprastą GIF animaciją - su šia programa be jokių pamokų sukursite ją per kelias minutes.



**BB FlashBack** - vaizdų įrašymo programą. Šia programa galima filmuoti vaizdą, vykstantį ekrane. Tai patogi priemonė, ruošiant vaizdines, demonstracines priemones.

**Delphi** programavimo sistema. Delphi - daug kur pripažinta viena iš lyderių, kuriant programas Windows sistemos platformai. Taikant objektinio programavimo tendencijas, daugelis paprastų ir profesionalių programuotojų, su Delphi gali sukurti nuo primityvių iki pačių sudėtingiausių, bei aukštos kokybės programų. Programos kūrimą Delphi sistemoje galima suskirstyti į šiuos etapus:

- Projekto sąsajos kūrimas.
- Savybių nustatymas.
- Programos teksto rašymas.

Delphi projektuose kiekvienam langui automatiškai sukuriama du failai. Viename jų, kurio vardo plėtinys yra pas, pateikiamas lango modelio aprašymas objektine Paskalio kalba. Šį failą vadinsime lango programos modulių. Antrame faile, kurio plėtinys dfm, o pats vardas sutampa su programos modulio vardu, kuriama išsamus lango aprašymo dvejetainis kodas.

Paskalio kalbos programų moduliai yra programos sudėtinės dalys, kurios sujungiamos į vieningą sistemą pagrindinėje programoje. Joje aprašoma keletas standartinių veiksmų, kuriuos reikia atlikti paleidžiant projekto programą ir atveriant jos pagrindinį langą. Šios programos tekstas Paskalio kalba yra su vardo plėtiniumi dpr.

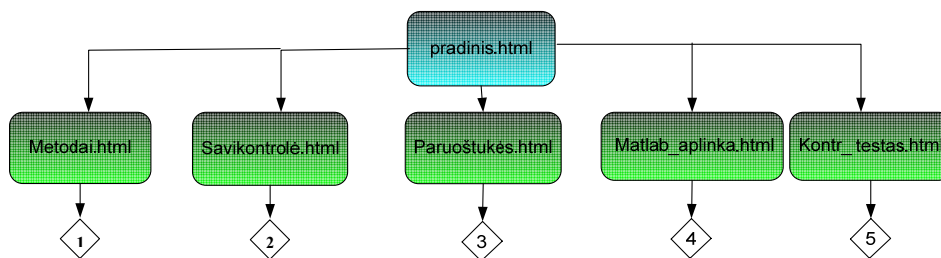
Projektuose būna dar vienas dvejetainis failas, kuris vadinamas išteklių failu. Jame saugomi informaciniai duomenys apie projekte kuriamą taikomąją programą, programai skiriamas ženklinimo vaizdas.. Šio failo plėtinys yra res.. Duomenys apie projekto tvarkymui parinktas Delphi aplinkos parametrų reikšmes saugo failai su vardų plėtiniais dof ir dsk. Jų kūrimu ir tvarkymu programuotojui rūpintis nereikia. Šį darbą automatiškai atlieka pati programavimo aplinka.

Delphi aplinkoje programos galima sėkmingai rengti tvarkant vien tik programų modulius, o visų kitų projekto failų tvarkymą patikėti pačiai programavimo aplinkai. Siekiant, kad aplinkos automatiškai kuriami failai nepridarytų programuotojui rūpesčių, rekomenduojama kiekvienam Delphi projektui paskirti po atskirą katalogą.

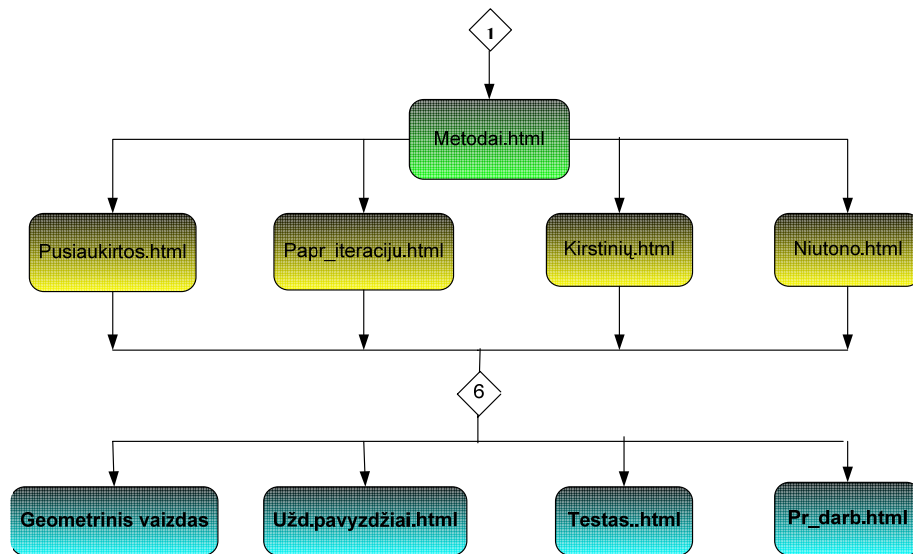
Atlikus programos kompiliavimą, mašininis kodas įrašomas faile su plėtiniumi exe, kuris gali būti naudojamas atskirai nuo kitų projekto failų[15].

### 5.5. Kompiuterinės priemonės struktūrinė ir hiperryšių schema

Įvairių šalių pedagogų tyrimai parodė, kad technologijos padeda moksleiviams geriau ir greičiau apdoroti žinias bei išsiugdyti reikiamų įgūdžių. Iširta, kad kompiuterius naudojančių moksleivių testų rezultatai vidutiniškai yra 10–15% geresni nei nenaudojančių [6]. Mokomųjų kompiuterinių programų kūrimas yra labai sudėtingas ir daug žinių (ne tik dalykinių) reikalaujantis procesas. Todėl prieš pradėdant kurti kompiuterinę priemonę patogu nusibraižyti struktūrinę schemą.



22 pav. Pagrindinio puslapio struktūrinė schema



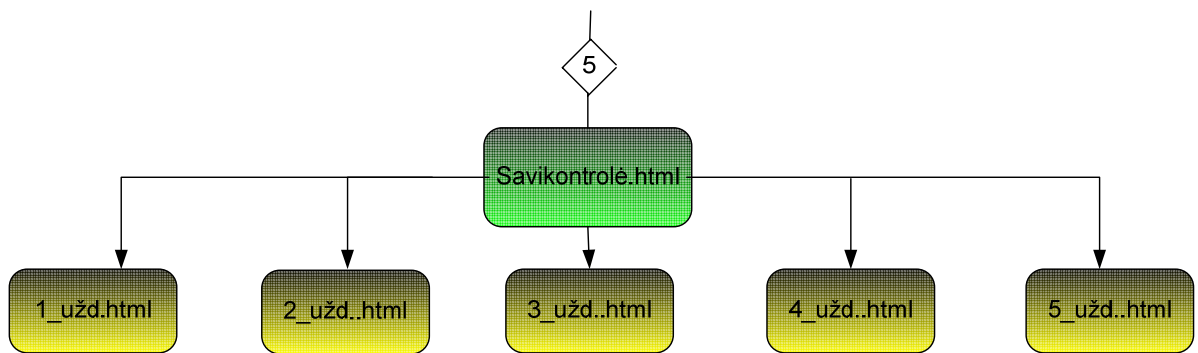
23 pav. Puslapio “Metodai” struktūrinė schema

Kiekvienam metodui skirtame puslapyje yra teorija, uždavinių sprendimo pavyzdžiai, geometrinė metodo interpretacija.

Paruošukų puslapyje yra pateikti uždavinių sprendimo Matlab aplinkoje algoritmų kodai. Studentas, gavęs užduotį turi pasirinkti programą, sugebėti pakoreguoti kodą ir išspręsti užduotį.

Praktinių darbų puslapyje yra pateiktos praktinių darbų užduotys.

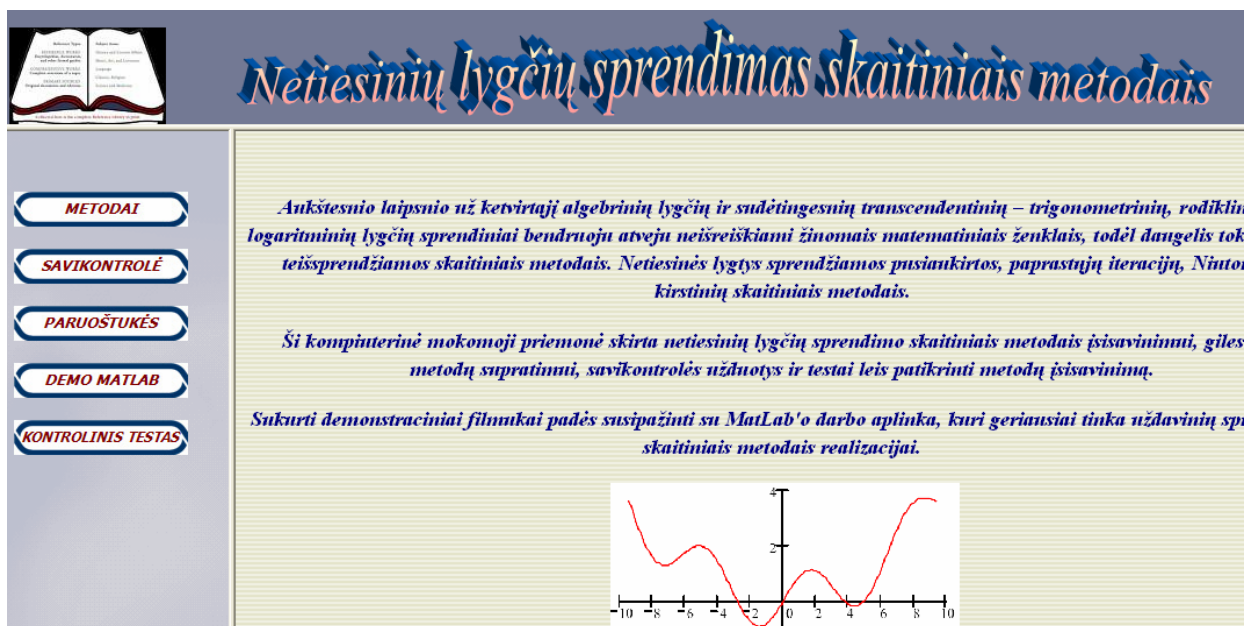
Savikontrolės puslapyje Java Script kodu sukurtos penkios savikontrolės užduotys.



24 pav. Puslapio “Savikontrolė” struktūrinė schema

### 5.6. Kompiuterizuota teorinė medžiaga

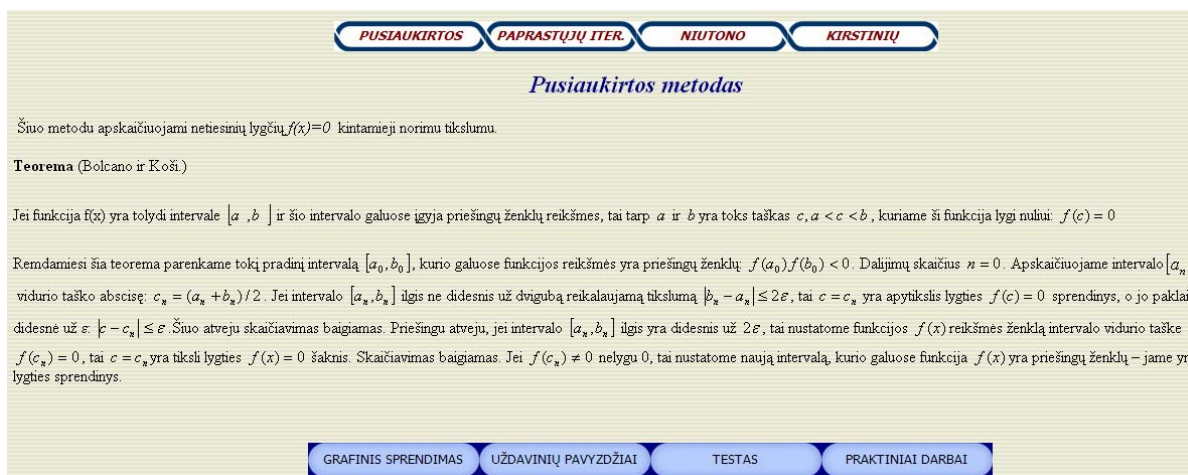
FrontPage redaktoriumi svetainėje sukuriamas interaktyvusis meniu su *Java Script*, taip svetainės lankytojai nukreipiami į pasirinktus puslapius. Tai svarbiausias valdymo objektas, esantis interaktyvioje mokymo svetainėje.



25 pav. Pradinis svetainės langas

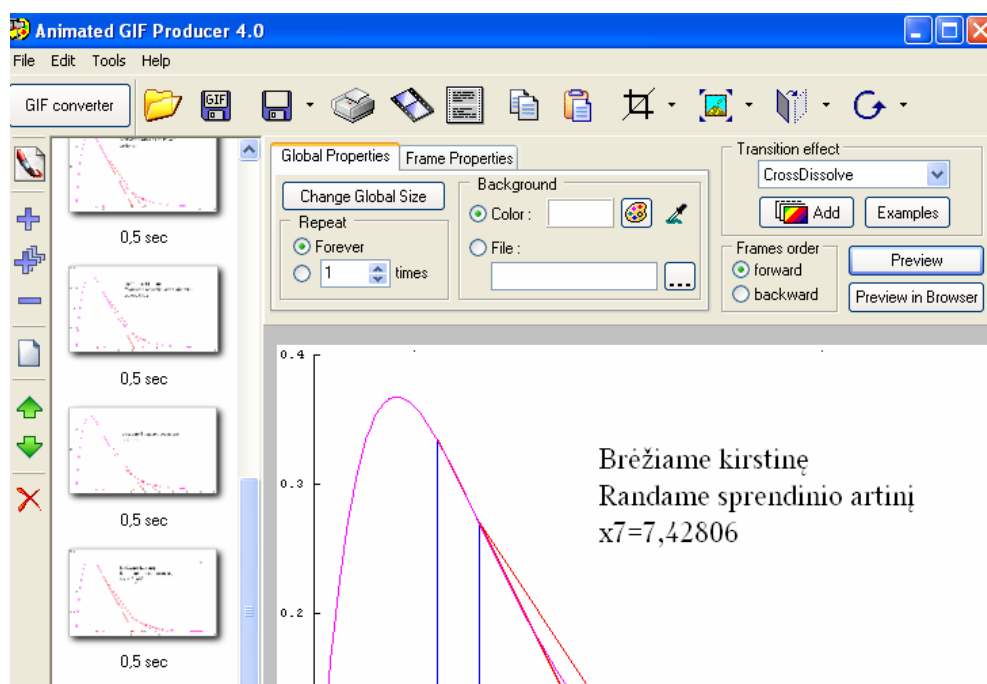
Sukurtoje interaktyviojoje svetainėje pateikiama teorinė medžiaga, vaizdinės priemonės, uždaviniai, testai bei laboratoriniai darbai.

Pasirinkus nuorodą „Metodai“, atveriamė kiekvienam metodui atskirai sukurtą puslapį.

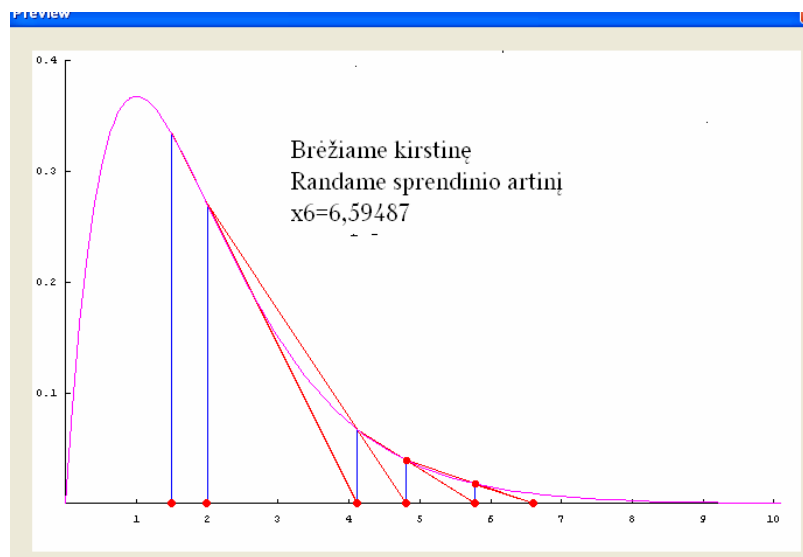


26 pav. Metodų aprašymo langas

Animacijos kūrimo programėle Animated GIF Producer sukurtas grafinis uždavinio sprendimo algoritmas:



27 pav. Programos kūrimo langas



28 pav. Animuotas vaizdas

### 5.7. Testai

Testų kūrimui buvo panaudota **Java Script** scenarijų kalba.

**Pažymėkite teisingą atsakymą**

Niutono metodas dar vadinamas

- liestinių metodu
- tiesės metodu
- sankirtos metodu
- stygų metodu

---

Duota netiesinė lygtis  $x^3 - x - 0,231 = 0$ . Pradinis artinys  $x_0 = 3$ . Antrasis artinys lygus

- 1,525
- 2,085
- 1,225

---

Niutono metodo konvergavimo greitis

- mažesnis nei pusiakirtos metodo
- didesnis nei pusiakirtos metodo
- toks pat, kaip pusiakirtos metodo
- didesnis nei kirstinių metodo

---

Niutono metodo algoritmo pirmas žingsnis

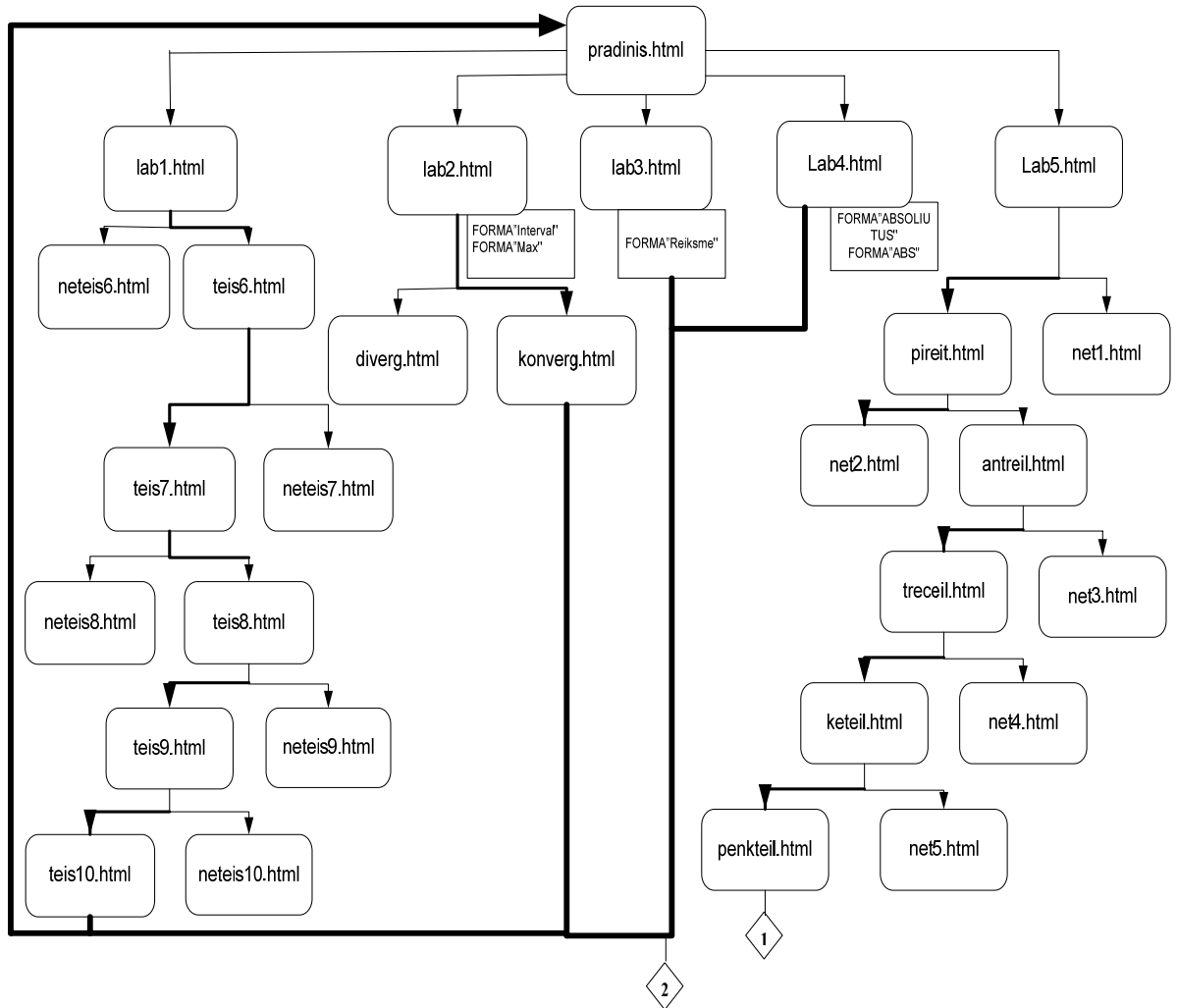
- Tikslumo tikrinimas

29 pav. Testų atlikimo langas

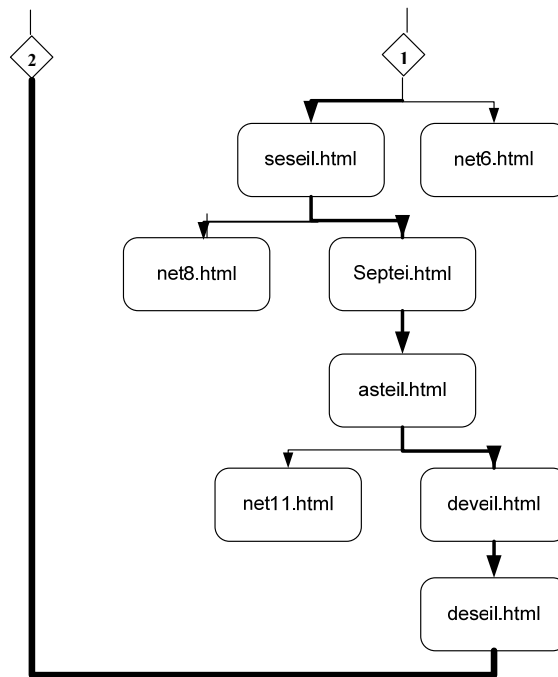
## 5.8. Savikontrolės užduotys

Puslapyje „Savikontrolė“ Studentams pateikiamos 5 savikontrolės užduotys, skirtos metodų įsisavinimui. Šioje mokomojoje priemonėje užduotys skirtos studentų praktinių įgūdžių, sprendžiant netiesines lygtis skaitiniais metodais, formavimui. Atliekant užduotis reikia gerai žinoti teorinę medžiagą.

30 pav. pateikta savikontrolės užduočių struktūrinė schema.



30 pav. Savikontrolės užduočių schema(1)



31 pav. Savikontrolės užduočių schema(2)

**1 užduotis**

Šioje užduotyje pirmame ekrane pateiktas dviejų funkcijų grafikas. Šių funkcijų susikirtimo taškas ir yra duotos netiesinės lygties sprendinys. Pirmiausia studentas turi teisingai pasirinkti intervalo, kuriame yra sprendinys, galų koordinates. Įvedus intervalo reikšmes, jos yra tikrinamos. Jei įvestos teisingos reikšmės, tai išvedamas pranešimas, kad reikšmės parinktos teisingai ir jis gali toliau atlikti darbą. Sekantis žingsnis būtų, maksimalios funkcijos reikšmės duotame intervale radimas. Tam kad būtų galima įvesti maksimalią funkcijos reikšmę į atitinkamą lauką, būtina ją apskaičiuoti pagal formulę, kurią studentai turi žinoti. Įvedus maksimalią reikšmę, ji tikrinama. Priklausomai nuo gautos maksimalios reikšmės, reikia parašyti atsakymą, ar iteracinis procesas konverguoja, ar diverguoja.

Paveikslėlyje pavaizduoti funkcijų  $y=x$  ir  $y=-2\ln x$  grafikai. Įrašykite intervalo, kuriame yra sprendinys, reikšmes. Pasirinkite sveikas, mažiausias intervalo reikšmes.

Pradinis taškas:  Galinis taškas:

patvirtinti atsakymą

Teisingai pasirinkę intervalo galų koordinates, apskaičiuokite funkcijos  $q(x)$  maksimalią reikšmę duotame intervale ir įrašykite į žemiau esantį laukelį.

Maksimali reikšmė:

patvirtinti atsakymą

32 pav. 1 užduotis

## 2 užduotis

Užduotyje reikia teisingai parinkti intervalo galų reikšmes. Intervalo reikšmių parinkimas priklauso nuo funkcijos ženklo intervalo vidurio taške. Jei funkcijos reikšmės ženklas vidurio taške yra neigiamas, tai vidurio taško reikšmė tampa intervalo pradine reikšme, o jei vidurio taško reikšmė yra teigiama, tai ji tampa intervalo galine reikšme. Procesas yra baigiamas, kai intervalo galų reikšmių absoliutinis dydis yra mažesnis už nustatytą absoliutinį dydį.

**Moksleivis sprendė uždavinį. Sprendimo eiga surašė į lentelę. Bet nežinia iš kur atsiradusi rašalo dėmė uždengė intervalo reikšmes. Padėkite moksleiviui įrašyti tinkamas intervalo galų reikšmes.**

n	$a_n(-)$	$b_n(+)$	$x_n = (a_n + b_n) / 2$	$f(x_n)$	$ b_n - a_n  / 2$
0	-1	0	-0,5	-1,331746	0,5
1	-0,5	0	-0,25	-0,213411	0,25
2	-0,25	0	-0,125	0,383473	0,125
3	-0,25	-0,125	-0,1875	0,082319	0,0625
4	-0,25	-0,1875	-0,21875	-0,066256	0,03125
5	-0,21875	-0,1875	-0,203125	0,007858	0,015625
6	-0,21875	-0,203125	-0,210938	-0,029243	0,007813
7	-0,207031	-0,203125	-0,207031	-0,010703	0,003906
8	-0,205078	-0,203125	-0,205078	-0,001425	0,001953
9	-0,204102	-0,203125	-0,204102	0,003216	0,000977
10	-0,204590	-0,203125	-0,204590	0,000895	0,000488

Sprendimas

Parinkite intervalo reikšmių porą, kai **n=6**:

GRĮŽTI

33pav. 2 užduotis

## 3 užduotis

Užduotis yra įrašyti lygties sprendinius tam tikru tikslumu. Studentai turi sugebėti iš lentelės atrinkti sprendinius. Sprendinio įvedimas yra tikrinamas. Todėl iš karto galima pasitikrinti atsakymą.

**Lygties  $4^x + 5x + 1 = 0$  sprendimas parodytas pateiktoje lentelėje.**

n	$a_n$	$b_n$	$c_n$	$y(c_n)$	$\varepsilon$
0	-1	0	-0,5	-1	0,5
1	-0,5	0	-0,25	0,457107	0,25
2	-0,5	-0,25	-0,375	-0,2804	0,125
3	-0,375	-0,25	-0,3125	0,08592	0,0625
4	-0,375	-0,3125	-0,34375	-0,09782	0,03125
5	-0,34375	-0,3125	-0,328125	-0,0061	0,015625
6	-0,328125	-0,3125	-0,320313	0,039873	0,007813
7	-0,328125	-0,3203125	-0,324219	0,016877	0,003906
8	-0,328125	-0,3242188	-0,326172	0,005386	0,001953
9	-0,328125	-0,3261719	-0,327148	-0,00036	0,000977

Užrašykite lygties sprendinį

GRĮŽTI

34pav. 3 užduotis



#### 4 užduotis

Šiame praktiniame darbe reikia surasti duotos lygties sprendinį, kai duotas pradinis sprendinio artinys. Kad studentas galėtų rasti sprendinį, jam pateikiamos dvi skaičiuoklės. Viena apskaičiuoja funkcijos reikšmę duotame taške. Kitoje skaičiuoklėje skaičiuojama naujai gauto sprendinio paklaida. Lentelė pildoma tol, kol paklaida neviršija vienos šimtosios. Apskaičiuavus paklaidą, galima patikrinti, ar teisingai buvote parinkę sprendinio artinius, ar teisingai gavote paklaidą. Užpildžius lentelę, reikia įrašyti sprendinį. Sprendinio reikšmė patikrinama.

Duota cheminės pusiausvyros lygtis  $-8x^3 + 11x^2 - 10x + 2 = 0$ .  
 Paprastųjų iteracijų metodu 0,01 tikslumu reikia rasti pusiausvyros koncentraciją, t.y. rasti  $x$  reikšmę, kai pradinis sprendinio artinys lygus 1.

**SPRENDIMAS**

Pirmoje lentelėje, įvedę sprendinio artinius galite surasti funkcijos reikšmę. Funkcijos reikšmė apskaičiuojama pagal formulę:  
 $x = (-8x^3 + 11x^2 + 2) / 10$   
 Veiksnius atlikite tol, kol rasite sprendinį.

Sprendinio artinys $x_n$	OK	Funkcijos reikšmė taške $x_n$
	<input type="button" value="OK"/>	

Sprendinio artiniai	Funkcijos reikšmės		$x_n - x_{n+1}$	Patikrink
		<input type="button" value="OK"/>		<input type="checkbox"/>
		<input type="button" value="OK"/>		<input type="checkbox"/>
		<input type="button" value="OK"/>		<input type="checkbox"/>
		<input type="button" value="OK"/>		<input type="checkbox"/>
		<input type="button" value="OK"/>		<input type="checkbox"/>
		<input type="button" value="OK"/>		<input type="checkbox"/>

Įrašykite sprendinį, pasirinkdami penkis skaičius po kablelio.

35 pav. 4 užduotis

#### 5 užduotis

Užduotyje pateikiami atskiri programos, parašytos MatLab aplinkoje fragmentai. Studentas pagal uždavinio sprendimo algoritmą turi teisingai parašyti programą.

**Lygties  $2x - \tan x = 0$  sprendinio paieška Niutono metodu 0,0001 tikslumu, kai pradinis artinys  $x_0 = 1,5$ .**  
**Užduotis:** žemiau pateikta sprendinio paieškos MatLab aplinkoje programos fragmentai. Iš pateiktų fragmentų, teisingai sudarykite programą ir sužinosite duotos lygties sprendinį norimu tikslumu.  
 (eilutes žymėkite tokia tvarka, kokia tvarka jos turėtų būti surašytos programoje)

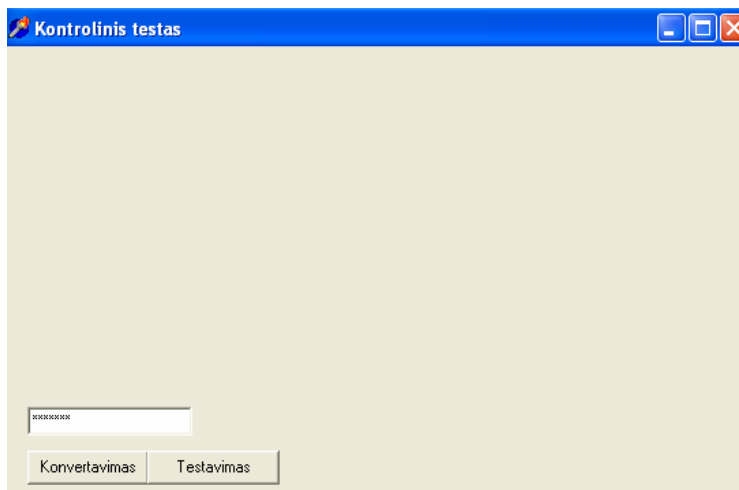
Programos fragmentai	Teisingai parašyta programa
<pre>format short; x0=x1; x0=1.5; disp(['u ' num2str(u)]); x1=x0-(2*x0-tan(x0))/(2-1/(cos(x0))^2); u=abs(x0-x1); while abs(x0-x1)&gt;0.0001     disp(x1);     x1=x0-(2*x0-tan(x0))/(2-1/(cos(x0))^2); end;</pre>	

36 pav. 5 užduotis

## 5.9. Kontrolinis testas

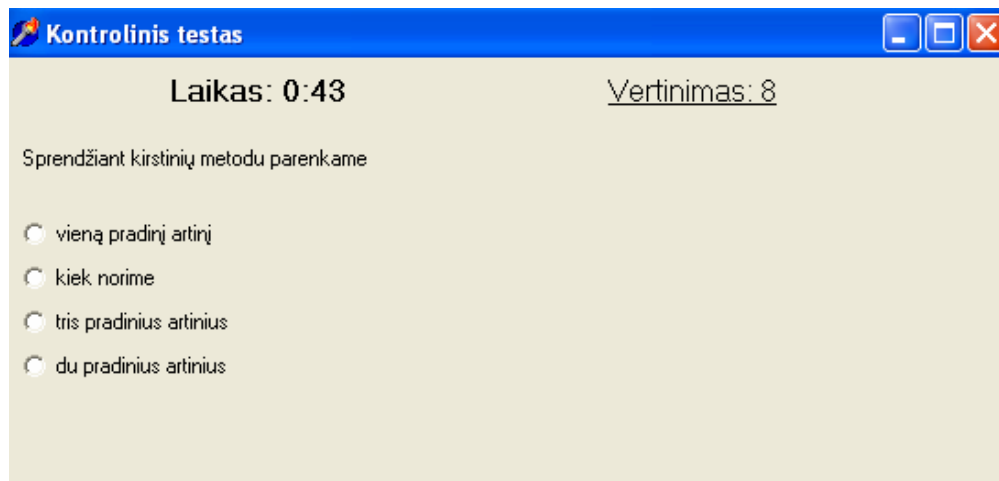
Kontrolinis testas sukurtas Delphi programa. Programos kodas 2 priede. Studentui pateikiama 10 sugeneruotų klausimų. Atlikus testą, iš karto parašomas įvertinimas.

Kadangi tai yra kontrolinis testas, tai tekstinio failo konvertavimą ir testo paleidimą atlieka dėstytojas, prieš tai įvedęs slaptažodį.



37 pav. Pradinis testo langas

Kiekvienam klausimo atsakymui skirta 1 minutė. Jei per tą laiką klausimas neatsakomas, jis įvertinamas kaip neatsakytas. Jei klausimas atsakomas greičiau nei per minutę, laiko limitas pereina kito klausimo atsakymui.

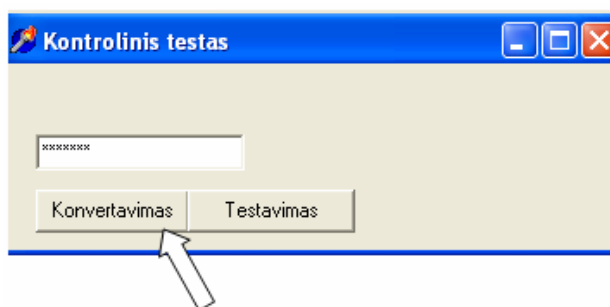


38 pav. Testo vertinimo langas

### 5.10. Instrukcija svetainės naudotojui

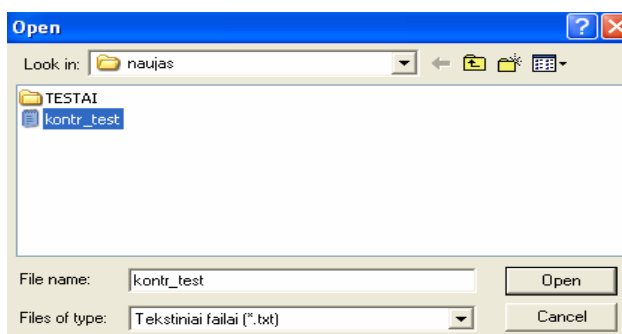
Svetainės patalpinta adresu <http://www.kvtk.lt/~vilma/>. Vartotojo sąsaja paprasta, neperkrauta valdymo elementais. Norimus duomenis galima peržvelgti pasirinkus norimas nuorodas meniu. Programa yra paprasta vartoti, todėl net ir mokytojui, nepakankamai gerai išmanančiam informacines technologijas, nereikės jaustis nesmagiai.

Norint atlikti kontrolinį testą, dėstytojas turi įvesti slaptažodį (šiuo atveju “vilmaja”). Įvedus slaptažodį reikia konvertuoti tekstinį failą kontr.txt.



39 pav. Pradinis kontrolinės užduoties langas

Atsivėrusiame lange pasirinkti tekstinį failą, kuriame surašyti klausimai ir atsakymų variantai.



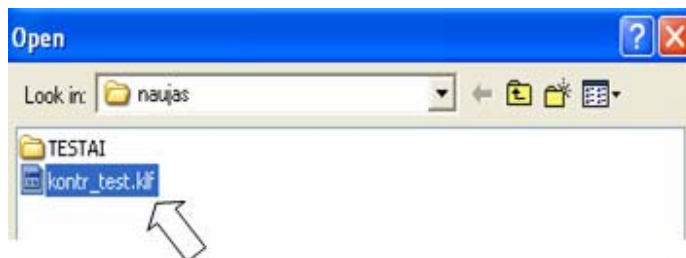
40 pav. Testo pasirinkimo langas

Tekstiniame faile neturi būti tuščių eilučių. Jei bus nors viena tuščia eilutė, bus rodomas pranešimas.



41 pav. Klaidų pranešimo langas

Pranešimas reiškia, kad antroji tekstinio failo eilutė yra tuščia. Jei tekstinis failas teisingai paruoštas, po konvertavimo grįžtama į pradinį programos langą. Įvedamas tas pats slaptažodis ir pasirenkame testavimą. Atsivėrusiame lange pasirenkame failą su priedvardžiu „klf“



**42 pav. Failo pasirinkimas**

Pradedama atsakinėti į testo klausimus. Baigus testą, parašomas įvertinimas.

Tekstinis failas įdėtas tuo pačiu adresu, kaip ir svetainė. Šį failą reikia nusikopijuoti į savo kompiuterį ir tada pagal aukščiau aprašytus nurodymus galima atlikti kontrolinį testą. Šį testą dėstytojas gali bet kada keisti, redaguoti.

Sukurta kompiuterinė mokomoji priemonė gali būti mobili. Kiekvienas studentas galės ją turėti bet kurioje atminties laikmenoje.

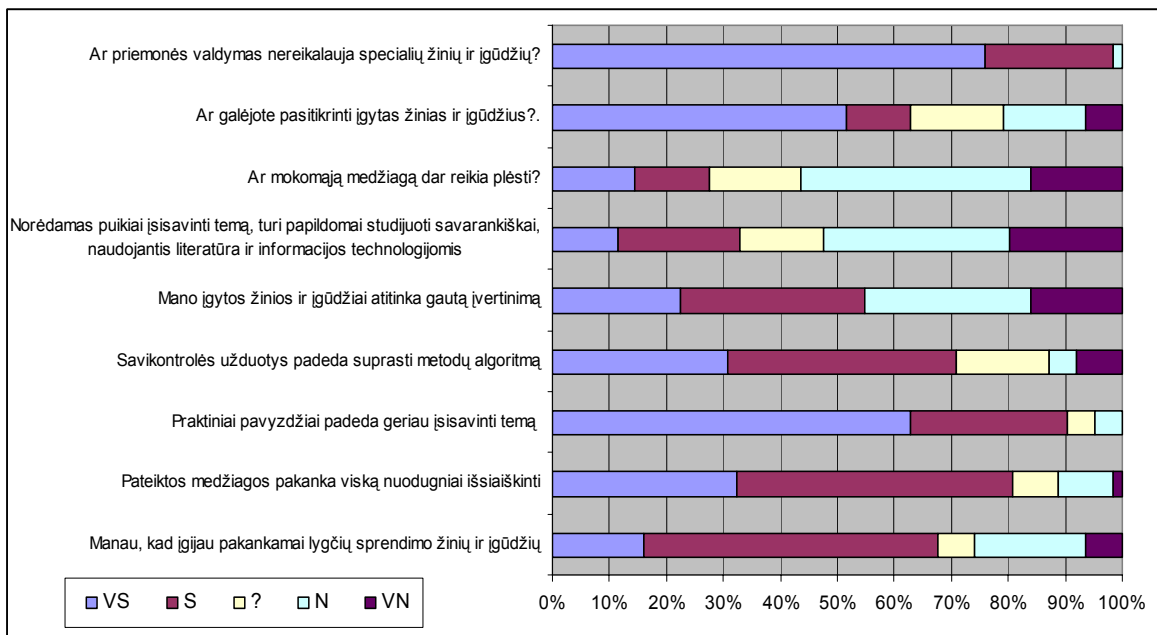
## 6. *STUDENTŲ KOMPIUTERINĖS PRIEMONĖS VERTINIMAS*

Sukurta internetinę svetainę vertino studentai. Svetainės vertinimui buvo apklausti 62 studentai. Studentams pateikti klausimai 2 darbo priede. Apklauskos rezultatai apibendrinti 43 lentelėje ir 44 paveiksle:

1. Visiškai sutinkate (VS)
2. Sutinkate su tam tikromis sąlygomis (S)
3. Negalite nuspręsti arba neturite savo nuomonės šiuo klausimu (?)
4. Esate linkęs(usi) nesutikti (N)
5. Visiškai nesutinkate (VN)

**43 lenelė. Studentų vertinimo rezultatai**

	VS	S	?	N	VN
Manau, kad įgijau pakankamai lygčių sprendimo žinių ir įgūdžių	10	32	4	12	4
Pateiktos medžiagos pakanka viską nuodugnai išsiaiškinti	20	30	5	6	1
Praktiniai pavyzdžiai padeda geriau įsisavinti temą	39	17	3	3	0
Savikontrolės užduotys padeda suprasti metodų algoritmą	19	25	10	3	5
Mano įgytos žinios ir įgūdžiai atitinka gautą įvertinimą	14	20	0	18	10
Norėdamas puikiai įsisavinti temą, turi papildomai studijuoti savarankiškai, naudojantis literatūra ir informacijos technologijomis	7	13	9	20	12
Ar mokomąją medžiagą dar reikia plėsti?	9	8	10	25	10
Ar galėjote pasitikrinti įgytas žinias ir įgūdžius?.	32	7	10	9	4
Ar priemonės valdymas nereikalauja specialių žinių ir įgūdžių?	47	14	0	1	0



44 pav. Studentų vertinimo procentinis pasiskirstymas

Iš apklauso rezultatų matosi, kad kompiuterių panaudojimas daro teigiamą įtaką mokinių skaitinių metodų mokymosi motyvacijai. Visuose keltuose klausimuose teigiamų atsakymų daugiau negu neigiamų. Naudojant kompiuterines priemones, dalykas tampa naudingesnis, lengviau suprantamas, savikontrolės užduotys leidžia patikrinti įgytas žinias ir įgūdžius, beveik nebereikia papildomos literatūros temų studijavimui.

### Apklausos apibendrinimas

Apžvelgus tyrimų rezultatus galima teigti, kad kompiuterių panaudojimas skaitinių metodų mokymo procese padeda dėstytojams pasiekti užsibrėžtus tikslus:

- Leidžia vaizdžiau ir dinamiškiau pateikti netiesinių lygčių sprendimo metodus,
- Padeda ugdyti studentų savarankiško darbo įgūdžius,
- Skatina studentų mokymosi motyvaciją, domėjimąsi skaičiuojamąja matematika;
- Ugdo, įtvirtina studentų darbo su kompiuteriu ir naujomis technologijomis gebėjimus bei įgūdžius.

## IŠVADOS

- Pateikta netiesinių lygčių sprendimo metodų realizacija trijose kompiuterinėse programose ir atliktas sprendimo kompiuterinėse programose palyginimas.
  - Išnagrinėjus ir palyginus trijų programų– Excel, Matlab ir Mathcad galimybes spręsti netiesines lygtis skaitiniais metodais, nustatyta, kad šių sistemų galimybes skiriasi. Pagal patogumą ir efektyvumą patogiausia yra Matlab aplinka:
    - patogiausia nustatyti šaknų izoliacijos intervalus,
    - patogiu keisti funkcijos grafiko parametrus,
    - realizuotos paelementės operacijos .
    - funkcija `ginput(k)` gražina pažymėtų taškų (taškų skaičius  $k$ ) koordinates.
    - patogi vartotojo sąsaja.
  - Ieškant daugianario šaknų MatLab ir Mathcad galimybės panašios. MatLab aplinkoje naudojama funkcija `roots`. Pakanka teisingai užrašyti daugianario koeficientus. Mathcad aplinkoje daugianario šaknims rasti naudojamas simbolinio skaičiavimo įrankis `solve` ir vidinė funkcija `polyroots`
  - Sukurta kompiuterinė mokomoji priemonė netiesinėms lygtims spręsti: pateikta teorinė medžiaga, kontrolinis testas, savikontrolės užduotys, vaizdinė priemonė, skirtos nagrinėtų skaitinių metodų geometrinei interpretacijai.
  - Atliktas eksperimentinis priemonės tyrimas. Pagal studentų apklausos rezultatus nustatyta, kad kompiuterinė priemonė padeda geriau suprasti ir įsisavinti netiesinių lygčių sprendimo metodus, savikontrolės užduotys įtvirtina uždavinių sprendimo įgūdžius, įtvirtina studentų darbo su kompiuteriu ir naujomis technologijomis gebėjimus.
- Kompiuterinė mokomoji programa gali būti naudojama įvairiais mokymo(si) tikslais: aiškinant naują temą; kaip papildoma vaizdinė priemonė, kartojant, įtvirtinant mokymosi medžiagą, įvertinant žinių įsisavinimą.

## LITERATŪRA

- [1] Būda V., Čiegis R. Skaičiuojamosios matematikos įvadas. Leidybos centras, 1995.
- [2] Čiegis R., Būda V. Skaičiuojamoji matematika: vadovėlis V.: TEV, 1997.
- [3] Daunoras J. Dinaminių sistemų modeliavimas naudojant MatLAB/Simulink. K.: Technologija, 2001.
- [4] Dosinas G. ir kt. Skaitiniai metodai. K.: Technologija, 2000.
- [5] Дьяконов В.П. MATLAB 6.5 SP1/7.0 Simulink 5/6. Основы применения. М.: Солон – пресс, 2005. p. 326-330.
- [6] Janušauskaitė S., Furmonavičienė Z.// Studentų motyvacijos mokytis matematiką tyrimas, Matematika ir matematikos dėstymas: Konferencijos pranešimų medžiaga [Kaunas, 1997]. Kaunas, 1997, p. 23-27.
- [7] Krylovas A., Vilkelis R.// Pirmo kurso studentų mokyklinės matematikos žinios, Lietuvos mokslas ir pramonė, Matematika ir Matematikos dėstymas – 2001: Konferencijos pranešimų medžiaga [Kaunas, 2001]. Kaunas, p.30-34.
- [8] Lipeikienė J. Matematika ir kompiuteris. Vilnius. 2002.
- [9] Markauskaitė L. Kaip įvertinti kompiuterinę mokomąją programą // Kompiuterizuotas mokymas Lietuvoje konferencijos darbai, 1996. p 65–74.
- [10] Plukas K. Skaičiavimo metodai ir algoritmai. I dalis. K.: Technologija, 1995.
- [11] Plukas K. Skaičiavimo metodai ir algoritmai. II dalis. K.: Technologija, 1997.
- [12] Самарский А. А., Гулин А. В. Численные методы. Учебное пособие. М.: Наука, 1989.
- [13] Turskienė S. Uždavinių sprendimas kompiuterinėmis matematikos sistemomis. Šiauliai 2003.
- [14] Turskienė S. Kompiuterinių matematikos sistemų programavimo galimybės. Šiauliai, 2004.
- [15] Vidžiūnas A. Delphi 5. Programavimas ir vaizdiniai komponentai. K.: Smaltija, 2001.
- [16] Вержбицкий В. М. Линейная алгебра и нелинейные уравнения. М.: МониКС 21 век, 2005
- [17] Вержбицкий В. М. Математический анализ и обыкновенные дифференциальные уравнения. М.: МониКС 21 век, 2005.
- [18] Vivet M. Evaluation of Teaching Material, Calisce, 1996.



- [19] Wibe J. Course Evaluation, COPERNICUS PROJECT 1445'FLEXIBLE & DISTANT LEARNING THROUGH TELEMATIC NETWORKS'/ Wibe J., Jennings Ch., 1997.
- [20] Dagienė V. Informacinių technologijų taikymo švietime konceptualusis pagrindimas[interaktyvus]. 2003, sausis [žiūrėta 2008m. gegužės 15d.] Prieiga per internetą: <http://www.leidykla.vu.lt/inetleid/inf-mok/25/str12.html>.

## ***1 Priedas. Anketa***

Šis klausimynas skirtas išaiškinti studentų nuomonę apie svetainę.

Pažymėkite kiekvieną teiginį, geriausiai išreiškiantį Jūsų nuomonę. Anketos atsakymai bus naudojami tik magistriniame darbe, o rezultatai publikuojami tik statistiškai.

1. Visiškai sutinkate (VS)
2. Sutinkate su tam tikromis sąlygomis (S)
3. Negalite nuspręsti arba neturite savo nuomonės šiuo klausimu (?)
4. Esate linkę(usi) nesutikti (N)
5. Visiškai nesutinkate (VN)

Prašome atsakyti į visus klausimus.

	VS	S	?	N	VN
Manau, kad įgijau pakankamai lygčių sprendimo žinių ir įgūdžių.....					
Pateiktos medžiagos pakanka viską nuodugnai išsiaiškinti.....					
Praktiniai pavyzdžiai padeda geriau įsisavinti temą.....					
Savikontrolės užduotys padeda suprasti metodų algoritmą.....					
Mano įgytos žinios ir įgūdžiai atitinka gautą įvertinimą.....					
Norėdamas puikiai įsisavinti temą, turi papildomai studijuoti savarankiškai, naudojantis literatūra ir informacijos technologijomis.....					
Ar mokomąją medžiagą dar reikia plėsti?.....					
Ar galėjote patikrinti įgytas žinias ir įgūdžius?.....					
Ar priemonės valdymas nereikalauja specialių žinių ir įgūdžių?.....					

## ***2 Priedas.Delphi programos kodas***

```

unit Testas;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms,
  Dialogs,
  StdCtrls, ExtCtrls;

type
  TForm1 = class(TForm)
    OpenDialog1: TOpenDialog;
    Label1: TLabel;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    RadioButton3: TRadioButton;
    RadioButton4: TRadioButton;
    RadioButton5: TRadioButton;
    RadioButton6: TRadioButton;
    RadioButton7: TRadioButton;
    Button1: TButton;
    Button2: TButton;
  end;
  
```

```

    Timer1: TTimer;
    Label2: TLabel;
    Edit1: TEdit;
    Label3: TLabel;
    Label4: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure RadioButton1Click(Sender: TObject);
    procedure RadioButton2Click(Sender: TObject);
    procedure RadioButton3Click(Sender: TObject);
    procedure RadioButton4Click(Sender: TObject);
    procedure RadioButton5Click(Sender: TObject);
    procedure RadioButton6Click(Sender: TObject);
    procedure RadioButton7Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;
    Skai : integer;
    Ap : boolean;
    Visas_laikas : integer;
    Duotas : integer;

implementation

{$R *.DFM}

Uses
    Klausimai;

procedure TForm1.Button1Click(Sender: TObject);
begin
    Form1.Label4.Visible := false;
    if Paslaptis then
        Formavimas
    else
        Form1.Edit1.Visible := true
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
    Form1.Label4.Visible := false;
    if Paslaptis then
        Pradzia
    else
        Form1.Edit1.Visible := true
end;

procedure TForm1.RadioButton1Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 10 then

```

```

        Duotas := 10;
        Vertinimo_organizavimas
    end;

procedure TForm1.RadioButton2Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.RadioButton3Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.RadioButton4Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.RadioButton5Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.RadioButton6Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.RadioButton7Click(Sender: TObject);
begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
        Duotas := 60;
    Vertinimo_organizavimas;
end;

procedure TForm1.FormCreate(Sender: TObject);

```

```

begin
  Klausimas := NIL;
  Form1.Label2.Caption := '';
  Pradinis_vaizdas;
  Form1.Edit1.Text := '';
  Form1.Edit1.PasswordChar := '*';
  Skai := 0;
  Form1.Label3.Visible := false;
  Form1.Label4.Visible := false;
  Ap := false
end;

procedure TForm1.Timer1Timer(Sender: TObject);
var
  Min, Sek : integer;
  Nulis : string;
begin
  Skai := Skai + 1;
  if Ap and (Skai > Duotas) then begin
    Visas_laikas := Visas_laikas - Skai;
    Duotas := Trunc(Visas_laikas/Kiekis) + 1;
    If Duotas > 60 then
      Duotas := 60;
    Vertinimo_organizavimas
  end;
  Min := (Duotas - Skai) div 60;
  Sek := (Duotas - Skai) mod 60;
  if Sek div 10 > 0 then
    Nulis := ' '
  else
    Nulis := '0';
  Form1.Label3.Caption := 'Laikas: ' + IntToStr(Min) + ':' + Nulis +
IntToStr(Sek)
end;

end.

unit Klausimai;

interface

type

  sa = ^ka;
  ka = record
    pas : string[100];
    tei : boolean;
    ats : sa
  end;

  sk = ^kk;
  kk = record
    kla : string[100];
    kie : integer;
    ats : sa;
    buv : sk;
    sek : sk
  end;

```

```

var

    Klausimas : sk;
    Kiekis : integer;
    f : TextFile;
    fa : file of ka;
    fk : file of kk;
    kl : integer;
    k : TextFile;
    Visi : integer;
    Teisingi : integer;
    Atsakyti : integer;
    PrLaikas : integer;
    GaLaikas : integer;

Procedure Pradzia;
Procedure Formavimas;
Procedure Vertinimo_organizavimas;
Procedure Pradinis_vaizdas;
Function Paslaptis : boolean;

implementation

uses
    Testas, Dialogs, SysUtils;

{-----Slaptazdzio patikrinimas-----}
Function Paslaptis : boolean;
var
    Eilute : string;
begin
    Form1.Edit1.Visible := False;
    Eilute := Form1.Edit1.Text;

    if Form1.Edit1.Text = 'vilmaja' then
        Paslaptis := true
    else
        Paslaptis := false;
    Form1.Edit1.Text := ''
end;

{-----Pradinis vaizdas-----}
Procedure Pradinis_vaizdas;
begin
    Form1.Label1.Caption := '';
    Form1.RadioButton1.Checked := False;
    Form1.RadioButton1.Visible := False;
    Form1.RadioButton2.Checked := False;
    Form1.RadioButton2.Visible := False;
    Form1.RadioButton3.Checked := False;
    Form1.RadioButton3.Visible := False;
    Form1.RadioButton4.Checked := False;
    Form1.RadioButton4.Visible := False;
    Form1.RadioButton5.Checked := False;
    Form1.RadioButton5.Visible := False;
    Form1.RadioButton6.Checked := False;
    Form1.RadioButton6.Visible := False;
    Form1.RadioButton7.Checked := False;
    Form1.RadioButton7.Visible := False;

```

```

end;

{-----Tekstiniu failu tikrinimas-----}
Function Tikrinimas (var f : TextFile) : boolean;
var
  S : string;
  Teis, Klai : integer;
  Prad, Tikr : boolean;
  Verdiktas : string;
  Eilute : integer;
begin
  Reset (f);
  Teis := 0;
  Klai := 0;
  Prad := true;
  Tikr := true;
  Eilute := 0;
  While not EoF(f) and Tikr do begin
    ReadLn (f, S);
    Eilute := Eilute + 1;
    if Length(S) > 0 then
      case S[1] of
        '#' : if Prad then
              Prad := false
            else
              if (Teis = 1) and (Klai >= 1) and (Klai < 7) then begin
                Teis := 0;
                Klai := 0;
              end
            else begin
              Tikr := false;
              Verdiktas := 'Teisingi ir net. atsakymai' +
                IntToStr(Eilute)
            end;
        '+' : if Prad then begin
              Tikr := false;
              Verdiktas := 'Pirmas simbolis +' + IntToStr(Eilute)
            end
            else
              Teis := Teis + 1;
        '-' : if Prad then begin
              Tikr := false;
              Verdiktas := 'Pirmas simbolis -' + IntToStr(Eilute)
            end
            else
              Klai := Klai + 1;
        else begin
              Tikr := false;
              Verdiktas := 'Kitoks simbolis' + IntToStr(Eilute)
            end
            end
          else begin
            Tikr := false;
            Verdiktas := 'Tuščia eilutė' + IntToStr(Eilute)
          end
        end;
    if (Teis = 1) and (Klai >= 1) and (Klai < 7) then
      else begin
        Tikr := false;

```

```

    Verdiktas := 'Teisingi ir net. atsakymai' + IntToStr(Eilute)
end;
if Tikr then
else begin
    Form1.Label4.Visible := true;
    Form1.Label4.Caption := Verdiktas
end;
Tikrinimas := Tikr
end;

{-----Strukturiniu failu formavimas-----}
Procedure Formavimas;
var
    S : string;
    Vardas : string;
    Atsakymas : sa;
    Pirmas : boolean;
begin
    Form1.OpenDialog1.Filter := 'Tekstiniai failai (*.txt)|*.TXT';
    if Form1.OpenDialog1.Execute then begin
        AssignFile (f, Form1.OpenDialog1.FileName);
        Vardas := Form1.OpenDialog1.FileName;
        if Tikrinimas (f) then begin
            Reset (f);
            AssignFile (fk, Copy(Vardas, 1, Length(Vardas)-3) + 'klf');
            Rewrite (fk);
            AssignFile (fa, Copy(Vardas, 1, Length(Vardas)-3) + 'atf');
            Rewrite (fa);
            New(Klausimas);
            New(Atsakymas);
            Pirmas := true;
            While not EOF(f) do begin
                ReadLn (f, S);
                if S[1] = '#' then begin
                    if Pirmas then
                        Pirmas := false
                    else
                        Write (fk, Klausimas^);
                    Klausimas^.kla := Copy(S, 2, Length(S) - 1);
                    Klausimas^.kie := 0
                end;
                if (S[1] = '-') or (S[1] = '+') then begin
                    Atsakymas^.pas := Copy(S, 2, Length(S) - 1);
                    if S[1] = '+' then
                        Atsakymas^.tei := true
                    else
                        Atsakymas^.tei := false;
                    Klausimas^.kie := Klausimas^.kie + 1;
                    Write (fa, Atsakymas^)
                end
            end;
            Write (fk, Klausimas^);
            CloseFile (fk);
            CloseFile (fa)
        end;
        CloseFile (f);
    end
end;
end;

```



```

{-----Klausimas pateikimas formoje-----}
Procedure Pateikimas;
var
  Atsakymas : sa;
  Ats_Kiekis : integer;
begin
  Form1.Label1.Caption := Klausimas^.kla;
  Atsakymas := Klausimas^.ats;
  Ats_Kiekis := Klausimas^.kie;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton1.Visible := True;
    Form1.RadioButton1.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton1.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton2.Visible := True;
    Form1.RadioButton2.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton2.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton3.Visible := True;
    Form1.RadioButton3.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton3.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton4.Visible := True;
    Form1.RadioButton4.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton4.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton5.Visible := True;
    Form1.RadioButton5.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton5.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton6.Visible := True;
    Form1.RadioButton6.Caption := Atsakymas^.pas;
    Ats_Kiekis := Ats_Kiekis - 1;
    Atsakymas := Atsakymas^.ats
  end
  else
    Form1.RadioButton6.Visible := False;
  if Ats_Kiekis <> 0 then begin
    Form1.RadioButton7.Visible := True;

```

```

        Form1.RadioButton7.Caption := Atsakymas^.pas;
    end
    else
        Form1.RadioButton7.Visible := False;
        Ap := true;
        Form1.Label3.Caption := '';
        Form1.Label3.Visible := true;
        Skai := 0
    end;

{-----Is rato pasalinamas klausimas-----}
Procedure Pasalinimas;
begin
    if Kiekis > 1 then begin
        Klausimas := Klausimas^.buv;
        Klausimas^.sek^.sek^.buv := Klausimas;
        Klausimas^.sek := Klausimas^.sek^.sek
    end
end;

{-----Atliekamas vertinimas-----}
Function Vertinimas(Atsakymas : sa) : boolean;
var
    Teisingas : boolean;
    Ats_Kiekis : integer;
begin
    Teisingas := False;
    Ats_Kiekis := Klausimas^.kie;
    if Ats_Kiekis > 0 then begin
        Teisingas := Atsakymas^.tei And Form1.RadioButton1.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then begin
        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton2.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then begin
        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton3.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then begin
        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton4.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then begin
        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton5.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then begin

```

```

        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton6.Checked;
        Atsakymas := Atsakymas^.ats;
        Ats_Kiekis := Ats_Kiekis - 1
    end;
    if Ats_Kiekis > 0 then
        Teisingas := Teisingas Or Atsakymas^.tei And
Form1.RadioButton7.Checked;
        Vertinimas := Teisingas
    end;

    {-----Rezultatu pateikimas-----}
    Procedure Pabaiga;
    begin
        Form1.Label2.Caption := 'Vertinimas: ' +
IntToStr(Trunc(Teisingi/Atsakyti*10 + 0.45))
    end;

    {-----Testavimo organizavimas-----}
    Procedure Testavimas;
    var
        Zingsnis, i : integer;
    begin
        Randomize;
        if Kiekis <> 0 then begin
            Zingsnis := Trunc(Random(Kiekis));
            For i := 1 to Zingsnis do
                Klausimas := Klausimas^.sek;
                Pateikimas;
            end
        end;
    end;

    {-----Vertinimo organizavimas-----}
    Procedure Vertinimo_organizavimas;
    begin
        Atsakyti := Atsakyti + 1;
        If Vertinimas(Klausimas^.ats) then
            Teisingi := Teisingi + 1;
        Pabaiga;
        Pasalinimas;
        Kiekis := Kiekis - 1;
        Pradinis_vaizdas;
        if Kiekis > 0 then begin
            Testavimas;
            Pabaiga
        end
        else begin
            Form1.Label3.Visible := false;
            Ap := false
        end
    end;

    {-----Strukturiniu failu formavimas-----}
    Procedure Pradzia;
    var
        Vardas: string;
        Ein: sk;
        Atsakymas : sa;
        i : integer;

```

```

begin
  Form1.OpenDialog1.Filter := 'Failas su klausimais (*.klf)|*.KLF';
  if Form1.OpenDialog1.Execute then begin
    AssignFile (fk, Form1.OpenDialog1.FileName);
    Vardas := Form1.OpenDialog1.FileName;
    Reset (fk);
    AssignFile (fa, Copy(Vardas, 1, Length(Vardas)-3) + 'atf');
    {$I-}
    Reset (fa);
    {$I+}
    if IOResult = 0 then begin
      Kiekis := 0;
      While not EOF(fk) do begin
        New(Ein);
        Read (fk, Ein^);
        Kiekis := Kiekis + 1;
        if Kiekis = 1 then begin
          Klausimas := Ein;
          Klausimas^.sek := Klausimas;
          Klausimas^.buv := Klausimas
        end
        else begin
          Ein^.sek := Klausimas^.sek;
          Klausimas^.sek := Ein;
          Ein^.buv := Klausimas;
          Ein^.sek^.buv := Ein;
          Klausimas := Ein
        end;
        New(Klausimas^.ats);
        Read (fa, Klausimas^.ats^);
        Atsakymas := Klausimas^.ats;
        For i := 2 to Klausimas^.kie do begin
          New(Atsakymas^.ats);
          Atsakymas := Atsakymas^.ats;
          Read (fa, Atsakymas^);
        end
      end;
      CloseFile (fk);
      CloseFile (fa);
      Visi := Kiekis;
      Teisingi := 0;
      Atsakyti := 0;
      Form1.Button1.Visible := false;
      Form1.Button2.Visible := false;
      Kiekis := 10;
      Visas_laikas := Kiekis * 30;
      Duotas := Trunc(Visas_laikas/Kiekis) + 1;
      Testavimas
    end
    else begin
      Form1.Label4.Visible := True;
      Form1.Label4.Caption := 'Truksta Failo'
    end
  end
end
end;

end.

```

### 3 priedas. Užduotys

#### Netiesinės lygties sprendimas iteraciniais metodais

**Darbo tikslas:** išspręsti lygtį taikant iteracinius metodus.

**Užduotis:**

1. Pasirinkite iš lentelės dėstytojo nurodytą lygtį.
2. Raskite lygties šaknis:  
Niutono metodu,  
Kirstinių metodu.
3. Užduotį atlikite MS Excel ir Matlab programa.
4. Palyginkite skirtingais metodais ir skirtingomis programomis gautus rezultatus.

**Darbo eigos etapai:**

Nubraižykite funkcijos  $f(x)$  grafiką.

2. Pasirinkite vieną šaknį ir nustatykite jos intervalą.
3. Sukonstruokite iteracines sekas kirstinių ir Niutono metodais.
4. Suraskite šaknį Matlab ir MS Excel programa.
5. Nustatykite, kuris metodas greičiau artėja (konverguoja) prie tikrosios reikšmės, kaip tai priklauso nuo pradinio taško parinkimo. Pavaizduokite skirtingų metodų konvergavimą grafiškai.

1	$e^x + 2x - 3 = 0$	2	$3^x + 5x - 3 = 0$
3	$4^x + 5x + 1 = 0$	4	$1,5^x + 2x - 3 = 0$
5	$0,5^x + (x - 3)^2 - 1 = 0$	6	$2^x + 5x^2 - 3 = 0$

### 4 priedas. Ruošiniai lygčių sprendimui Matlab aplinkoje

% sprendinio paieška pusiaukirtos metodu	% sprendinio paieška Niutono metodu	% funkcijos reikšmių lentelės sudarymas
<pre>function psk % Išspręsti lygtį x^3 - cosx=0 % Įvedam funkciją f(x) f = inline('x.^3 - cos(x)'); root1 = bisec(f, -0.6, -0.4) root2 = bisec(f, -0.2, 0.2) function c = bisec(f, a, b) while b - a &gt; eps c = (a - b) / 2 + b; if f(c) * f(a) &gt; 0 a = c; else b = c; end end</pre>	<pre>clear; x0=1,5; x1=x0-(2*x0-tan(x0))/(2-1/(cos(x0))^2); u=abs(x0-x1); disp(['u ' num2str(u)]); while abs(x0-x1) &gt; 0,0001 x0=x1; x1=x0-(2*x0-tan(x0))/(2-1/(cos(x0))^2); end; format short; disp(x1);</pre>	<pre>clear; a=1; x0=-5; xn=7; xh=0.5; xv=x0:xh:xn; [m,n]=size(xv); for i=1:n yv(i)=fy(a,xv(i)); end; disp(' Funkcijos reikšmių lentelė'); disp(' x y'); [xv' yv'] %grafiko formavimas xv=x0:xh/10:xn; [m,n]=size(xv); for i=1:n yv(i)=fy(a,xv(i)); end; plot(xv,yv);</pre>

## 5 priedas. Darbo Matlab aplinkoje demonstracinis filmukas

Su vaizdų įrašymo programą BB FlashBack sukurtas demonstracinis filmukas. Šiame filme demonstruojam, kaip susikurti M-failus, kaip paleisti juos vykdymui.

