



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
KOMPIUTERIŲ TINKLŲ KATEDRA**

Vytautas Boska

**ALGORITMAVIMO PRAKTIKOS MOKOMOSIOS
PRIEMONĖS SUDARYMAS IR TYRIMAS**

Magistro darbas

Recenzentas

prof. Rimantas Plėštys

2008-05-19

Vadovas

doc. dr. Kazys Baniulis

2008-05-19

Atliko

IFT-6 gr. stud. Vytautas Boska

2008-05-19

Kaunas 2008

SUMMARY

One of the most complicated teaching program parts of the information technologies is programming. The aims of teaching to program include not only transfer of information, but are close-knitted with training of critical thinking as well. Schoolchildren's low level of structural thinking handicaps understanding of algorithmization, schoolchildren's learning motivation diminishes. Teachers can apply modern teaching methods and use paperless educational programs and tests expediently in order to seek for teaching quality.

The main aim of this paper is to analyze the usage possibilities of active technologies in the programming lessons, to create easily soaking up and controlled computerized teaching tool.

Having done the analysis of computerized teaching tools created to do algorithmization exercises and used nowadays in Lithuanian schools, it could be stated that there is no Lithuanian computerized teaching tool letting to seek for comprehensive algorithmization teaching (learning) goals.

Made-up resource for learning "Algorithm" is created for senior pupils, teachers and first year students. The main attention is designed for logical structural exposition, visualization of material, teaching (learning) interactivity of themes in this tool. TestTool program which is create as tests of three different complexity levels will help to implement pupils' self-evaluation, self-reflection.

For widening of usage possibilities in created program, the final product is placed in Web server.

TURINYS

1. ĮVADAS.....	7
2. ANALITINĖ DALIS.....	9
2.1. MOKOMŪJŲ KOMPIUTERINIŲ PROGRAMŲ VERTINIMO KRITERIJAI	9
2.2. MOKOMŪJŲ KOMPIUTERINIŲ PRIEMONIŲ PASIRINKIMAS.....	11
2.3. MOKOMŪJŲ PROGRAMŲ ANALIZĖ.....	14
3. PROJEK TINĖ DALIS	22
3.1. MKP „ALGORITMAI“ PASKIRTIS IR PAGRINDIMAS	22
3.2. PRIEMONĖS STRUKTŪRA, REIKALAVIMAI, VARTOTOJAI	23
3.3. MKP REALIZACIJOS TECHNINIŲ PRIEMONIŲ PARINKIMAS IR APRAŠYMAS	25
3.4. MKP „ALGORITMAI“ REALIZACIJOS TECHNINĖS PRIEMONĖS	28
3.5. ALGORITMAVIMO PRAKTIKOS INFORMACINIO MODELIO SUDARYMAS	29
3.5.1. DAŽNIAUSIAI DAROMŲ KLAIDŲ ANALIZĖ	29
3.5.2. ŠAKOTO ALGORITMO INFORMACINIAI MODELIAI	33
3.5.3. CIKLINIO ALGORITMO INFORMACINIAI MODELIAI.....	37
3.6. ALGORITMO KONSTRAVIMO MOKOMOSIOS PRIEMONĖS REALIZACIJA SU TESTTOOL.....	44
4. MOKYMO PRIEMONĖS VARTOTOJO DOKUMENTACIJA	47
4.1. MKP „ALGORITMAI“ APRAŠYMAS.....	47
4.2. MKP „ALGORITMAI“ ĮDIEGIMAS	52
5. MKP „ALGORITMAI“ KOKYBĖS ĮVERTINIMAS	53
5.1. TYRIMO TIKSLAS IR UŽDAVINIAI.....	53
5.2. TYRIMO REZULTATAI	54
5.3. MKP „ALGORITMAI“ ĮVERTINIMAS PAGAL PASIRINKTUS VERTINIMO KRITERIJUS	57
5.4. EKSPERTŲ APIBENDRINIMAI	59
6. DARBO REZULTATAI, IŠVADOS	60
LITERATŪRA.....	62
PRIEDAI.....	63
Priedas Nr. 1. MKP priemonių aprašymas.	63
Priedas Nr. 2. MKP vertinimo kriterijai	73
Priedas Nr. 3. Algoritmų struktūrizavimo dokumentacija.....	77
Priedas Nr. 4. Anketa mokiniams.....	86

LENTELĖS

1 lentelė. Mokomųjų programų vertinimo kriterijai	10
2 lentelė. MKP analizė.	15
3 lentelė. MKP vertinimo suvestinė pagal kriterijų grupes.	17
4 lentelė. Tiesinis algoritmas. Klaidų pavyzdžiai ir komentarai.	29
5 lentelė. Šakotas algoritmas. Klaidų pavyzdžiai ir komentarai.	30
6 lentelė. Ciklinis algoritmas. Klaidų pavyzdžiai ir komentarai.	31
7 lentelė. Sąlygos sakinio kontekstinio grafo detalus aprašymas.	34
8 lentelė. Sąlygos sakinys. Užduotis Nr1.	35
9 lentelė. Sąlygos sakinys. Užduotis Nr2.	36
10 lentelė. Ciklo sakinio kontekstinio grafo detalus aprašymas.	38
11 lentelė. Pirmos užduoties ciklo sakinio kontekstinis grafas.	39
12 lentelė. Penktos užduoties ciklo sakinio kontekstinis grafas.	40
13 lentelė. Užduočių sprendimo algoritmai.	41
14 lentelė. Ciklo konstravimo veiksmų sekos grafas.	42
15 lentelė. Veiksmų sekos grafas su taisyklėmis.	43
16 lentelė. MKP „Algoritmai“ įvertinimas.	57
17 lentelė. Algoritmų analizė. Priskyrimo sakinys 1.	77
18 lentelė. Algoritmų analizė. Priskyrimo sakinys 2.	78
19 lentelė. Algoritmų analizė. Sąlygos sakinys.	79
20 lentelė. Algoritmų analizė. Sumos, sandaugos ir kiekio skaičiavimas.	80
21 lentelė. Algoritmų analizė. Sumos, sandaugos ir kiekio skaičiavimas (teigiamo arba neigiamo).	82
22 lentelė. Algoritmų analizė. Minimumo, maksimumo radimas.	84

PAVEIKSLIAI

1 pav. PaskMok programa. Temos.	12
2 pav. Programa „Informatika“.	12
3 pav. Programa „Programavimo pradmenys“. Turinys.	13
4 pav. Elektroninis informatikos vadovėlis.	13
5 pav. Komensio logo programa. Darbinis langas.	14
6 pav. MKP kriterijaus „Mokomoji medžiaga“ įvertinimas.	17
7 pav. MKP kriterijaus „Vartotojo sąsaja“ įvertinimas.	18
8 pav. MKP kriterijaus „Vartotojų administravimo galimybės“ įvertinimas.	18
9 pav. MKP kriterijaus „Interaktyvumas“ įvertinimas.	19
10 pav. MKP kriterijaus „Techninės savybės“ įvertinimas.	19
11 pav. MKP kriterijaus „Dokumentacija ir papildomos priemonės“ įvertinimas.	20
12 pav. MKP kriterijaus „Ekonominis veiksmingumas“ įvertinimas.	20
13 pav. MKP vertinimo suvestinė.	21
14 pav. MKP „Algoritmai“	22
15 pav. MKP „Algoritmai“ struktūra	23
16 pav. MKP „Algoritmai“ vartotojų veiklos diagrama	24
17 pav. MKP „Algoritmai“ išsaugojimo galimybės kompiuterinėse laikmenose.	25
18 pav. MKP „Algoritmai“ realizacijos techninės priemonės.	28
19 pav. Sąlygos sakinio kontekstinis grafas.	33
20 pav. Pirmo uždavinio sprendimo blokinė schema	34
21 pav. Antro uždavinio sprendimo blokinė schema	36
22 pav. Ciklo sakinio kontekstinis grafas.	37
23 pav. TestTool programa sukurtų testų sudėtingumo lygiai (pagal sukurtus pavyzdžius).	44
24 pav. Pirmo lygio užduotis	44
25 pav. Antro lygio užduotis	45
26 pav. Trečio lygio užduotis	45
27 pav. Pirmo lygio ciklinė užduotis	46
28 pav. Antro lygio ciklinė užduotis	46
29 pav. Trečio lygio ciklinė užduotis	47
30 pav. Freimų išsidėstymas puslapyje	48
31 pav. „Popup“ tipo languose yra pateiktos uždavinio struktūrogramos.	48
32 pav. Algoritmo sprendimo filmas	49
33 pav. Ciklinio algoritmo pavyzdys. Rodyklių pagalba mokiniams gali sekti algoritmo žingsnius ir stebėti besikeičiančius rezultatus.	49
34 pav. Pirmo testo pavyzdys.	50
35 pav. Antro testo pavyzdys	50
36 pav. Antro testo rezultatas	51

37 pav. Trečio testo pavyzdys	51
38 pav. Ketvirtą testo pavyzdys.....	52
39 pav. Testo TestTool programa pavyzdys.....	52
40 pav. IT naudojimo namuose galimybių diagrama	54
41 pav. Priimtinausių mokymosi formų diagrama	55
42 pav. MKP „Algoritmai“ mokomosios medžiagos vertinimo diagrama.....	55
43 pav. MKP „Algoritmai“ valdymo vertinimo diagrama	56
44 pav. MKP „Algoritmai“ naudojimo ir naudingumo diagrama.....	56
45 pav. MKP „Algoritmai“ įvertinimas procentais	58
46 pav. PaskMok programa. Užduotis	63
47 pav. PaskMok programa. Kontrolinės užduoties įvertinimas.....	64
48 pav. Programa „Informatika“. Temos.	65
49 pav. Programa „Informatika“. Teorija.	66
50 pav. Programa „Informatika“. Savarankiškos užduotys.	66
51 pav. Programa „Informatika“. Testas.....	66
52 pav. Programa „Programavimo pradmenys“. Teorija.....	68
53 pav. Programa „Programavimo pradmenys“ Programos veikimo modelis.	68
54 pav. Programa „Programavimo pradmenys“. Animuotas uždavinio aiškinimas.....	69
55 pav. Programa „Programavimo pradmenys“. Tikrinamasis darbas.....	69
56 pav. Elektroninis informatikos vadovėlis. Teorija.	70
57 pav. Elektroninis informatikos vadovėlis. Užduotys.	70
58 pav. Komenskio logo programa. Procedūros ir rezultato langai	71
59 pav. Komenskio logo programa. Sukurti projektai.	72

TRUMPINIMAI IR PAAIŠKINIMAI

IT – informacinės technologijos;

IKT – informacinės ir komunikacinės technologijos;

MKP – mokomoji kompiuterinė programa

1. ĮVADAS

Pasaulyje vis daugiau informacinių ir komunikacinių priemonių skverbiasi į visas gyvenimo sritis. Kiekvieno žmogaus galimybė konkuruoti rinkoje priklauso nuo mokėjimo naudotis moderniomis informacijos technologijos priemonėmis ir jas plačiai taikyti visose savo veiklos srityse, tiek darbe, tiek ir gyvenime. Todėl svarbu užtikrinti visą gyvenimą trunkantį mokymąsi.

Mokykla – tai pirmoji pakopa, skatinanti visuomenės informacinį raštingumą. Tikslingas mokymas naudoti informacijos technologiją yra vienas iš būdų paruošti moksleivį sėkmingam ateities gyvenimui. Mokykloje įgyto informacinio išprusimo lygio atitikimas tarptautiniams reikalavimams, sudarytų mūsų šalies moksleiviams lygias galimybes sėkmingai tęsti studijas užsienio aukštosiose mokyklose.

Vis daugiau informacinės komunikacinės technologijos integruojamos į mokymo ir mokymosi procesą. IKT naudojimas pamokose, padaro mokymosi procesą įdomesniu, patrauklesniu mokiniams, kas padeda geriau ir lengviau pasisavinti naują mokymo medžiagą. Naujos programos suteikia mokytojams galimybę efektyviau ir įdomiau dirbti, tačiau jų nėra tiek daug, kad mokytojas galėtų pasirinkti jam tinkančią ir mokiniams prieinamą programą.

Be kompiuterio jau sunkiai įsivaizduojamas dabartinis gyvenimas. Todėl natūralu, kad juo mokytis dirbti reikia pradėti kuo anksčiau. Šiuo metu mokyklose pažintis su kompiuteriu pradedama 5 klasėje.

Vienas iš sunkiausiai įveikiamų ir suvokiamų dalykų vaikams yra algoritmavimas. Nors su algoritmu vaikai supažindinami 6 klasėje („Komenskio Logo“), vyresnėse klasėse programavimas ir algoritmavimas jiems sunkiai sekasi.

Mokant bet kurios programavimo kalbos, didelę reikšmę turi algoritmų atlikimas. Programavimo kalbų konstrukcijos mokiniams yra neįprastos. Norint jas suprasti, reikia ne tik išklaudyti mokytojo aiškinimą ar perskaityti atitinkamą teorijos dalį, bet ir savarankiškai atlikti keletą algoritmų, kuriuose yra panaudotos panašios konstrukcijos. Yra žinoma, kad mokiniai neįgis algoritmavimo įgūdžių, jeigu jie bus tik mokytojo ar draugų aiškinimų klausytojai.

Problema, kad vaikai sunkiai suvokia algoritmų sudarymo principus. Pamokų metu trūksta laiko praktiniam darbui, o moksleivių ugdymas priklauso nuo mokytojo ir pačių moksleivių aktyvumo. Šiuo metu pasirenkamas programavimo kursas mokykloje yra dėstomas 9 – 10 klasėje. Šiam kursui yra skiriamos 34 valandos. Remiantis pagrindinio ugdymo informacinių technologijų bendrąja programa yra nagrinėjamos šios temos:

- Algoritmo samprata, užrašymo būdai
- Programavimo kalbos, kompiliatoriai

- Algoritmų rengimas, kodavimas programavimo kalba ir programos vykdymas kompiuteriu
- Programos ir vartotojo dialogas
- Duomenų įvedimas ir išvedimas, spausdinimo formatai
- Pagrindiniai algoritmų veiksmai: priskyrimas, šakojimas, ciklas
- Duomenų tipai
- Programos sudarymo etapai
- Programos teisingumas, kontroliniai duomenys
- Programavimo stilius ir kultūra
- Paprasčiausi algoritmai ir jų programavimas [15]

Norint sudominti mokinius, reikia ieškoti naujų mokymo formų ir būdų. Čia į pagalbą mokytojas gali pasitelkti įvairias mokomasias programas, testus. Taip ne tik praturtindamas mokymosi medžiagą ar mokymosi procesą, bet suteikdamas mokiniui galimybę mokytis nuotoliniu būdu, pasirinktu laiku, individualiu tempu, patogioje vietoje.

Mokomųjų priemonių sukurtų šiam kursui nėra daug. Daugelis mokomųjų programų yra morališkai pasenusios ir sunkiai pritaikomos šiuolaikinei kompiuterinei įrangai.

Darbo tikslas – išanalizuoti interaktyvių technologijų taikymo galimybes programavimo pamokose, sukurti lengvai įsisavinamą ir valdomą MKP, kuri būtų skirta vyresniųjų klasių mokiniams ir pirmo kurso studentams.

Darbo uždaviniai:

- Esamų mokomųjų programų analizė pagal pasirinktus vertinimo kriterijus;
- MKP „Algoritmai“ struktūrinis projektavimas;
- Parinkti technologijas MKP struktūrinių komponentų įgyvendinimui;
- Išanalizuoti dažniausiai daromas tipines mokinių klaidas;
- Sukurti algoritmų informacinius modelius. Parinkti taisykles tradiciniams algoritmavimo uždaviniams spręsti.
- Realizuoti mokomąją priemonę su TestTool.
- Parengti MKP „Algoritmai“ dokumentaciją.
- Atlikti MKP „Algoritmai“ kokybės įvertinimą.

2. ANALITINĖ DALIS

Informacinės technologijos (IT) vis labiau įsilieja į mokymo ir mokymosi procesą, tampa neatsiejama jo dalimi. Visuomenės pokyčiai, milžiniška technologijų plėtra ir įvairovė verčia naujai žvelgti į mokymą kaip visuotinį procesą. Iš esmės kinta mokymo koncepcijos, pedagoginės nuostatos, tikslai, metodai. Šiuolaikines informacines priemones ir technologijas skatinama vartoti visuose mokomojo proceso lygmenyse, mokant ir besimokant įvairių dalykų [13].

Organizuodamas ugdymo procesą, siekdamas, kad mokiniai sėkmingai įsisąmonintų mokomąją medžiagą, būtų kūrybiški, norėtų daugiau patys sužinoti, mokytojas taiko įvairius mokymo metodus [16]. Dažnai tai, kokios informacinės komunikacinės technologijos bus pasirenkamos konkrečiai pamokai ar popamokinei veiklai, priklauso nuo turimų informacinių komunikacinių technologijų resursų mokykloje, nuo mokinių gebėjimų dirbti su informacinėmis komunikacinėmis technologijomis. Svarbiausia, kad taip organizuotas ugdymo procesas duotų teigiamų rezultatų, skatintų moksleivių mokymosi motyvaciją, ugdytų jų kompetenciją dirbti su naujomis technologijomis.

Pamokų, kuriose remiamasi informacinėmis komunikacinėmis technologijomis, vertė ir efektyvumas priklauso nuo mokytojo sugebėjimo tinkamai pritaikyti mokomasias kompiuterines programas, pasitelkti reikiamą medžiagą iš interneto [7].

Šioje dalyje aptariamos mokomosios programos skirtos algoritmavimo ir programavimo uždaviniams spręsti.

2.1. MOKOMŪJŲ KOMPIUTERINIŲ PROGRAMŲ VERTINIMO KRITERIJAI

Mokomųjų programų nuolat daugėja, todėl pirmiausiai būtina apžvelgti įvairius jų klasifikavimo požymius. Vis daugiau ir daugiau atsiranda nemokamų, atvirųjų programų. Programos nagrinėjamos atsižvelgiant į mokymo dalykus: kokiems dalykams mokyti skirtos šios programos, kokių mokymo tikslų jomis siekiama ir pan. [12]

Mokomosios kompiuterinės priemonės buvo vertinamos remiantis mokyklų, vykdančių bendrojo lavinimo programas, aprūpinimo mokomosiomis kompiuterinėmis priemonėmis tvarkos aprašo 2 priedu, rastu adresu www.emokykla.lt/admin/file.php?id=381. (Dokumentas pateikiamas priede Nr. 2).

Remdamasis šiuo dokumentu iškelti vertinimo kriterijai:

1 lentelė. Mokomųjų programų vertinimo kriterijai.

Eil. Nr.	MKP vertinimo kriterijai	MKP vertinimo aprašymas
1	Mokomoji medžiaga	<p>Ar pateikta <u>teorinė medžiaga</u>? Ar ji detali? Ar lengvai suprantama?</p> <p>Ar pateikti <u>pavyzdžiai</u>? Ar pavyzdžiai atitinka teorinę medžiagą? Pavyzdžių tikslingumas.</p> <p>Ar perskaičius teorinę medžiagą pateikiami <u>savikontrolės klausimai</u>? Ar savikontrolės klausimuose yra pateikiamos nuorodos į teisingus atsakymus? Ar savikontrolės klausimai atitinka teorinę medžiagą?</p> <p>Ar suteikiama galimybė besimokančiajam atlikti <u>mokomąsias užduotis</u>? Ar jos atitinka pasirinktą temą?</p> <p>Ar pateiktos <u>kontrolinės užduotys</u>? Ar programa pateikia žinių įvertinimus?</p> <p>Ar pateikiama <u>klaidų analizė</u>? Ar nurodoma kur besimokantysis suklydo? Ar pateikiamos nuorodos į teisingą atsakymą (nuoroda į teoriją)?</p> <p>Ar suteikiama galimybė <u>modeliuoti programas</u>? Ar modeliavimas vykdomas grafinėje sąsajoje?</p> <p>Ar pateikiama <u>daug užduočių</u> (mokiniam pateikiami skirtingi užduočių variantai)? Ar užduotys nesikartoja?</p> <p>Ar pačiose programose yra pastebėta daug <u>klaidų</u>?</p>
2	Vartotojo sąsaja	<p>Ar grafinė vartotojo sąsaja? Ar valdymas vykdomas pele? Ar <u>patogus programos valdymas</u>?</p> <p>Ar <u>ergonomiška</u> ir kokybiška daugialypė terpė?</p>
3	Administravimo galimybės	<p>Ar yra <u>apskaitos apie vartotojus</u> galimybės?</p> <p>Ar galima mokymosi <u>procesą individualizuoti</u>?</p> <p>Ar yra <u>asmeninių duomenų ir informacijos apsauga</u>?</p>

4	Interaktyvumas	<p>Ar yra <u>grįžtamasis ryšys</u>? Ar <u>pranašesnis už tradicines mokymo priemones</u>?</p> <p>Ar pritaikyta <u>savarankiškam darbui</u>?</p> <p>Ar yra <u>papildomos mokymosi ir mokymosi organizavimo priemonės</u>: pagalbos sistema; žinynai ir žodynai; paieška; vartotojo vadovas, kurso kalendorius; užrašų knygelė; studijų gidas ir kt.?</p>
5	Bendravimo ir bendradarbiavimo galimybės	Ar įmanoma <u>sinchroniškai</u> ir <u>asinchroniškai</u> bendrauti?
6	Techninės savybės	<p>Ar programa <u>stabiliai veikia</u>? Kokia <u>darbo sparta</u> ir ar geras prieinamumas?</p> <p>Ar <u>nesudėtingas programos įdiegimas</u> ir <u>suderinimas su esama technine ir sisteme įranga</u>?</p>
7	Dokumentacija ir papildomos priemonės	<p>Ar pateikta <u>bendra informacija</u>?</p> <p>Ar yra išsamus <u>įdiegimo, naudojimo ir valdymo galimybių aprašymas</u>?</p> <p>Ar <u>pateiktos metodinės rekomendacijos ir papildomos metodinės priemonės</u> mokiniui bei mokytojui?</p> <p>Ar <u>taisyklingas stilius ir kalba</u>?</p>
8	Ekonominis veiksmingumas	<p><u>Kaina</u>. Ar reikalingos <u>papildomos sąnaudos</u> ir ištekliai programos įdiegimui?</p> <p>Ar mokomąją kompiuterinę <u>programą galima nuolatos naudoti ir atnaujinti</u> esant reikalui?</p>

2.2. MOKOMŲJŲ KOMPIUTERINIŲ PRIEMONIŲ PASIRINKIMAS

Rengimasis gyventi informacinėje visuomenėje, intensyvėjantis technologijų naudojimas įvairiose gyvenimo srityse, žinių visuomenės kūrimas vis daugiau reikalauja skirti dėmesio tiems įrankiams, kurie paverčia kompiuterį mokomąja priemone: programinei įrangai, kompiuterinėms programoms. Išskiriamos trys pagrindinės programų grupės:

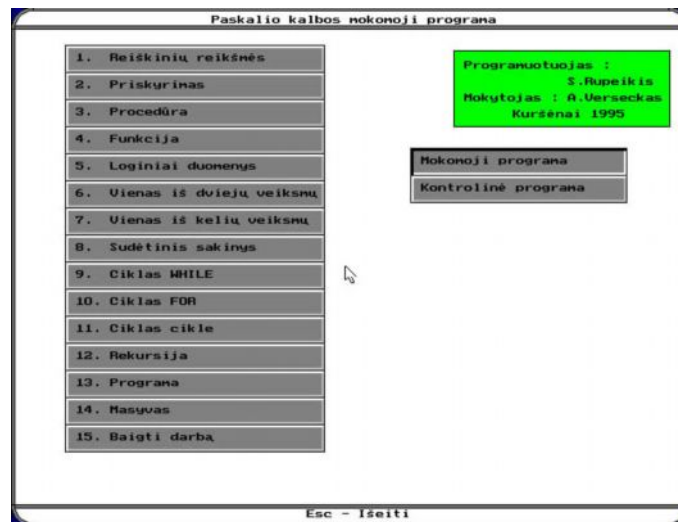
1. Sisteminė programinė įranga,
2. Bendrosios paskirties programinė įranga,

3. Įvairių dalykų mokomosios programos [5].

Mokomųjų kompiuterinių priemonių skirtų mokyti algoritmavimo nėra daug. Apžvelgsime dažniausiai mokyklose naudojamas programas:

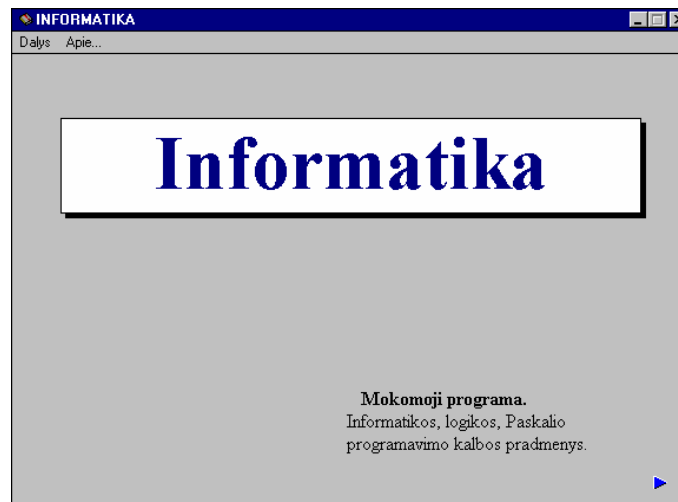
PaskMok. Autoriai: S.Rupeikis, A.Verseckas. Kuršėnai 1995.

Programa pateikia mokiniams atlikti keletą algoritmų arba jų fragmentų ir patikrina gautų rezultatų teisingumą. Pradėjus veikti programai, atsiranda meniu, kuriame galima pasirinkti vieną iš 14 temų.



1 pav. PaskMok programa. Temos.

Informatika. Autorius A.Bačiulis. 1997



2 pav. Programa „Informatika“.

Programa skirta bendrojo lavinimo mokyklų 10–12 klasių mokiniams. Tai kompiuterizuotas V.Dagienės ir G.Grigo vadovėlio „Informatika“ variantas – elektroninė knyga su testu žinioms patikrinti.

Programavimo pradmenys. Autoriai: A. Verseckas, A. Jarockis Kuršėnų L.Ivinskio gimnazija.

E-vadovėlių sudaro 11 skyrių.

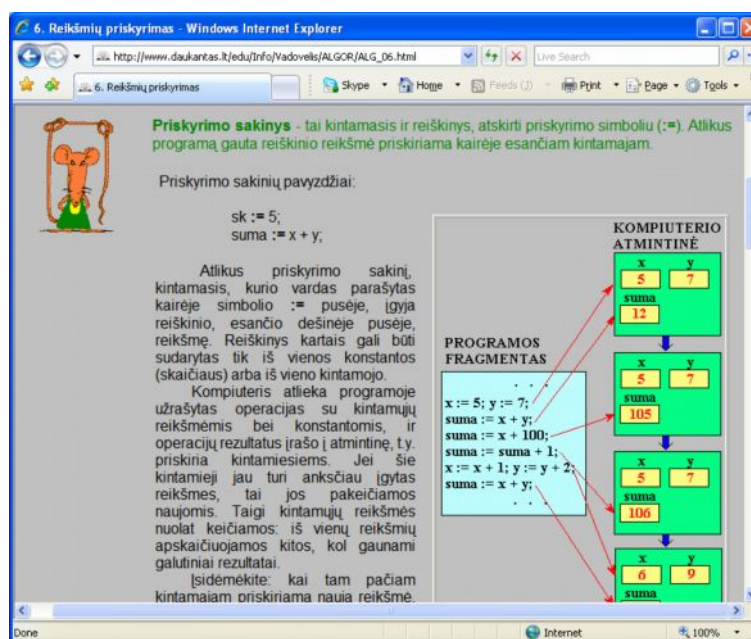


3 pav. Programa „Programavimo pradmenys“. Turinys.

Kiekviename skyriuje yra aiškinamoji tekstinė ir animacinė medžiaga.

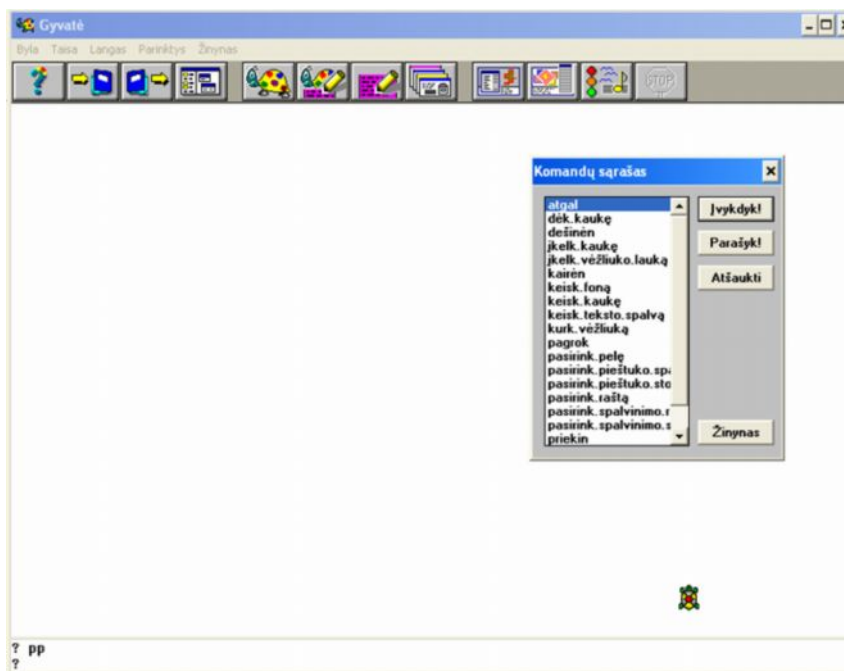
Elektroninis informatikos vadovėlis. <http://www.daukantas.lt/edu/Info/Vadovelis/>

Kompiuterinis „Informatikos vadovėlis“ sukurtas Alytaus Vidzgirio vidurinėje mokykloje. Autorius – informatikos vyr. mokytojas V.Žemaitis. Vadovėlyje mokomoji medžiaga pateikta pagal Valentinos Dagienės vadovėlius INFORMATIKOS PRADMENYS.



4 pav. Elektroninis informatikos vadovėlis.

Komenskio Logo programa labiausiai tinka kūrybiškam darbui – programavimui, projektavimui, kurį gali nesunkiai atlikti net jaunesnio amžiaus vaikai. Iš vienos pusės, sistema nesudėtinga, greitai perprantami jos pagrindiniai veiksmai, vaizdžiai išreiškiami rezultatai, iš kitos pusės – ji turtinga įvairiomis šiuolaikinių programavimo kalbų idėjomis bei konstrukcijomis, patogiomis priemonėmis kompiuterio programinei įrangai projektuoti bei kurti. [6].



5 pav. Komenskio logo programa. Darbinis langas.

Detalesnis esamų MKP aprašymas pateiktas priede Nr.1

2.3. MOKOMŪJŲ PROGRAMŲ ANALIZĖ

Surašius MKP vertinimo kriterijus pasirinkta tokia vertinimo sistema:

- ✓ 3 – puikiai atitinka iškeltus kriterijus;
- ✓ 2 – gerai atitinka iškeltus kriterijus;
- ✓ 1 – iš dalies atitinka kriterijus;
- ✓ 0 – neatitinka keliamų kriterijų;

Buvo pasirinktas ekspertinis vertinimo tipas. Apklausti trys informacinių technologijų mokytojai, dirbantys su šiomis programomis. Gauti rezultatai pateikti 2 lentelėje.

2 lentelė. MKP analizė.

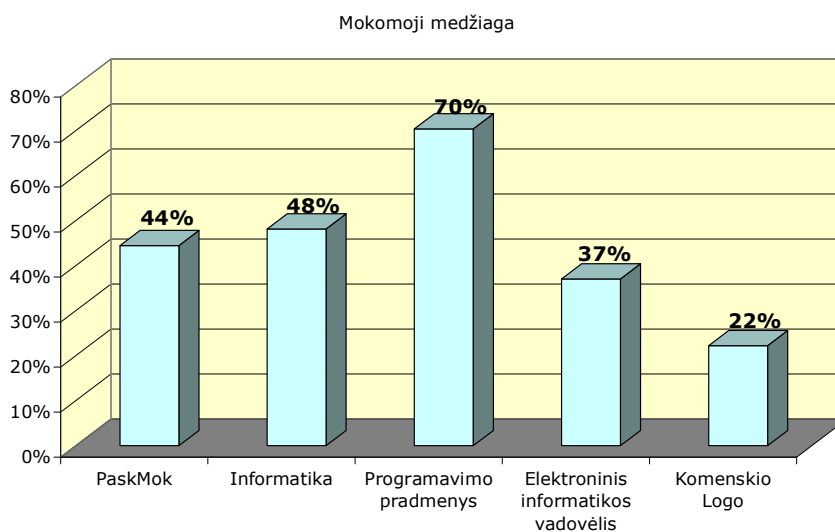
	PaskMok	Informatika	Programavimo pradmenys	Elektroninis informatikos vadovėlis	Komenskio Logo
Mokomoji medžiaga					
Teorinė dalis	0	2	3	3	1
Pavyzdžiai	0	2	3	3	3
Savikontrolės klausimai	0	0	3	1	0
Mokomosios užduotys	3	2	3	1	0
Kontrolinės užduotys	3	2	2	1	0
Programų modeliavimas	0	0	3	0	2
Klaidų analizė	0	2	0	0	0
Užduočių įvairovė (užduotys skirtingos)	3	0	1	1	0
Klaidos (pateikus klaidingą atsakymą, programa jį užskaito kaip teisingą)	3	3	1	0	0
Vartotojo sąsaja:					
Patogumas	2	2	3	2	3
Ergonomiškumas	2	2	3	2	3
Vartotojų administravimo galimybės					
Vartotojų administravimo priemonės	1	0	0	0	0
Mokymosi proceso individualizavimo galimybės	1	0	0	0	0
Asmeninių duomenų ir informacijos saugumas.	1	0	0	0	2
Interaktyvumas					
Grįžtamasis ryšys	1	2	1	0	1
Pranašumas prieš tradicines mokymo priemones	2	2	3	1	3
Savarankiško darbo galimybės	1	2	3	1	2
Papildomos mokymosi priemonės	1	1	2	1	1
Mokymosi organizavimo priemonės	0	0	0	0	0

Bendravimo ir bendradarbiavimo galimybės					
Sinchroninės priemonės	0	0	0	0	0
Asinchroninės priemonės	0	0	0	0	0
Techninės savybės					
Veikimo stabilumas	3	3	3	2	2
Darbo sparta	3	3	3	3	3
Nesudėtingas programos įdiegimas	3	3	3	3	1
Suderinimas su esama technine ir sisteminė įranga	2	2	3	3	3
Dokumentacija ir papildomos priemonės					
Bendra informacija	0	3	2	1	3
Įdiegimo aprašymas	0	2	1	0	0
Naudojimo ir valdymo galimybių aprašymas	0	2	2	0	0
Metodinės rekomendacijos ir papildomos metodinės priemonės mokiniui	0	0	0	0	0
Metodinės rekomendacijos ir papildomos metodinės priemonės mokytojui	0	0	0	0	0
Taisyklingas stilius ir kalba	3	3	3	3	3
Ekonominis veiksmingumas					
Kainos prieinamumas	3	3	3	3	3
Papildomos sąnaudos ir ištekliai programos įdiegimui	2	2	3	3	3
Programą galima nuolatos naudoti ir atnaujinti esant reikalui	0	0	2	1	2

Susumuoti duomenys, pagal kriterijų grupes, pateikti 3 lentelėje.

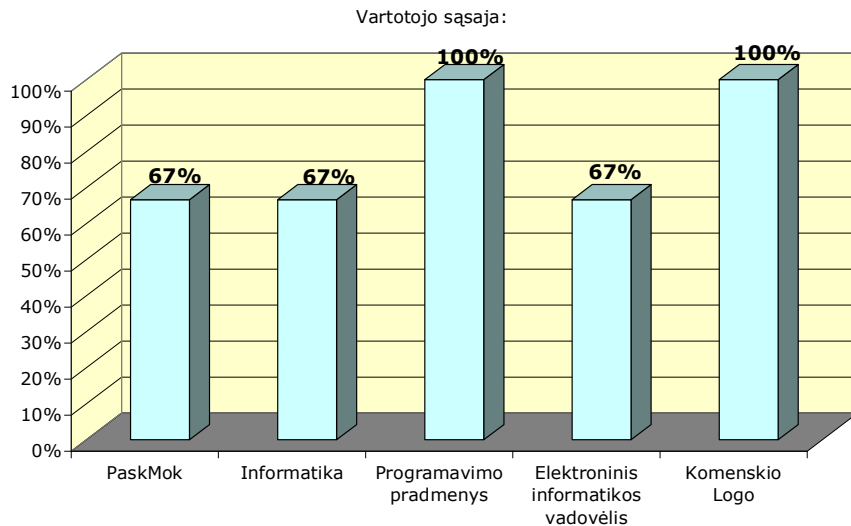
3 lentelė. MKP vertinimo suvestinė pagal kriterijų grupes.

	PaskMok	Informatika	Programavimo pradmenys	Elektroninis informatikos vadovėlis	Komenskio Logo
Mokomoji medžiaga	12	13	19	10	6
Vartotojo sąsaja:	4	4	6	4	6
Vartotojų administravimo galimybės	3	0	0	0	2
Interaktyvumas	5	7	9	3	7
Bendravimo ir bendradarbiavimo galimybės	0	0	0	0	0
Techninės savybės	11	11	12	11	9
Dokumentacija ir papildomos priemonės	3	10	8	4	6
Ekonominis veiksmingumas	5	5	8	7	8



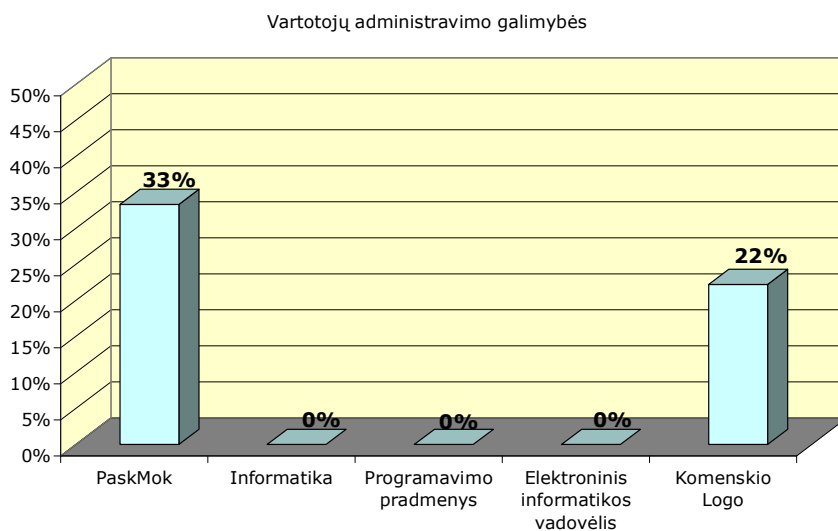
6 pav. MKP kriterijaus „Mokomoji medžiaga“ įvertinimas.

Pasirinktus kriterijus mokamajai medžiagai geriausiai atitinka MKP „Programavimo pradmenys“ – 70% (žr. 6 pav.). „PaskMok“ programoje visiškai nėra teorinės medžiagos, pavyzdžių, savikontrolės klausimų, klaidų analizės. Bet šioje MKP puikiai įvertintos mokomosios, kontrolinės užduotys, didelė užduočių įvairovė. Klaidų analizė yra tik MKP „Informatika“. Programoje „Elektroninis informatikos vadovėlis“ yra pateikta teorinė medžiaga su pavyzdžiais. Nėra galimybės testuoti, įvertinti mokinių žinias.



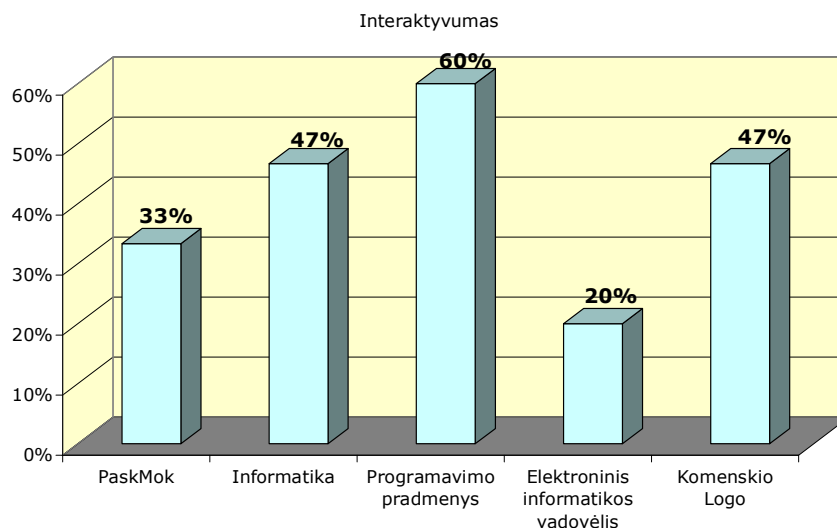
7 pav. MKP kriterijaus „Vartotojo sąsaja“ įvertinimas.

Vartotojo sąsaja MKP „Programavimo pradmenys“ ir „Komensio Logo“ yra įvertinta puikiai (žr. 7 pav.). Kitos MKP turi trūkumų. Pavyzdžiui „PaskMok“ programa dirba Dos terpėje ir yra gana sudėtinga pasinaudoti kita papildoma programine įranga (pav. skaičiuokliu).



8 pav. MKP kriterijaus „Vartotojų administravimo galimybės“ įvertinimas.

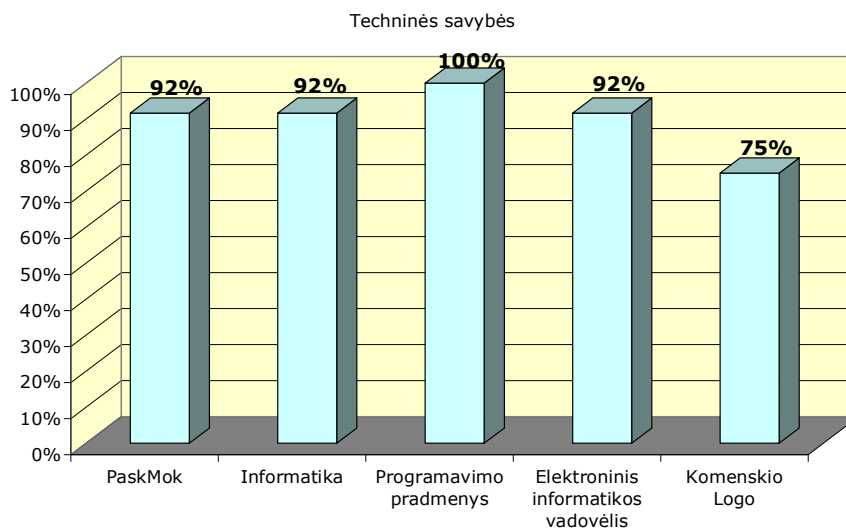
Minimaliai administruoti vartotoją galima tik MKP „PaskMok“ (žr. 8 pav.). Čia vartotojui, atliekant kontrolinę užduotį, yra suteikiama galimybė prisiregistruoti. Atlikus užduotį specialia klavišų kombinacija išiname iš įvertinimų lapo.



9 pav. MKP kriterijaus „Interaktyvumas“ įvertinimas.

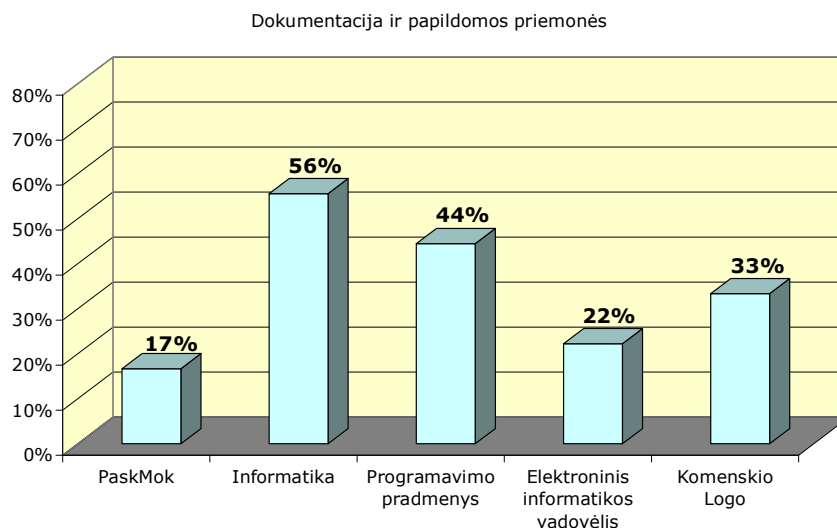
Kriterijui „Interaktyvumas“ geriausiai atitinka MKP „Programavimo pradmenys“ (žr. 9 pav.). Šioje priemonėje mokinys gali spręsti užduotis, testus, reikalaujančius teksto įrašymo ar manipuliavimo grafiniais objektais, yra galimybė programos pavyzdį įkelti tiesiai į programavimo aplinką ir stebėti gaunamus rezultatus.

Bendravimo ir bendradarbiavimo galimybių šiose MKP nėra.



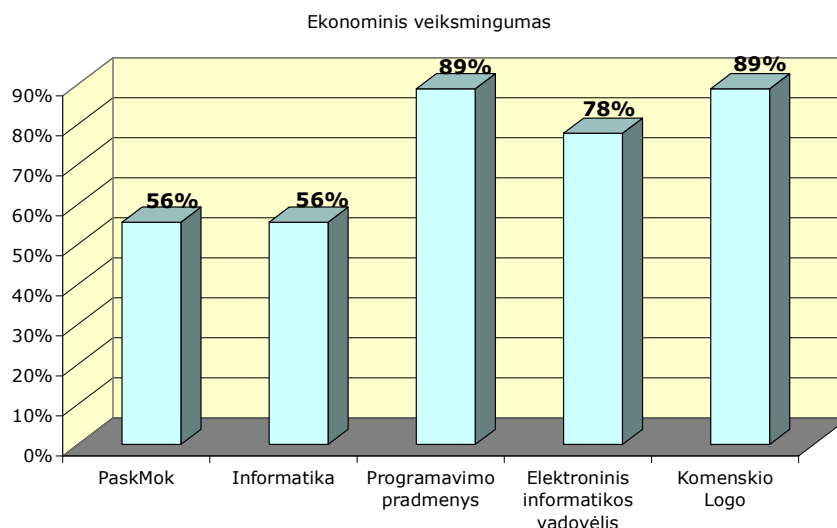
10 pav. MKP kriterijaus „Techninės savybės“ įvertinimas.

Visos MKP atitinka keliamus techninius reikalavimus (žr. 10 pav.). Programos veikia stabiliai ir sparčiai. Problemų kyla diegiant MKP „Komensio Logo“ ir su lietuviškais simboliais programose „PaskMok“ ir „Informatika“.



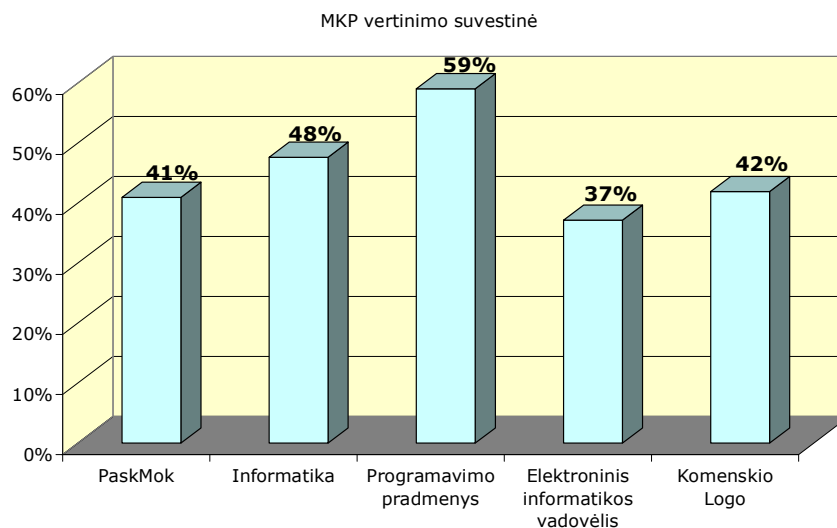
11 pav. MKP kriterijaus „Dokumentacija ir papildomos priemonės“ įvertinimas.

Diegimo instrukcija ir valdymo galimybių aprašymas yra pateikiamas tik MKP „Informatika“ (žr. 11 pav.). Metodinių rekomendacijų tiek mokiniui, tiek mokytojui nėra nė vienoje iš priemonių. Mažiausiai bendros informacijos yra pateikta apie MKP „PaskMok“.



12 pav. MKP kriterijaus „Ekonominis veiksmingumas“ įvertinimas.

Visos pateiktos MKP yra nemokamos (žr. 12 pav.). Jas diegiant papildomos programinės įrangos reikalauja tik „PaskMok“ ir „Informatika“. Apie MKP atnaujinimo galimybes informacijos nėra pateikiama.



13 pav. MKP vertinimo suvestinė.

Susumavus visus įvertinimus matyti (žr. 13 pav.), kad geriausiai iškeltus kriterijus atitinka A. Versecko ir A. Jarockio MKP „Programavimo pradmenys“. Aišku vienareikšmiškai sakyti, kad ši programa yra geriausia negalima. Kiekviena MKP turi savo paskirtį. „Programavimo pradmenys“ yra skirta daugiau savarankiškam mokymuisi. MKP „PaskMok“ – žinių kontrolei, o „Elektroninis informatikos vadovėlis“ – teorinei medžiagai dėstyti. Norint pasiekti gerų mokymo rezultatų, reikia naudoti kelias MKP.

MKP trūkumai:

- MKP „Informatika“ yra parengta pagal jau seniai mokykloje nenaudojamą V.Dagienės ir G.Grigo informatikos vadovėlį;
- Nėra užduočių klasifikacijos pagal sudėtingumo lygius (visos MKP);
- Įdiegus programą atsiranda problema su lietuviškais simboliais („Informatika“, „PaskMok“);
- Prie šiuolaikinio ekrano rezoliucijos yra iškraipomas vaizdas („Informatika“, „PaskMok“);
- Problemiškas papildomų programų naudojimas („PaskMok“).

3. PROJEK TINĖ DALIS

3.1. MKP „ALGORITMAI“ PASKIRTIS IR PAGRINDIMAS

MKP „Agoritmai“ pagrindinis tikslas supažindinti mokinius su algoritmu, jo savybėmis, tipais, pavyzdžiais bei uždaviniais. Svarbu, kad medžiagos pateikimas būtų paprastas ir lengvai suprantamas mokiniams. Jiems turi būti suteikta galimybė patiems dalyvauti algoritmo kūrimo procese.

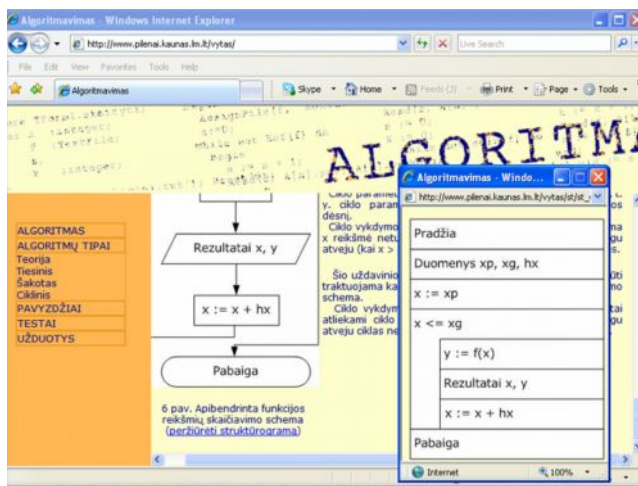
Kadangi mokykliniuose vadovėliuose mažai dėmesio skiriama algoritmavimo uždaviniais, mokiniams gana sunku įsisavinti šią temą. Skaityti vien tik taisykles ir nematyti algoritmo kūrimo principo yra nuobodu. Mokiniai daug mieliau dirba su mokomosiomis programomis. Kadangi programa bus talpinama į Web serverį, tai mokiniai įgys galimybę savarankiškai mokytis ir atlikti praktines užduotis.

Šioje MKP pateikiama informacija bus naudinga ne tik mokiniui, bet ir mokytojui. Ši priemonė padės mokytojui išdėstyti medžiagą daug paprasčiau ir vaizdžiau.

Išanalizavus MKP buvo iškelti reikalavimai kuriamai priemonei:

1. Programa turėtų būti nedidelės apimties;
2. Neturėtų reikalauti papildomų išteklių jos diegimui ir techninei įrangai;
3. Patogi sąsaja, nesudėtingas valdymas;
4. Programoje pateikta teorinė medžiaga;
5. Turi būti pateiktos praktinės užduotys,

Kuriama priemonė „Algoritmai“, skirta 10 – 12 klasių moksleiviams, taip pat ja galės naudotis ir 1 kurso studentai, kuriems yra dėstomas programavimo kursas. Mokymo priemonėje mokiniai supažindinami su algoritmo sąvoka, savybėmis, algoritmų vaizdavimo būdais, algoritmų tipais.

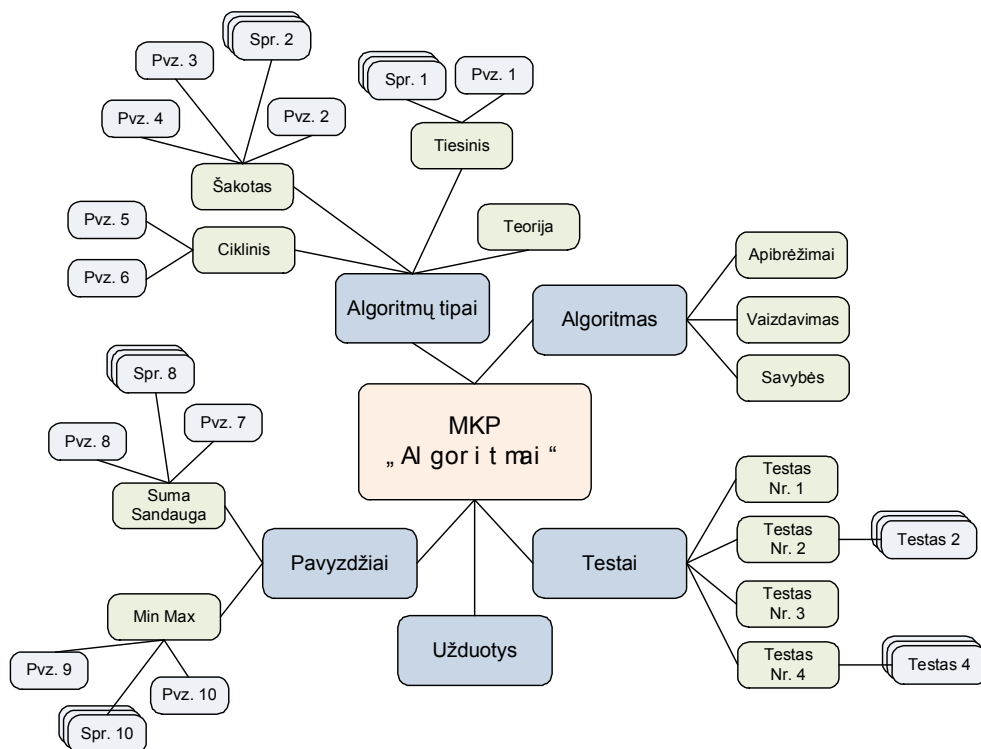


14 pav. MKP „Algoritmai“

3.2. PRIEMONĖS STRUKTŪRA, REIKALAVIMAI, VARTOTOJAI

Šią priemonę sudaro 5 dalys:

1. Algoritmas;
2. Algoritmų tipai;
3. Pavyzdžiai;
4. Testai;
5. Užduotys



15 pav. MKP „Algoritmai“ struktūra

Mokomoji kompiuterinė priemonė bus realizuojama atskiromis formomis, kurios išskviečiamos pasirinkus atitinkamus meniu punktus ir papunkčius.

Meniu punktą „Algoritmai“ sudarys trys papunkčiai: „Apibrėžimai“, „Vaizdavimas“, „Savybės“. Papunktyje „Apibrėžimai“ bus pateikiama teorinė medžiaga apie algoritmus. Papunktyje „Vaizdavimas“ pateikiami įvairūs algoritmo vaizdavimo būdai bei pavyzdžiai. Papunktyje „Savybės“ bus aprašytos pagrindinės algoritmų savybės.

Meniu punktą „Algoritmų tipai“ sudarys keturi papunkčiai: „Teorija“, „Tiesinis“, „Šakotas“, „Ciklinis“. Papunktyje „Teorija“ aptariama blokinių schemų ir struktūrogramų konstravimo principai. Papunkčiuose „Tiesinis“, „Šakotas“, „Ciklinis“ aptariamos atitinkamos algoritmų konstrukcijos, pateikiami statiniai ir aktyvūs pavyzdžiai.

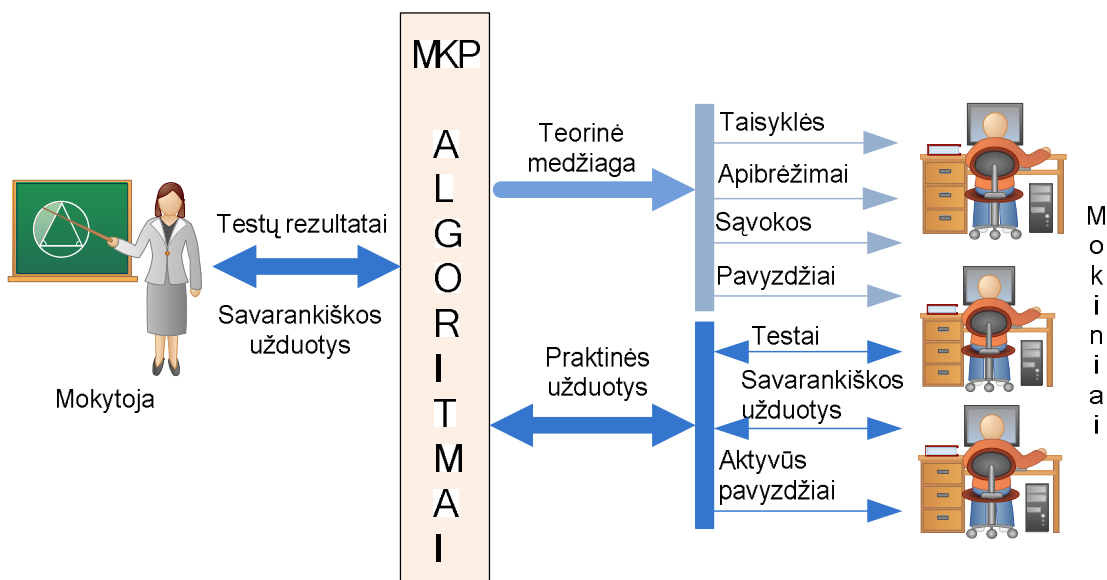
Menių punktą „Pavyzdžiai“ sudarys du papunkčiai „Suma Sandauga“ ir „Min Max“. Papunktyje „Suma Sandauga“ bus pateikiami sumos ir sandaugos skaičiavimo statiniai ir aktyvūs

pavyzdžiai. Papunktyje „Min Max“ bus pateikiami funkcijos minimumo ir maksimumo radimo algoritmų pavyzdžiai.

Menių punktą „Testai“ sudarys keturi papunkčiai „Testas Nr. 1“, „Testas Nr. 2“, „Testas Nr. 3“, „Testas Nr. 4“.

Pasirinkus menių punktą „Užduotys“ bus galima parsisiųsti užduočių lapą.

Mokomosios kompiuterinės priemonės vartotojai yra mokiniai ir mokytojai.



16 pav. MKP „Algoritmai“ vartotojų veiklos diagrama

Mokiniai šioje priemonėje galės naudotis teorine medžiaga ir atlikti praktines užduotis.

Teorinę medžiagą sudaro:

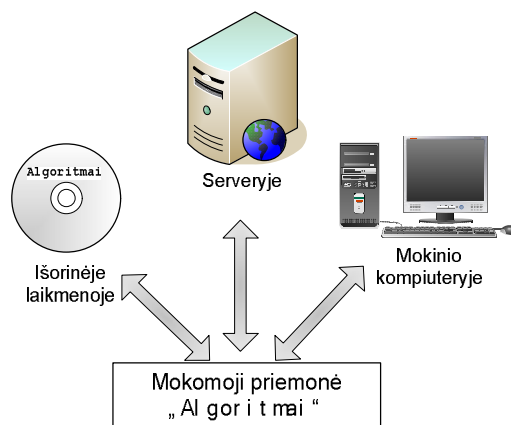
1. Taisyklės;
2. Apibrėžimai;
3. Sąvokos;
4. Pavyzdžiai.

Praktinės užduotys apima:

1. Testus;
2. Savarankiškas užduotis;
3. Aktyvius pavyzdžius.

Mokytojui priemonė pateiks mokinių testų rezultatus. Mokytojas skirs savarankiškas užduotis.

Priemonė galės būti įdiegta mokinio kompiuteryje, patalpinta serveryje ar įrašyta kompaktinėje plokštelėje.



17 pav. MKP „Algoritmai“ išsaugojimo galimybės kompiuterinėse laikmense

Reikalavimai vartotojo kompiuteriui:

- Operacinė sistema Windows 9X/ME/2000/XP/Vista;
- Kompiuterio techniniai reikalavimai turi atitikti instaliuotos operacinės sistemos keliamus reikalavimus;
- Interneto naršyklė (Internet Explorer, Opera, Mozilla);
- Adobe Flash Player, Java.

3.3. MKP REALIZACIJOS TECHNINIŲ PRIEMONIŲ PARINKIMAS IR APRAŠYMAS

MKP sukurti pasirinkta ši programinė įranga:

1. Macromedia Dreamweaver MX 2004;
2. ViewletCam;
3. Adobe Photoshop CS2;
4. CourseBuilderInteraction;
5. Macromedia Flash MX 2004;
6. Microsoft Office Visio 2003;
7. JavaScript programavimo kalbą.

Testams kurti buvo naudojama programa TestTool Author.

Macromedia Dreamweaver MX 2004 – tai modernus ir profesionalus HTML redaktorius, turintis ir tekstinio, ir vizualinio puslapių redagavimo galimybes. Dreamweaver vizualinio redagavimo galimybės padeda greitai kurti arba keisti projekto dizainą nerašinėjant kodų (yra galimybė pasirinkus atitinkamą darbo režimą kurti puslapius ir rašant tiesiogiai kodą). Dreamweaver programa galima kurti savo komandas ir objektus ir t.t. Iš kitų konkurentų jis išsiskiria šiuolaikinių funkcijų ir redagavimo priemonių gausa ir skirtingai negu MS FrontPage neprikuria nereikalingo kodo. Programoje „Dreamweaver“ yra galimybė naudoti naujausias WWW technologijas: „Dynamic HTML“, „Java“, „JavaScript“, „ActiveX“, ir pan.. Ši programa

yra pilnai suderinta su Macromedia Flash programa. Todėl su Macromedia Dreamweaver sėkmingai galima talpinti į puslapius ir flash tipo failiukus.

Macromedia Flash MX 2004. Su Macromedia Flash galima kurti ne tik animacinius objektus. Kadangi šioje programoje yra naudojamas ActionScript, su ja galime padaryti ir mokomojo, ir testavimo tipo programas. Galima kurti testus, kuriose atsakymus reikia nutempti į tam tikras vietas.

CourseBuilder skirta Dreamweaver (testų kūrimo programa). CourseBuilder yra nemokamas Macromedia Dreamweaver priedas. Įdiegę šį priedą į Dreamweaver sistemą, jūs labai paprastai galėsite kurti įvairius testus. CourseBuilder turi daug klausimų tipų: vilkti ir palikti, parinkti vieną, parinkti kelis, tiesa-melas, teksto įvedimo ir tiriamuosius pratimus.

ViewletCam – automatinis ekrane vykstančių veiksmų fiksavimo (įrašymo) įrankis su įvairiomis galimybėmis. Į įrašytą filmuką gana paprastai galima talpinti savo komentarus, įrašyti garsą, išsaugoti ir redaguoti pelės žymeklio nueitą kelią. Sukurtą bylą galima saugoti trimis formatais tai *avi*, *swf*, ir *gif*. *Swf* formatą galima konvertuoti į *exe* bylą.

Microsoft Office Visio 2003 – tai diagramų kūrimo programa, leidžianti kurti verslo ir technikos diagramas, kuriose sistemingai pateikiamos kompleksinės idėjos, procesai ir sistemos. Visio programa sukurtos diagramos leidžia vizualiai aiškiai, glaustai ir efektyviai pateikti tekstą, skaičius ir grafiką.

Adobe PhotoShop CS2 – tai populiariausia piešimo ir grafikos redagavimo programa. Jos pagalba galima ne tik redaguoti nuotraukas, bet ir tvarkyti kompiuterio ekrano atvaizdus. Yra galimybės sumažinti, paryškinti, apkarpyti norimą grafinį dokumentą.

JavaScript – objektiškai orientuota skriptų programavimo kalba, besiremianti prototipų principu. Dažniausiai kalba naudojama internetinių puslapių interaktyvumo realizacijai, bet taip pat naudojama ir kaip galimybė skriptais manipuluoti tam tikromis programomis. Paprastai JavaScript kalbos kodas įtraukiamas į HTML puslapius, tokiu būdu išplečiant statinius HTML puslapius dinaminio skripto funkcionalumu – galimas anketų parametrų tikrinimas, naujų langų atidarymas, suskleidžiamos hierarchinės struktūros rodymas, išsiskleidžiantis meniu ir daug kitų interaktyvumo formų. JavaScript kalba remiasi kelios pagrindinės svetainių kūrimo metodologijos – DHTML (Dinaminis HTML), AJAX, SPA. [9].

Testtool – tai nuotolinio testavimo sistema. Ji leidžia kurti tekstinius ir grafinius testus bei pateikti juos internete. TestTool sistemoje išskiriamos trys dalys: Studentas, Autorius, Administratorius.

Studento posistemis atlieka šias funkcijas:

- ✓ Studento autorizavimas.
- ✓ Testavimo būdo parinkimas (treniruotė ar atsiskaitymas).

- ✓ Grupių, kurioms priskirtas studentas, nustatymas.
- ✓ Testų, kuriuos gali atlikti studentas, parinkimas.
- ✓ Parinkto testo vaizdavimas.
- ✓ Testo klausimų įvertinimo parodymas.

Pagrindinė šios testavimo sistemos savybė, skirianti ją iš kitų tarpo – interaktyvios grafinės aplinkos studento veiklai sudarymas. Šiuo atveju studentas ne pasirenka, o pats konstruoja atsakymą. Studentas turi galimybę spręsti tą patį testą treniruotės ir atsiskaitymo tikslu. Taip atsiranda galimybė sudaryti testų grandinę su tam tikrais apribojimais, kai studentui suteikiama galimybė spręsti tolesnį testą.

Administratoriaus posistemė atlieka šias funkcijas:

- ✓ Vartotojų valdymas – sukūrimas bei ištrynimasis.
- ✓ Vartotojų grupių valdymas – grupių kūrimas, laiko apribojimų (kada grupė gali laikyti testus) nustatymas, studentų priskyrimas grupėms.
- ✓ Kursų valdymas – kūrimas ir trynimasis, grupių bei testų priskyrimas.
- ✓ Testų valdymas – kūrimas ir klausimų priskyrimas jiems.
- ✓ Klausimų valdymas – kūrimas ir variantų priskyrimas jiems.
- ✓ Egzamino valdymas – kūrimas, grupės ir testo priskyrimas
- ✓ Rezultatai – egzaminų rezultatai

Autoriaus posistemė atlieka šias funkcijas:

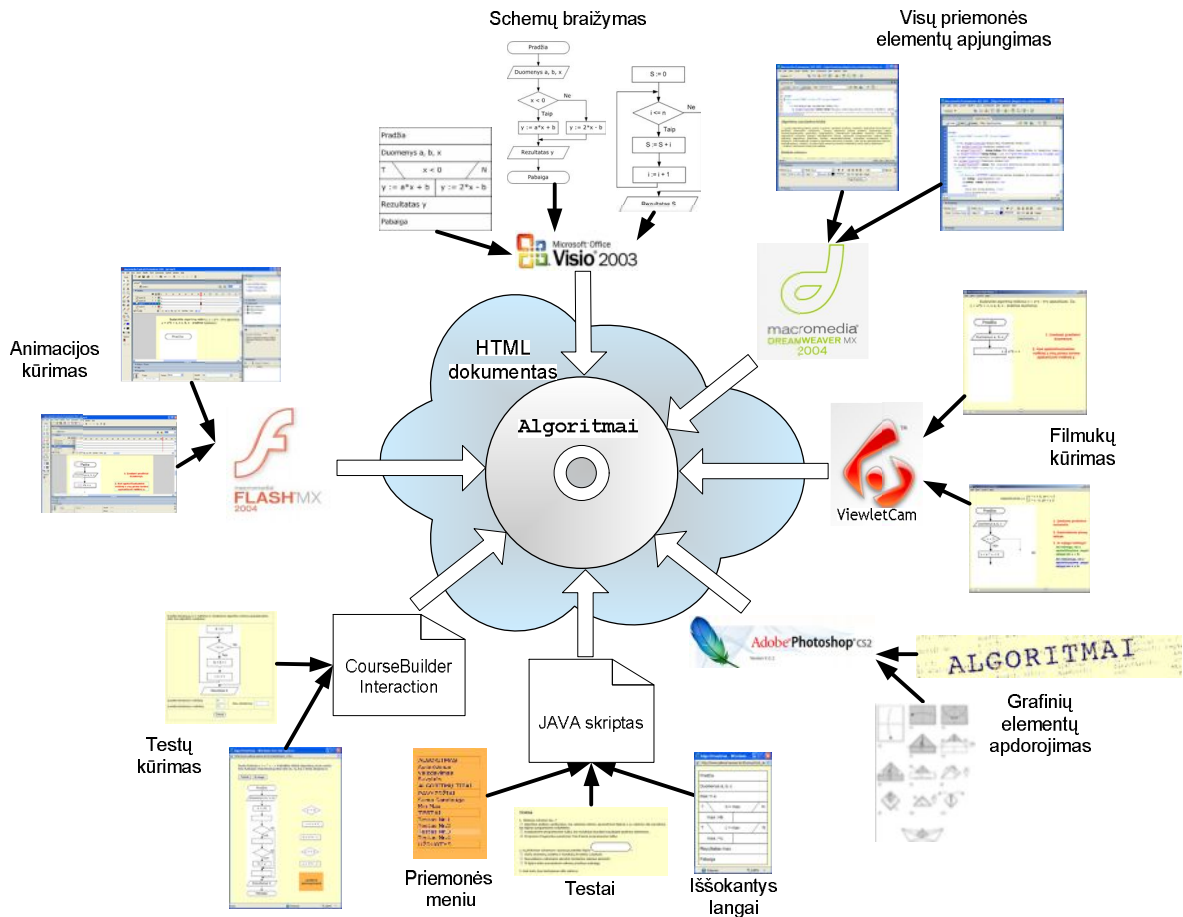
- ✓ Klausimų sukūrimas ir redagavimas.
- ✓ Klausimo įkėlimas į duomenų bazę tarnybinėje stotyje.
- ✓ Kuriant klausimus galima panaudoti šiuos elementus:

Priemonės kuriami objektai ir galimybės:

- ✓ Tekstiniai: vienos arba daugelio eilučių įvedimo laukai (angl. Label, text field, text area) ;
- ✓ Pasirinkimo: vieno arba keleto pasirinkimo laukai (angl. combo box, list box, radio button);
- ✓ Grafinius: linija, ovalas, užrašas, paveikslėlis, stačiakampis, lentelė, standartinė galimybė priskirti tekstą grafiniams objektams ir kt.;
- ✓ Objektų grupės;

Taip pat realizuotas itin nesudėtingas naujų komponentų įdiegimas pasitelkiant Java kalbą.

3.4. MKP „ALGORITMAI“ REALIZACIJOS TECHNINĖS PRIEMONĖS



18 pav. MKP „Algoritmai“ realizacijos techninės priemonės

MPK „Algoritmai“ bus kuriama HTML dokumento principu. Pačios priemonės kūrimui naudojama programą „Macromedia Dreamweaver MX 2004“. Ši programa pasirinkta, nes:

- Lengvai ir paprastai valdoma;
- Neprikuria bereikalingo programinio kodo (ko negalima pasakyti apie „Microsoft FrontPage“ programą);
- Yra galimybė į dokumentą talpinti Flash tipo bylas;
- Galima integruoti testų kūrimo programą „CourseBuilder Interaction“.

Blokinės schemas ir struktūrogramos bus braižomos programa „Microsoft Visio 2003“. Grafiniai objektai apdorojami su programa „Adobe Photoshop CS2“. Testai kuriami programos „Macromedia Dreamweaver“ priemone „CourseBuilder Interaction“, „JavaScript“ kalba ir programa „Macromedia Dreamweaver“ (šie testai paremti hipernuorodomis). Hipernuorodų principu bus sukurti ir aktyvūs pavyzdžiai. Su programomis „Macromedia Flash MX 2004“ ir „ViewletCam“ bus sukurti aktyvūs algoritmų sprendimo pavyzdžiai.

3.5. ALGORITMAVIMO PRAKTIKOS INFORMACINIO MODELIO SUDARYMAS

Norint sukurti gerą ir kokybišką mokamąją priemonę visų pirma reikia išsiaiškinti dažniausiai daromas mokinių klaidas. Sukurti taisykles, grafinius modelius tipiniams algoritmavimo uždaviniais spręsti.

Mokiniais buvo skirtos trijų tipų algoritmavimo užduotys:

- Tiesinio algoritmo;
- Šakoto (sąlygos) algoritmo;
- Ciklinio algoritmo.

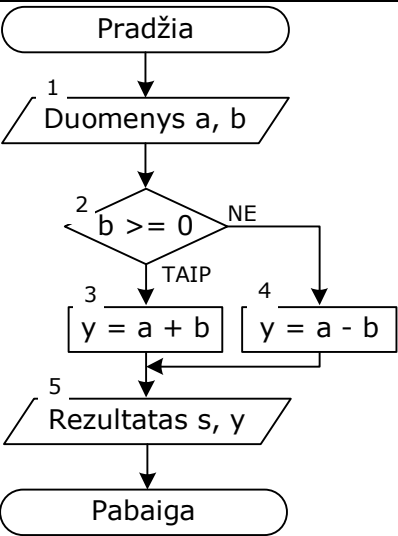
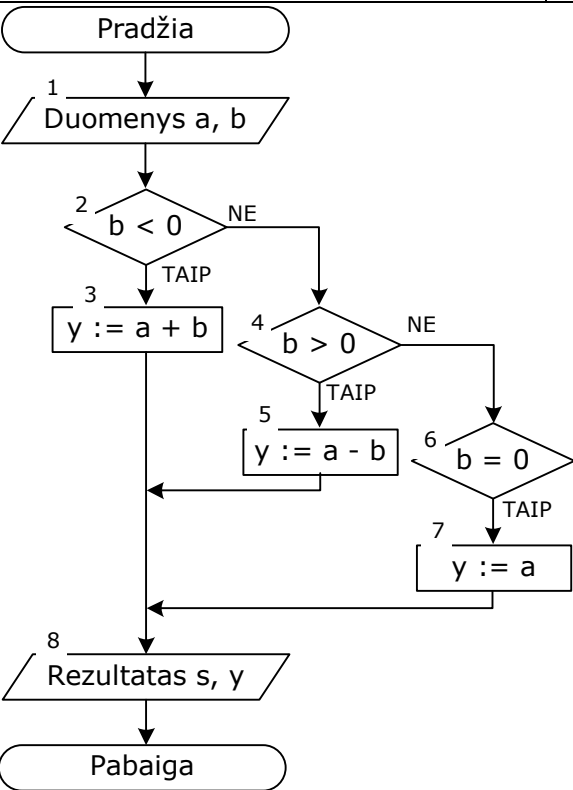
3.5.1. DAŽNIAUSIAI DAROMŲ KLAIDŲ ANALIZĖ

Tiesiniai algoritmai. Tai tokie algoritmai, kuriuose visi veiksmai atliekami nuosekliai vienas po kito be jokių alternatyvų ar veiksmų grupių kartojimo.

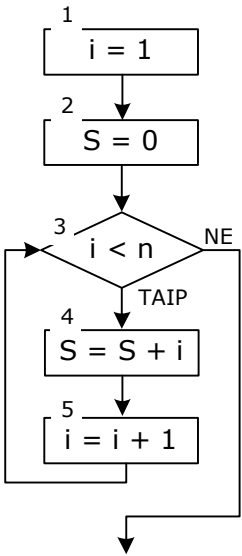
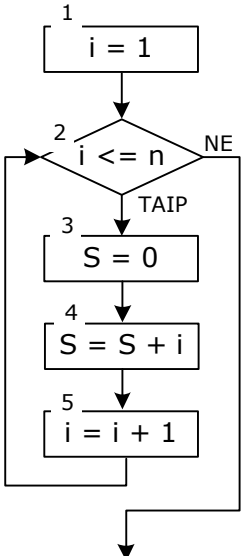
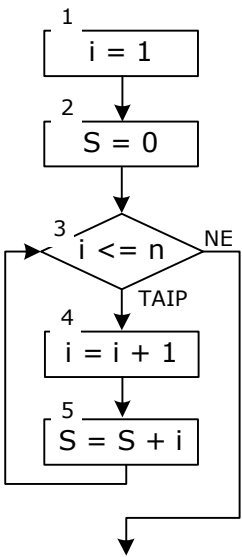
4 lentelė. Tiesinis algoritmas. Klaidų pavyzdžiai ir komentarai.

Algoritmas	Komentaras
<pre> graph TD Start([Pradžia]) --> Input[/1 Duomenys a, b, c/] Input --> Assign1[s := a + y] Assign1 --> Assign2[y := a * b + c] Assign2 --> Output[/4 Rezultatas s, y/] Output --> End([Pabaiga]) </pre>	<p><u>Užduotis.</u> Duota a, b ir c. Parašykite programą, kuri apskaičiuotų reiškinio $s = a + y$ reikšmę, kai $y = a * b + c$.</p> <p>Iš algoritmo fragmento matyti, kad pirma randama kintamojo s reikšmė, o tik po to kintamojo y reikšmė. 2 ir 3 blokus reikia sukeisti vietomis.</p>

Šakoti algoritmai. Tai algoritmai, kuriuose yra alternatyvūs sprendimo keliai, t.y. priklausomai nuo susidariusių sąlygų (ar tenkinama tikrinama sąlyga) skaičiuojama vienu ar kitu būdu.

Algoritmas	Komentaras
	<p><u>Užduotis.</u> Duoti a, ir b. Sudarykite algoritmą funkcijos y reikšmei rasti, kai $y = \begin{cases} a + b, & b < 0 \\ a - b, & b \geq 0 \end{cases}$.</p> <p>Reikia sukeisti 3 ir 4 elementus vietomis arba 2 bloko sąlygą keisti sąlyga $b < 0$.</p>
	<p><u>Užduotis.</u> Duoti a, ir b. Apskaičiuokite funkcijos y reikšmę</p> $kai \ y = \begin{cases} a + b, & b < 0 \\ a - b, & b > 0. \\ a, & b = 0 \end{cases}$ <p>Iš algoritmo matyti, kad čia nėra būtinas (jis nereikalingas) paskutinis sąlygos blokas 6.</p>

Cikliniai algoritmai. Cikliniuose skaičiavimo procesuose kai kurie veiksmai kartojami su vis naujomis kintamųjų reikšmėmis. Pasikartojančią skaičiavimo proceso dalį vadinsime ciklu. Uždavinio sprendimo algoritme gali būti daug ciklų. Vieno ciklo viduje gali būti kitas ciklas, o jame vėl naujas ciklas ir t.t.

Algoritmo fragmentas	Komentaras
 <pre> graph TD 1[i = 1] --> 2[S = 0] 2 --> 3{i < n} 3 -- TAIP --> 4[S = S + i] 4 --> 5[i = i + 1] 5 --> 3 3 -- NE --> Exit[] </pre>	<p><u>Užduotis.</u> Sudarykite algoritmą kuris susumuotų visus skaičius nuo 1 iki n.</p> <p>1) Šiuo atveju yra neteisingai nurodyta ciklo vykdymo sąlyga (blokas 3). Šis algoritmas susumuos skaičius nuo 1 iki n-1. Kad algoritmas būtų teisingas reikia pakeisti 3 bloko sąlyga į $i \leq n$.</p>
 <pre> graph TD 1[i = 1] --> 2{i <= n} 2 -- TAIP --> 3[S = 0] 3 --> 4[S = S + i] 4 --> 5[i = i + 1] 5 --> 2 2 -- NE --> Exit[] </pre>	<p>2) Šiame algoritme bloką 3 reikia iškelti prieš ciklą, nes kitaip visada kintamajam S priskiriamas 0 ir algoritmo rezultatas gaunamas neteisingas.</p>
 <pre> graph TD 1[i = 1] --> 2[S = 0] 2 --> 3{i <= n} 3 -- TAIP --> 4[i = i + 1] 4 --> 5[S = S + i] 5 --> 3 3 -- NE --> Exit[] </pre>	<p>3) Šiame algoritme bloką 4 reikia sukeisti vietomis su bloku 5, nes kitaip nebus susumuotas pirmas narys ir bus prisumuotas n+1 narys (gaunamas klaidingas rezultatas).</p>

<pre> graph TD 1[i = 1] --> 2[S = 0] 2 --> 3{i <= n} 3 -- TAIP --> 4[S = S + i] 4 --> 3 3 -- NE --> Exit(()) </pre>	<p>4) Šiame algoritme nėra keičiama ciklo kintamojo i reikšmė. Gaunamas amžinas ciklas. Tarp 3 ir 4 blokų reikia įterpti ciklo kintamojo keitimo bloką (sakinį) $i = i + 1$.</p>
<pre> graph TD 1[i = 1] --> 2[S = 0] 2 --> 3{i <= n} 3 -- TAIP --> 4[S = S * i] 4 --> 5[i = i + 1] 5 --> 3 3 -- NE --> Exit(()) </pre>	<p><u>Užduotis.</u> Sudarykite algoritmą kuris sudaugintu visus skaičius nuo 1 iki n.</p> <p>1) Šiame algoritme blogai yra parinkta pradinė sandaugos kaupimo kintamojo reikšmė (blokas 2). Sandaugos kaupimo kintamojo pradinė reikšmė turi būti lygi 1.</p>

Išanalizavus klaidas galima pastebėti, kad dažniausiai daromos trijų tipų klaidos:

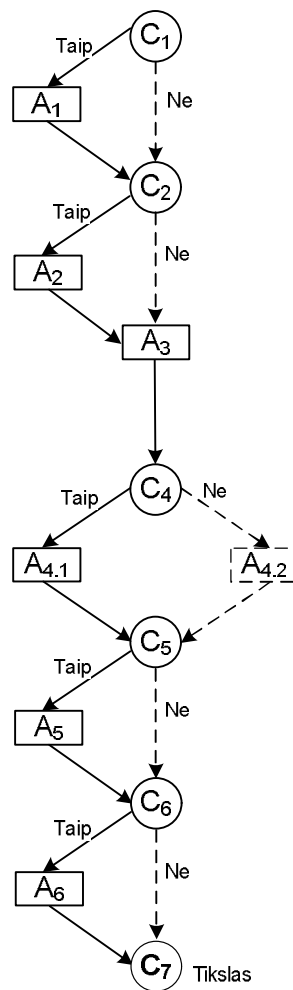
1. Algoritmo konstravimo klaidos;
2. Atskirų sakinių vietos nustatymo klaidos;
3. Sakinio konstrukcijos, aritmetinių veikslių, priskyrimo ir santykio operacijos klaidos.

3.5.2. ŠAKOTO ALGORITMO INFORMACINIAI MODELIAI

Grafiniam algoritmavimo taisyklių modeliui konstruoti naudosime kontekstinį grafą. Prieš konstruojant grafą, paprastam sąlygos sakiniui, išsiaiškinkime kokias būsenas jis gali įgyti:

1. Įvedami pradiniai duomenys;
2. Veiksmai prieš sąlygos sakinį;
3. Uždavinio sprendimo sąlyga;
4. Sąlygos reikšmės tikrinimas ir sąlygos sakinio veiksmai;
5. Veiksmai po sąlygos sakinio;
6. Duomenų išvedimas.

Išnagrinėjus sakinio būsenas sudarome sąlygos sakinio kontekstinį grafą:



19 pav. Sąlygos sakinio kontekstinis grafas.

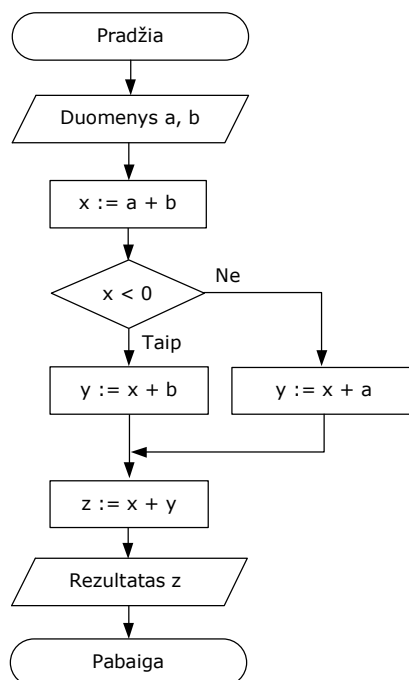
Čia A_i – veiksmo mazgas, $A_{i,j}$ – veiksmo mazgas, kurio vykdymas priklauso nuo uždavinio sprendimo sąlygos, C_i – konteksto (sąlygos) mazgas.

Eil. Nr.	Mazgų pavadinimai ir komentarai
1.	C_1 – ar reikia įvesti pradinis duomenis? A_1 – įvedame pradinis duomenis.
2.	C_2 – ar reikia atlikti veiksmus prieš sąlygos sakinį? A_2 – atliekame veiksmus prieš sąlygos sakinį.
3.	A_3 – pasirenkame uždavinio sprendimo sąlygą.
4.	C_4 – ar sąlygos reikšmė teisinga? $A_{4.1}$ – atliekame veiksmus kai sąlygos reikšmė teisinga. $A_{4.2}$ – atliekame veiksmus kai sąlygos reikšmė neteisinga.
5.	C_5 – ar reikia atlikti veiksmus po sąlygos sakinio? A_5 – atliekame veiksmus po sąlygos sakinio.
6.	C_6 – ar reikia išvesti rezultatus? A_6 – rezultatų išvedimas.
7.	C_7 – Tikslas.

Panagrinėkime kelis pavyzdžius:

Užduotis Nr1. Sudarykite algoritmą uždavinio, kuris apskaičiuotų kintamojo z reikšmę, kai $z = x + y$, $x = a + b$, o $y = \begin{cases} x + b, & \text{kai } x < 0, \\ x + a, & \text{kai } x \geq 0 \end{cases}$. Čia a ir b pradinės algoritmo reikšmės.

Uždavinio sprendimo algoritmas atrodys taip:



20 pav. Pirmo uždavinio sprendimo blokinė schema

8 lentelė. Sąlygos sakiny. Užduotis Nr1.

Kontekstinis grafas	Algoritmo būseną	Veiksmai
	B₁ 	C ₁ – ar reikia įvesti pradinis duomenis? (Taip). A ₁ – įvedame pradinis duomenis.
	B₂ 	C ₂ – ar reikia atlikti veiksmus prieš sąlygos sakinį? (Taip). A ₂ – atliekame veiksmus prieš sąlygos sakinį.
	B₃ 	A ₃ – pasirenkame uždavinio sprendimo sąlygą.
	B_{4.1} B_{4.2} 	C ₄ – ar sąlygos reikšmė teisinga? A _{4.1} – atliekame veiksmus kai sąlygos reikšmė teisinga. A _{4.2} – atliekame veiksmus kai sąlygos reikšmė neteisinga.
	B₅ 	C ₅ – ar reikia atlikti veiksmus po sąlygos sakinio? (Taip). A ₅ – atliekame veiksmus po sąlygos sakinio.
	B₆ 	C ₆ – ar reikia išvesti rezultatą sąlygos sakinio? (Taip). A ₆ – rezultatų išvedimo sakiny.

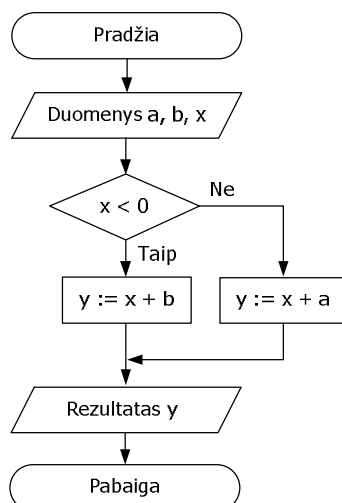
Kol pasiekiamas uždavinio tikslas yra atliekami šeši veiksmai (nuo B₁ iki B₆).

Šio uždavinio atveju yra apeinami visi kontekstinio grafo mazgai. Išskyrus, veiksmo mazgus A_{4.1} ir A_{4.2}. Čia bet kuriuo atveju yra praeinamas tik vienas veiksmo mazgas. Kuris mazgas praeinamas – priklauso nuo konkrečių kintamųjų *a* ir *b* reikšmių.

Panagrinėkime paprastesnę užduotį:

Užduotis Nr2. Sudarykite algoritmą uždavinio, kuris apskaičiuotų kintamojo *y* reikšmę,

kai $y = \begin{cases} x + b, & \text{kai } x < 0, \\ x + a, & \text{kai } x \geq 0 \end{cases}$. Čia *a*, *b* ir *x* pradinės algoritmo reikšmės.



21 pav. Antro uždavinio sprendimo blokinė schema

9 lentelė. Sąlygos sakiny. Užduotis Nr2.

Kontekstinis grafas	Algoritmo būseną	Veiksmai
	<p>B₁</p>	<p>C₁ – ar reikia įvesti pradinį duomenį? (Taip).</p> <p>A₁ – įvedame pradinį duomenį.</p>
	<p>B₂</p>	<p>C₂ – ar reikia atlikti veiksmus prieš sąlygos sakinį? (Ne)</p> <p>A₂ – atliekame veiksmus prieš sąlygos sakinį.</p>
	<p>B₃</p>	<p>A₃ – pasirenkame uždavinio sprendimo sąlygą.</p>
	<p>B_{4.1} B_{4.2}</p>	<p>C₄ – ar sąlygos reikšmė teisinga?</p> <p>A_{4.1} – atliekame veiksmus kai sąlygos reikšmė teisinga.</p> <p>A_{4.2} – atliekame veiksmus kai sąlygos reikšmė neteisinga.</p>
	<p>B₅</p>	<p>C₅ – ar reikia atlikti veiksmus po sąlygos sakinio? (Ne).</p> <p>A₅ – atliekame veiksmus po sąlygos sakinio.</p>
	<p>B₆</p>	<p>C₈ – ar reikia išvesti rezultatą sąlygos sakinio? (Taip)</p> <p>A₈ – rezultatų išvedimo sakiny.</p>

Šio uždavinio atveju grafo apėjimo kelias nesutampa su anksčiau nagrinėtu uždaviniu.

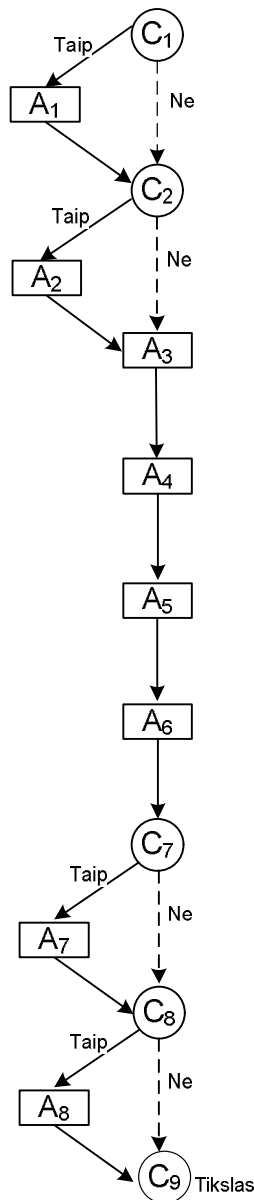
Neapeinami yra veiksmo mazgai A₂ ir A₅. Kitaip sakant, šame grafe nėra būsenų B₂ ir B₅.

3.5.3. CIKLINIO ALGORITMO INFORMACINIAI MODELIAI

Ciklinio algoritmo konstravimo eigoje galima išskirti šiuos veiksmus:

1. Pradinių duomenų įvedimas;
2. Veiksmai prieš ciklo sakinį;
3. Pradinės ciklo kintamojo reikšmės nustatymas;
4. Ciklo vykdymo sąlygos nustatymas;
5. Veiksmai ciklo sakinyje;
6. Ciklo kintamojo reikšmės keitimas
7. Veiksmai po ciklo sakinio;
8. Duomenų išvedimas

Išnagrinėjus ciklo algoritmo būsenas sudarome kontekstinį grafą:



22 pav. Ciklo sakinio kontekstinis grafas.

10 lentelė. Ciklo sakinio kontekstinio grafo detalus aprašymas.

Eil. Nr.	Mazgų pavadinimai ir komentarai
1.	C_1 – ar reikia įvesti pradinius duomenis? A_1 – pradinių duomenų įvedimo sakiny.
2.	C_2 – ar reikia atlikti veiksmus prieš ciklo sakinį? A_2 – atliekame veiksmus prieš ciklo sakinį.
3.	A_3 – pasirenkame pradinę ciklo kintamojo reikšmę.
4.	A_4 – pasirenkame ciklo pabaigos sąlygą.
5.	A_5 – veiksmai ciklo sakinyje
6.	A_6 – ciklo kintamojo reikšmės keitimas.
7.	C_7 – ar reikia atlikti veiksmus po ciklo sakinio? A_7 – atliekame veiksmus po ciklo sakinio.
8.	C_8 – ar reikia išvesti rezultatus? A_8 – rezultatų išvedimas.
9.	C_9 – tikslas

Panagrinėkime kelis pavyzdžius ir sudarykime šių algoritmų kontekstinius grafus ir blokines schemas.

Užduotis Nr.3. Duota funkcija $y = x^2 - x$, $x \in [m; n]$, x kinta žingsniu h ;

1. Atspausdinti visas funkcijos reikšmes tame intervale;
2. Apskaičiuoti reikšmių sumą tame intervale;
3. Rasti didžiausią funkcijos reikšmę;
4. Rasti mažiausią funkcijos reikšmę;
5. Rasti funkcijos vidurkį;
6. Apskaičiuoti reikšmių sandaugą tame intervale;

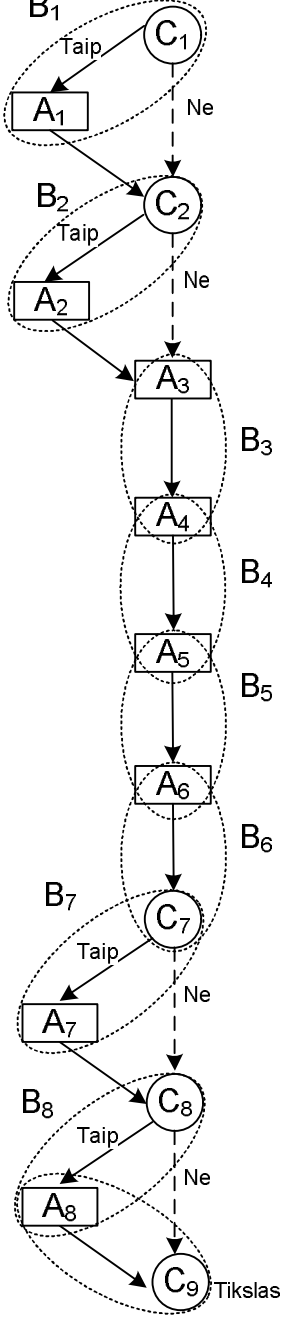
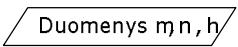
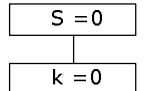
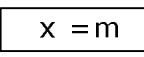
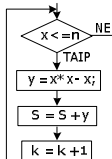
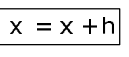
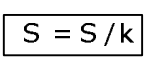
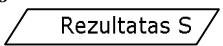
Šių užduočių sprendimo algoritmai pateikti 13 lentelėje. Sudarysime pirmos ir penktos užduočių sprendimo scenarijus. 11 lentelėje parodytas kontekstinio grafo apėjimo kelias.

11 lentelė. Pirmos užduoties ciklo sakinio kontekstinis grafas.

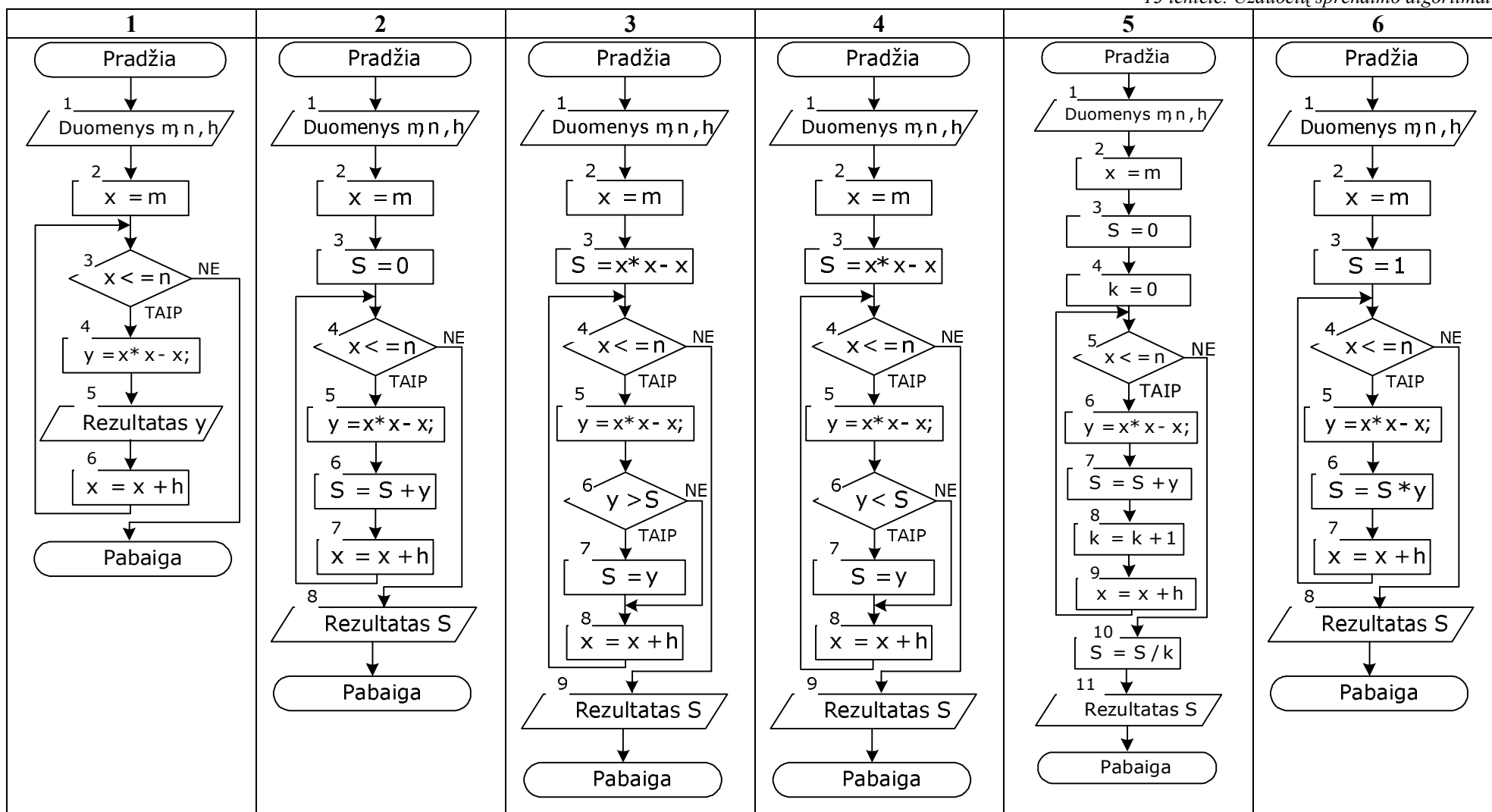
Kontekstinis grafas	Algoritmo būseną	Veiksmai
	B₁ 	C ₁ – ar reikia įvesti pradinį duomenį? (Taip) . A ₁ – pradinių duomenų įvedimo sakiny.
	B₂	C ₂ – ar reikia atlikti veiksmus prieš ciklo sakinį? (Ne) A ₂ – atliekame veiksmus prieš ciklo sakinį.
	B₃ 	A ₃ – pasirenkame pradinę ciklo kintamojo reikšmę
	B₄ $x < = n$	A ₄ – pasirenkame ciklo pabaigos sąlygą.
	B₅ 	A ₅ – atliekame veiksmus ciklo sakinyje.
	B₆ $x = x + h$	A ₆ – ciklo kintamojo reikšmės keitimas.
	B₇	C ₇ – ar reikia atlikti veiksmus po ciklo sakinio? (Ne) A ₇ – atliekame veiksmus po ciklo sakinio.
	B₈	C ₈ – ar reikia išvesti rezultatą po ciklo sakinio? (Ne) A ₈ – rezultatų išvedimo sakiny.
		C ₉ – tikslas.

Šio uždavinio atveju nebuvo apeinami veiksmo mazgai A₂ ir A₇ ir A₈.

12 lentelė. Penktos užduoties ciklo sakinio kontekstinis grafas.

Kontekstinis grafas	Algoritmo būsena	Veiksmai
	B₁ 	C ₁ – ar reikia įvesti pradinis duomenis? (Taip). A ₁ – pradinių duomenų įvedimo sakiny.
	B₂ 	C ₂ – ar reikia atlikti veiksmus prieš ciklo sakini? (Taip) A ₂ – atliekame veiksmus prieš ciklo sakini.
	B₃ 	A ₃ – pasirenkame pradinę ciklo kintamojo reikšmę
	B₄ $x \leq n$	A ₄ – pasirenkame ciklo pabaigos sąlygą.
	B₅ 	A ₅ – atliekame veiksmus ciklo sakinyje.
	B₆ 	A ₆ – ciklo kintamojo reikšmės keitimas.
	B₇ 	C ₇ – ar reikia atlikti veiksmus po ciklo sakinio? (Taip) A ₇ – atliekame veiksmus po ciklo sakinio.
	B₈ 	C ₈ – ar reikia išvesti rezultatą po ciklo sakinio? (Taip) A ₈ – rezultatų išvedimo sakiny.
C ₉ Tikslas		C ₉ – tikslas

Šios užduoties atveju yra apimami visi kontekstinio grafo veiksmo mazgai.

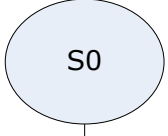
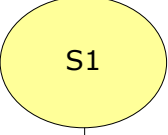
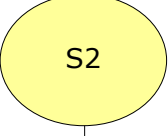
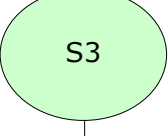
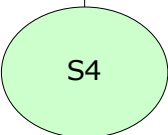
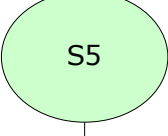
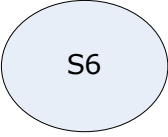


Galimas ir kitoks, paprastesnis, grafinis algoritmo sudarymo taisyklių aprašymas.
Galima naudoti veiksmų grafą su aprašytais taisyklėmis.

14 lentelė. Ciklo konstravimo veiksmų sekos grafas

Ciklo konstravimo veiksmų seka	Algoritmo dalis	Pavyzdžiai			
	Pradžia	Duomenys m, n, h			
	Pradinės kintamųjų reikšmės	$S = 0$	$S = 1$	$S := x * x - x$	
		$k := 0$			
	Ciklo parametras	$x = m$			
	Ciklo parametro ribos	$x \leq n$			
	Veiksmai cikle	Rezultatas y		$S = S + y$	
		$S := y$	$k := k + 1$	$S := S * y$	
	Ciklo parametro reikšmės keitimas	$x := x + h$			
Po ciklo	$S := S / k;$	Rezultatas S			

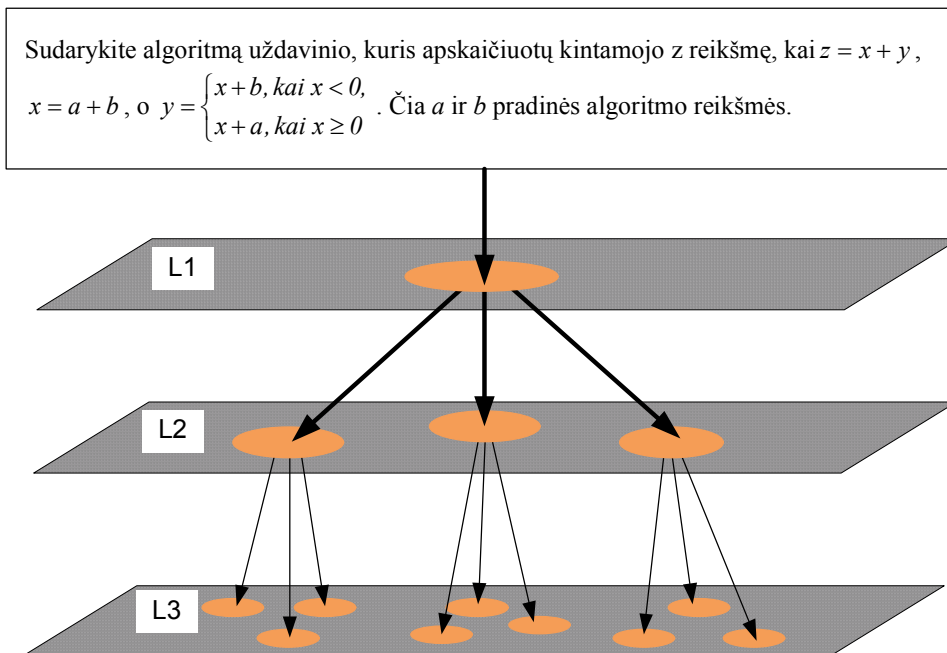
15 lentelė. Veiksmų sekos grafas su taisyklėmis.

Veiksmų seka	Taisyklės	Santykiai
Pradžia. Pradiniai duomenys 	Duomenys m, n, h	Yra dalis
Ciklo parengiamoji dalis 	Jei suma, tai	$S = 0$
	Jei sandauga, tai	$S = 1$
	Jei vidurkį, tai	$S = 0; k = 0$
	Jei min, max, tai	$S = x * x - x$
Ciklo parametras 	$x = m$	Yra dalis
Ciklo vykdymo sąlyga 	$x \leq n$	Yra dalis
Darbinė ciklo dalis 		$y = x * x - x$
	Jei apskaičiuojame reikšmes, tai	Rezultatas y
	Jei suma, tai	$S := S + y$
	Jei sandauga, tai	$S := S * y$
	Jei vidurkį, tai	$S := S + y;$ $k := k + 1$
	Jei min, tai	Jei $S > y$ tada $S = y$
	Jei max, tai	Jei $S < y$ tada $S = y$
Ciklo parametro reikšmės keitimas 	$x := x + h;$	Yra dalis
Veiksmai po ciklo. Pabaiga 	Jei suma, tai	Rezultatas S
	Jei sandauga, tai	Rezultatas S
	Jei min, tai	Rezultatas S
	Jei max, tai	Rezultatas S
	Jei vidurkis, tai	$s := s / k;$ Rezultatas S

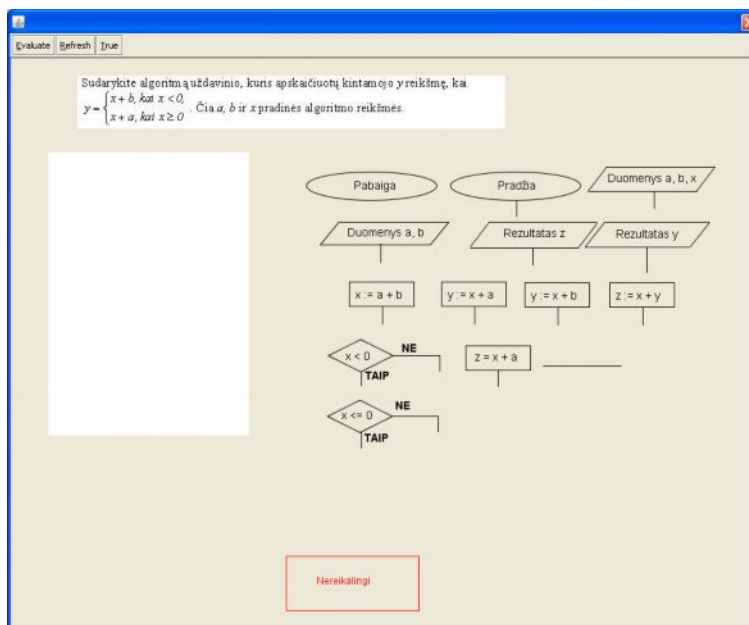
Uždaviniui Nr. 3 yra pateiktas ciklo konstravimo veiksmų grafas 14 lentelėje, o taisyklės pateiktos 15 lentelėje.

3.6. ALGORITMO KONSTRAVIMO MOKOMOSIOS PRIEMONĖS REALIZACIJA SU TESTTOOL

Atlikus mokinių klaidų analizę pastebėta, kad klaidos dažniausiai būna trijų tipų. Atsižvelgiant į tai TestTool programa buvo sukurti trijų sudėtingumo lygių testai.



23 pav. TestTool programa sukurtų testų sudėtingumo lygiai (pagal sukurtus pavyzdžius).



24 pav. Pirmo lygio užduotis

Pirmo lygio užduotyje mokiniams pateikiama sukonstruoti visą uždavinio sprendimo algoritmą. Testas sudarytas iš užduoties Nr.1 ir užduoties Nr.2 algoritmo elementų (12 ir 13 paveikslėliai).

Evaluate Refresh True

Sudarykite algoritmą uždavinio, kuris apskaičiuotų kintamojo y reikšmę, kai

$$y = \begin{cases} x + b, & \text{kai } x < 0, \\ x + a, & \text{kai } x \geq 0. \end{cases}$$

Čia a, b ir x pradinės algoritmo reikšmės.

Pradžia

Duomenys a, b, x

Rezultatas z	Duomenys a, b	Rezultatas y
$x := a + b$	$y := x + a$	$z := x + y$
$x \leq 0$		$z = x + a$
$x \geq 0$		
$x > 0$		

NE TAIP

$y := x + b$

Pabaiga

Nereikalingi

25 pav. Antro lygio užduotis

Antro lygio užduotyje algoritmo schema jau yra sukonstruota. Reikia sukelti tik kelis sakinius. Nereikalingus sakinius pašalinti.

Evaluate Refresh True

Sudarykite algoritmą uždavinio, kuris apskaičiuotų kintamojo y reikšmę, kai

$$y = \begin{cases} x + b, & \text{kai } x < 0, \\ x + a, & \text{kai } x \geq 0. \end{cases}$$

Čia a, b ir x pradinės algoritmo reikšmės.

Pradžia

Duomenys a, b, x

<	>	<=
NE	TAIP	

$x < 0$

$y := x + a$

$y := x + b$

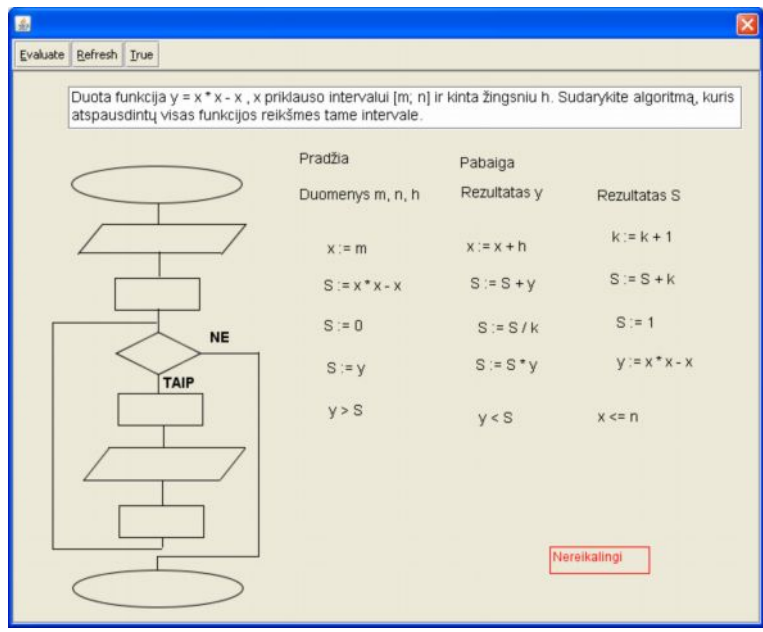
Rezultatas y

Pabaiga

Nereikalingi

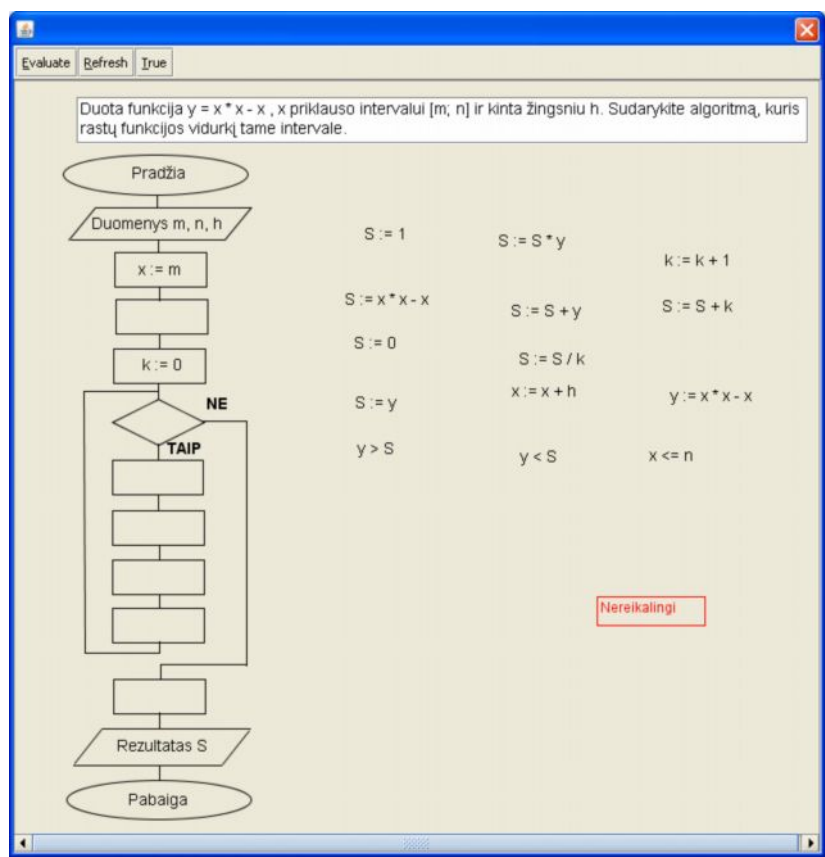
26 pav. Trečio lygio užduotis

Trečio lygio užduotyje algoritmas jau yra praktiškai sukonstruotas. Reikia „įrašyti“ tik trūkstamus ženklus ar kintamuosius.



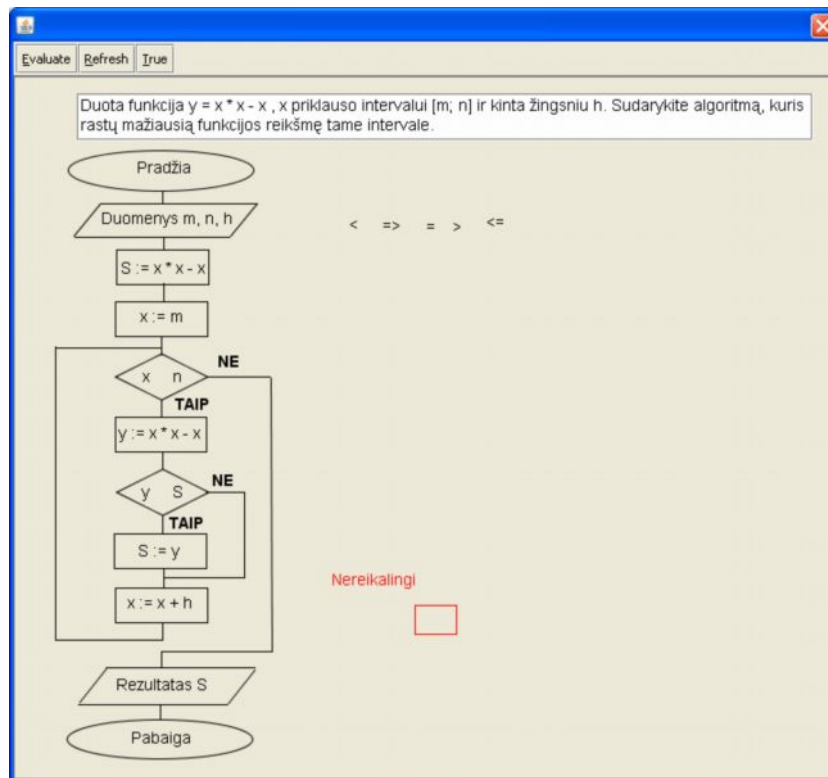
27 pav. Pirmo lygio ciklinė užduotis

Pirmo lygio ciklinė užduotis skiriasi nuo sąlygos sakinio užduoties tuo, kad čia jau yra pateikta algoritmo konstrukcija. Mokiniais reikia sukelti sakinius.



28 pav. Antro lygio ciklinė užduotis

Antro lygio užduotyje yra pateikta dalis algoritmo. Reikia pabaigti konstruoti algoritmą.



29 pav. Trečio lygio ciklinė užduotis

Trečio lygio užduotyje mokiniams reikia pabaigti konstruoti tik atskiras sakinių konstrukcijas.

Taip pat programa TestTool buvo realizuoti testai silpniau besimokantiems mokiniams. Šių algoritmų struktūrizavimo dokumentacija pateikiama 3 priede. Atlikinėdami testo užduotis mokiniai gaudavo taisykles (komentarų) konkrečiam algoritmo modeliui. Taisyklės pateikiamos 3 priede.

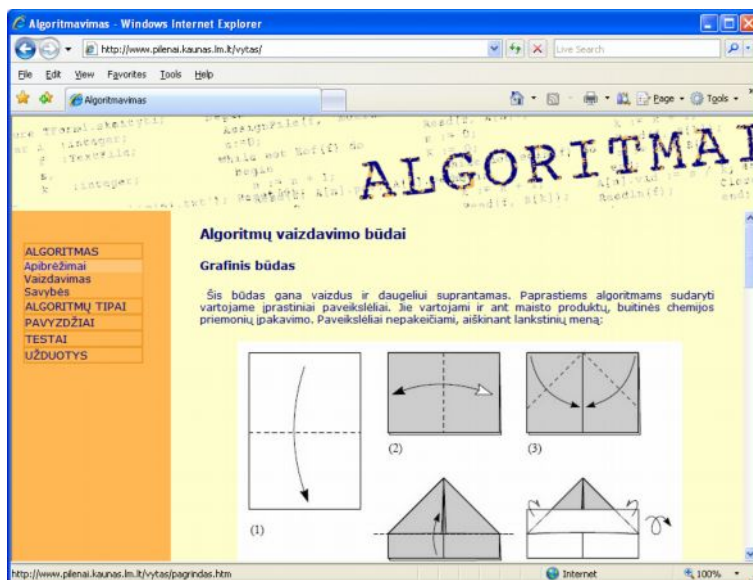
4. MOKYMO PRIEMONĖS VARTOTOJO DOKUMENTACIJA

4.1. MKP „ALGORITMAI“ APRAŠYMAS

Sukurta MKP „Algoritmai“ skirta 10 – 12 klasių moksleiviams, o taip pat ja galės naudotis ir 1 kurso studentai, kuriems yra dėstomas programavimo kursas. Mokymo priemonėje mokiniai supažindinami su algoritmo sąvoka, savybėmis, algoritmų vaizdavimo būdais, algoritmų tipais.

Ši mokymo priemonė kurta programa Macromedia Dreamweaver MX 2004. Pati mokomoji priemonė yra sukurta HTML dokumento principu naudojant freimus. Yra trys freimai:

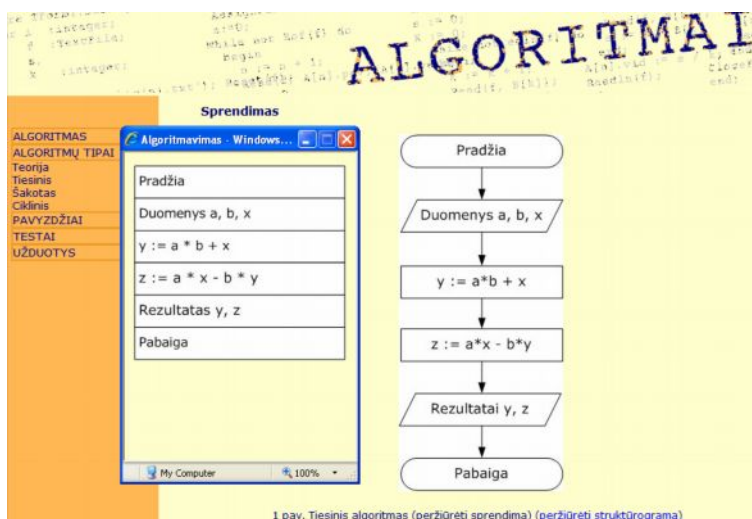
- ✓ Viršutinis – logotipui;
- ✓ Kairysis – valdymo meniu;
- ✓ Dešinysis – informacijai išvesti.



30 pav. Freimų išsidėstymas puslapyje

Mokomojoje priemonėje „Algoritmai“ mokiniams be teorinės–tekstinės medžiagos pateikiami ir algoritmų pavyzdžiai. Tai įvairios blokinės schemas ir struktūrogramos.

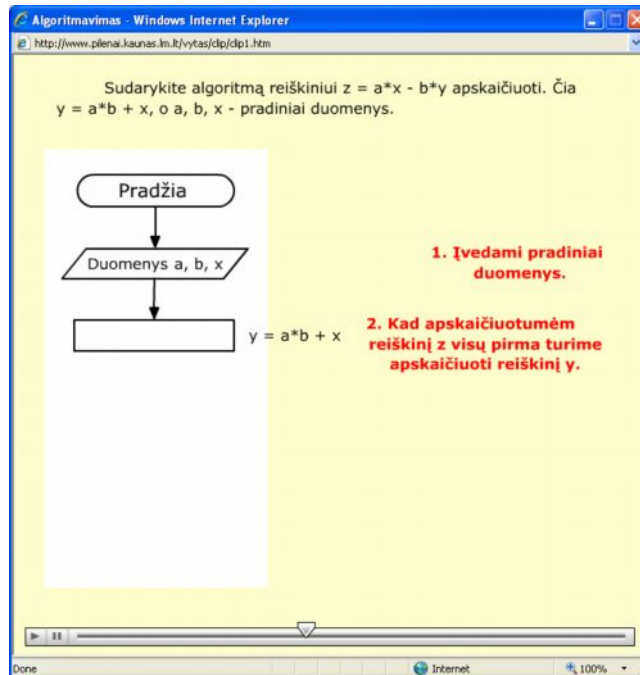
Algoritmų vaizdavimui naudojamos blokinės schemas. Šalia blokinės schemas mokinys gali peržiūrėti kaip atrodo algoritmo sprendimas struktūrogramoje. Tokiam medžiagos perteikimui panaudotas „iššokantis popup“ langas. Atsirandančiame lange yra išjungtas įrankių meniu, adreso juosta, užrakintas lango dydis. Tai garantuoja, kad mokiniui visada yra pateikiamas reikiamo dydžio langas, kurį jis sėkmingai gali „persitempti“ į norimą vietą. Kita priežastis yra ta, kad nubraižyti brėžiniai yra pakankamai dideli ir norima maksimaliai išnaudoti visą programinio lango plotą. Langas yra atveriamas maksimaliai mažas.



31 pav. „Popup“ tipo languose yra pateiktos uždavinio struktūrogramos

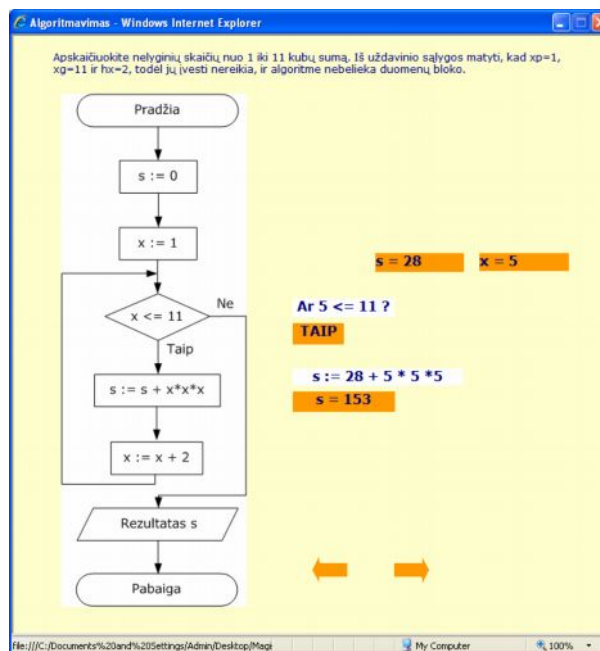
Be struktūrogramos peržiūrėjimo mokinys gali peržiūrėti algoritmo sprendimo filmuką, kuriame parodoma, kaip iš tekstinės užduoties yra konstruojamas algoritmas. Pateikiami taisyklės–komentarai (*Algoritmu tipai* → *Tiesinis* → *1 pav. peržiūrėti sprendimą*)

(Algoritmu tipai → Šakotas → 2 pav. peržiūrėti sprendimą). Šis filmukas yra sukurtas programa Macromedia Flash ir valdymo kokybei pagerinti nufilmuotas programa ViewletCam. Todėl jo peržiūrai kompiuteryje turi būti įdiegta Adobe Flash Player programa.



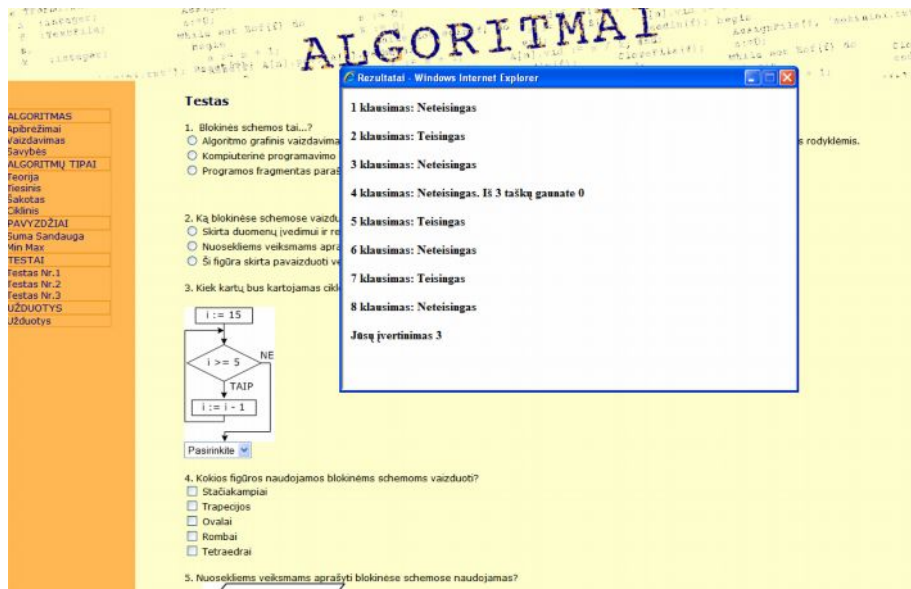
32 pav. Algoritmo sprendimo filmas

Yra pateikiami ir kai kurie ciklinių algoritmų sprendimo uždaviniai. Šiuose pavyzdžiuose mokinys gali valdyti ciklini algoritmą, stebėti kaip keičiasi kintamųjų reikšmės, kaip gaunamas algoritmo rezultatas (Pavyzdžiai → Suma Sandauga → 8 pav. peržiūrėti sprendimą) (Pavyzdžiai → Min Max → Užduotis. peržiūrėti sprendimą).



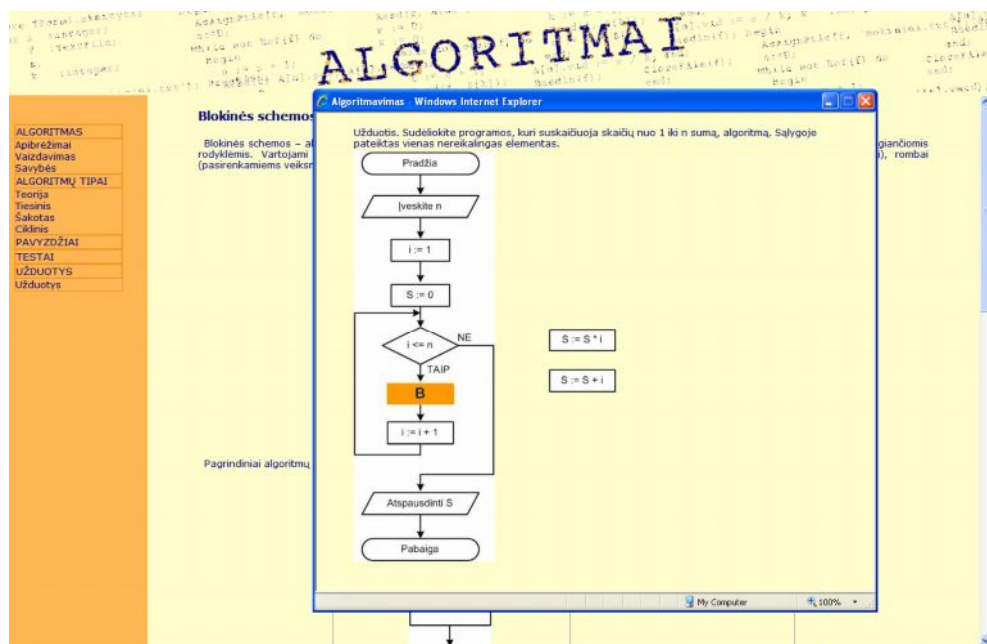
33 pav. Ciklinio algoritmo pavyzdys. Rodyklių pagalba mokinys gali sekti algoritmo žingsnius ir stebėti besikeičiančius rezultatus.

Savikontrolei yra pateikiami 4 testai. Pirmame teste reikia atsakyti į 8 pateiktus klausimus. Testo įvertinimas pateikiamas atskirame lange.



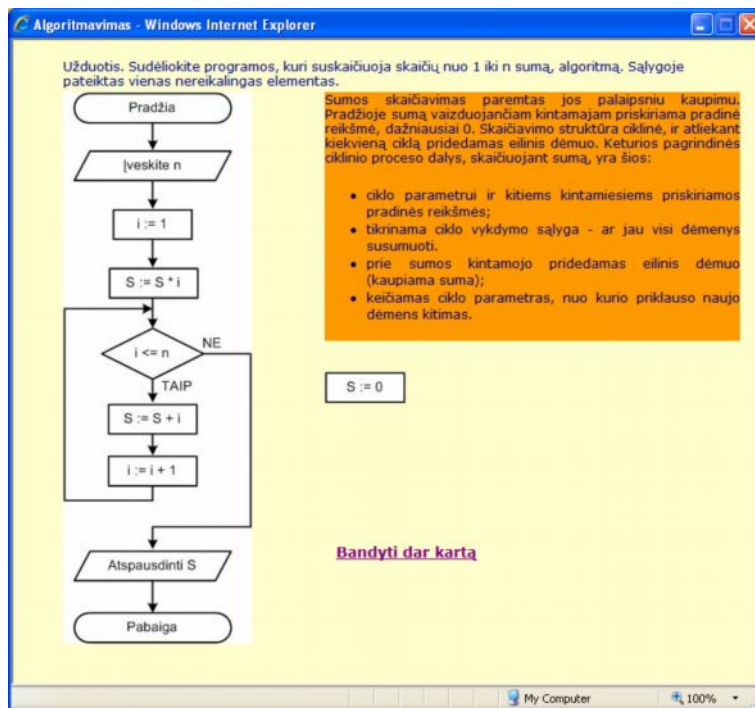
34 pav. Pirmo testo pavyzdys.

Antrame teste yra prašoma pabaigti dėlioti algoritmą (testas yra vykdomas paspaudžiant pele ant norimo elemento (vilkti nereikia)).



35 pav. Antro testo pavyzdys

Išsprendus užduotį mokiniui yra pateikiamas rezultatas. Jei užduotį jis atliko prastai (suklydo parinkdamas elementus) jam yra pateikiama teorinė medžiaga ir siūloma atlikti užduotį iš naujo.



36 pav. Antro testo rezultatas

Trečiame teste reikia suskaičiuoti kokį atsakymą duos algoritmas įvedus duomenis. Pradinius duomenis mokinys gali keisti pats

ALGORITMAS
ALGORITMŲ TIPAI
PAVYZDŽIAI
TESTAI
Testas Nr.1
Testas Nr.2
Testas Nr.3
UŽDUOTYS

Įveskite kintamųjų i ir n reikšmes ir remdamiesi algoritmo schema apskaičiuokite, koks bus algoritmo rezultatas.

S := 0

i <= n

Ne

Windows Internet Explorer
Tėisingas atsakymas 60. Jūsų pasirinkimas 452
OK

i := i + 1

Rezultatas S

Įveskite kintamojo i reikšmę: 4

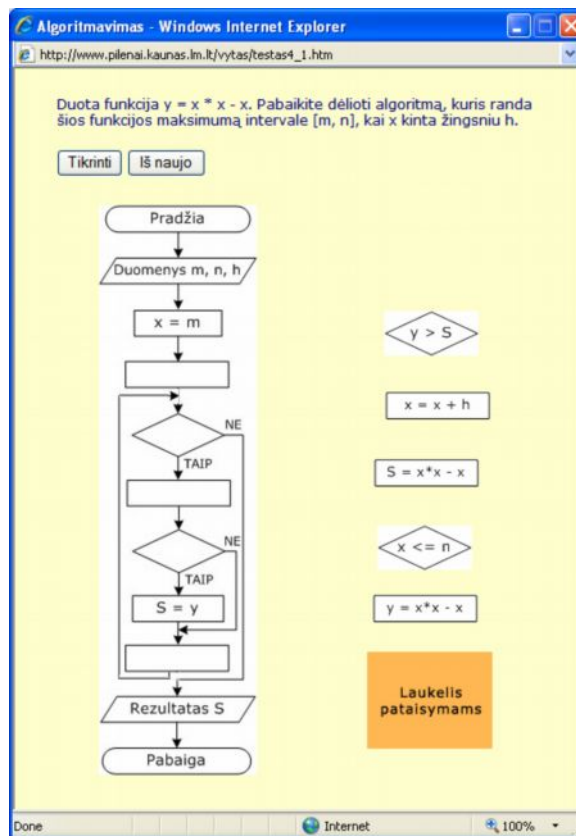
Įveskite kintamojo n reikšmę: 11

Jūsų atsakymas: 452

Tikrinā

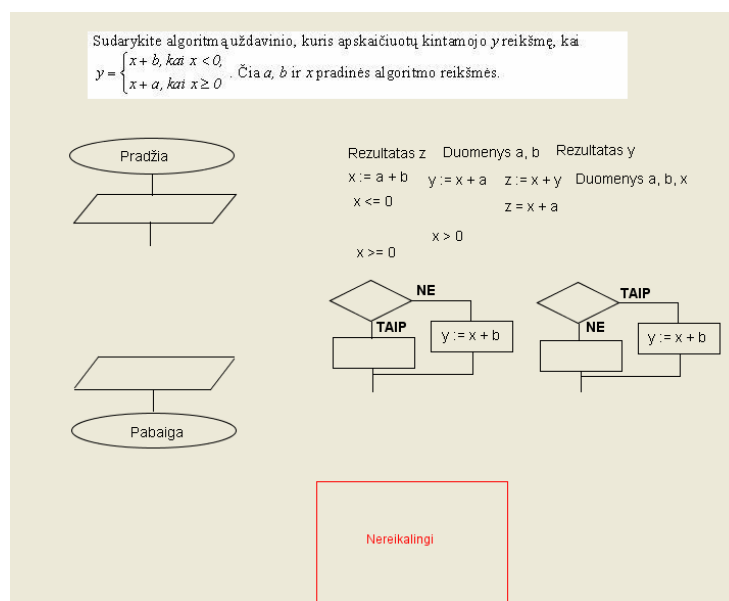
37 pav. Trečio testo pavyzdys.

Ketvirtas testas yra kurtas Macromedia Dreamweaver programos įskiepiu CourseBuilder Interaction. Šioje užduotyje mokinys turi pabaigti „sutampyti“ pateiktą algoritmą.



38 pav. Ketvirto testo pavyzdys

Kontroliniai MKP „Algoritmai“ testai yra sukurti programa TestTool 4.1.



39 pav. Testo TestTool programa pavyzdys

Testų veikimas pagrįstas objektų perkėlimo principu. Mokinys prisijungęs prie testo privalo sukonstruoti algoritmą. Testai yra trijų sudėtingumo lygių.

4.2. MKP „ALGORITMAI“ ĮDIEGIMAS

Kadangi MKP „Algoritmai“ yra sukurta HTML dokumento principu, jai papildamos instaliacijos nereikia. Svarbu, kad kompiuteryje būtų įdiegta operacinė sistema su interneto

naršykle. Programa „Algoritmai“ galima dirbti internete. Prisijungti reikia adresu <http://www.pilenai.kaunas.lm.lt/vytas>.

Norint MKP „Algoritmai“ patalpinti į savo serverį ją galima parsisiųsti adresu <http://www.pilenai.kaunas.lm.lt/vytas/Algoritmai.rar>. Atsiųstą bylą reikia išsiarchyvuoti. Programa paleidžiama įvykdžius bylą *index.htm*.

Papildomos programos:

Animaciniams filmukams žiūrėti reikalinga Adobe Flash Player programa. Ją galima parsisiųsti internetu adresu:

http://www.adobe.com/shockwave/download/download.cgi?P1_Prod_Version=ShockwaveFlash&promoid=BUIGP

Kad veiktų programa TestTool reikalinga Java aplinką. Ją galima parsisiųsti adresu: <http://www.java.com/en/download/installed.jsp>.

Testai buvo kuriami programa TestTool 4.1. Šios programos aplinką galima rasti adresu: <http://testtool.ktu.lt/index.php?action=5>. Išsamus TestTool4 sistemos vartotojo vadovas yra pateikiamas adresu <http://pilis.if.ktu.lt/tt4/web/index.php?page=begin>.

5. MKP „ALGORITMAI“ KOKYBĖS ĮVERTINIMAS

5.1. TYRIMO TIKSLAS IR UŽDAVINIAI

MKP „Algoritmai“ eksperimentinio tyrimo tikslas – supažindinti mokinius su mokomąja programa, sužinoti mokinių ir mokytojų nuomonę apie šią programą. Atliekant MKP „Algoritmai“ eksperimentinį tyrimą, mokiniai algoritmavimo kurso buvo mokomi šia mokomąja priemone. Šiame eksperimente dalyvavo 10 klasės mokiniai. Norint tiksliau įvertinti šią priemonę buvo atlikta anketinė apklausa. Anketa paruošta taip, kad galima būtų apklausti mokinius naudojant kompiuterį. Ruošiant anketą panaudoti išrinkimo langeliai, kurie leidžia pelės spragtelėjimu pažymėti išrenkamus elementus (anketos klausimai pateikiami 4 priede).

Šios apklausos uždavinys – sužinoti, koks mokinių požiūris į mokomąsias kompiuterines priemones, nustatyti sukurtos MKP silpnąsias ir stipriąsias puses. Anketa sudaro dvi dalys. Pirmoje dalyje pateikti bendro pobūdžio klausimai, kuriais siekiama sužinoti respondentų galimybes ir poreikius. Šios anketos dalies rezultatai, parodo ar moksleiviai kompiuterį mokymo tikslams naudoja tik mokykloje, ar gali pasinaudoti kompiuteriu kitose

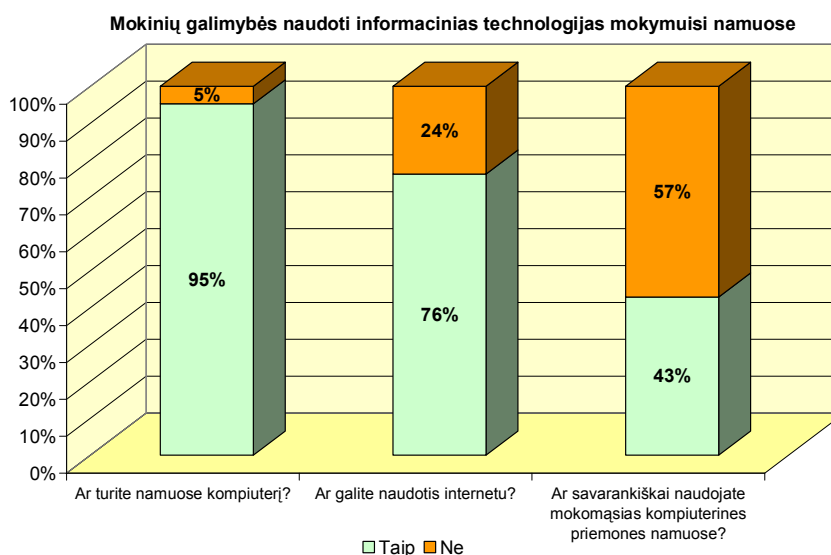
vietose. Antroji anketos dalis skirta sužinoti mokinių nuomonę apie MKP „Algoritmai“ ir ją įvertinti.

5.2. TYRIMO REZULTATAI

Atliekant tyrimą naudojami kiekybiniais ir kokybiniais tyrimo metodais.

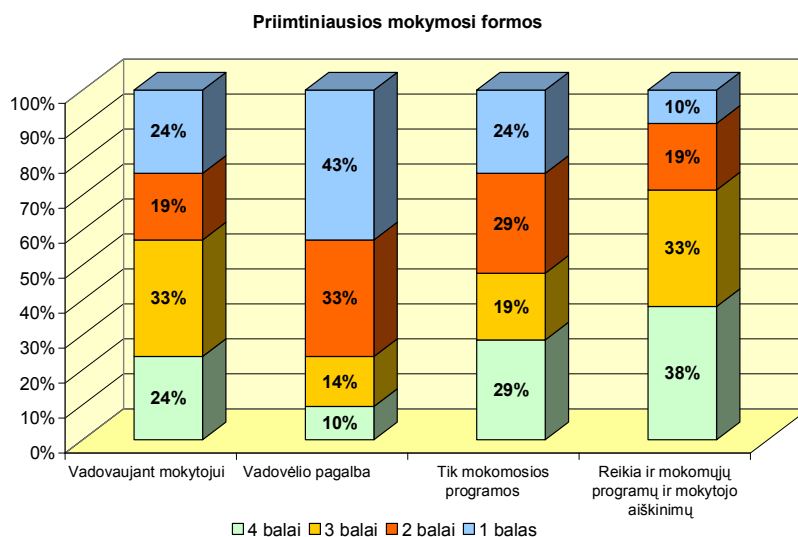
Reiškinio ar objekto vertinimo kriterijai gali būti kiekybiniai, kurie lengvai išmatuojami, lengvai išreiškiami skaičiais arba kokybiniai, kuriuos yra sudėtinga, o kartais ir neįmanoma įvertinti skaičiais.

Eksperte dalyvavo 21 mokinys. Baigus algoritmavimo mokymo kursą mokiniams reikėjo atsakyti į anketos klausimus.



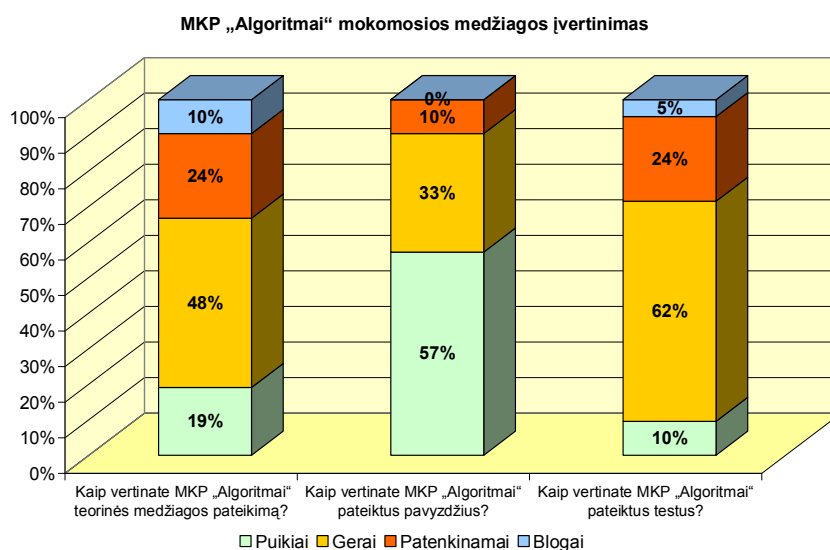
40 pav. IT naudojimo namuose galimybių diagrama

Iš atsakymų, pateiktų anketos pirmoje dalyje (žr. 40 pav.), galima matyti, kad dauguma mokinių turi galimybę namuose dirbti kompiuteriu. Net 76% apklaustųjų gali naudotis mokomosiomis priemonėmis patalpintoms internete, bet tik 43% savarankiškai namuose naudoja mokomasias kompiuterines priemones.



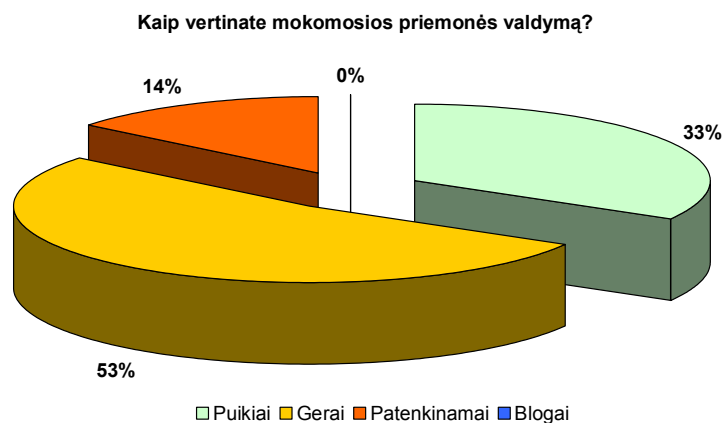
41 pav. Priimtinausių mokymosi formų diagrama

Diagramoje „priimtinausios mokymosi formos“ (žr. 41 pav.) išryškėja, kad priimtinausia mokymosi forma mokiniams yra IT technologijų taikymas mokymosi procese. Mokiniai nurodo, kad mokytojo pagalba pamokoje yra reikalinga. Iš pateiktų formų mokiniams labiausiai nepatinka dirbti su vadovėliu.



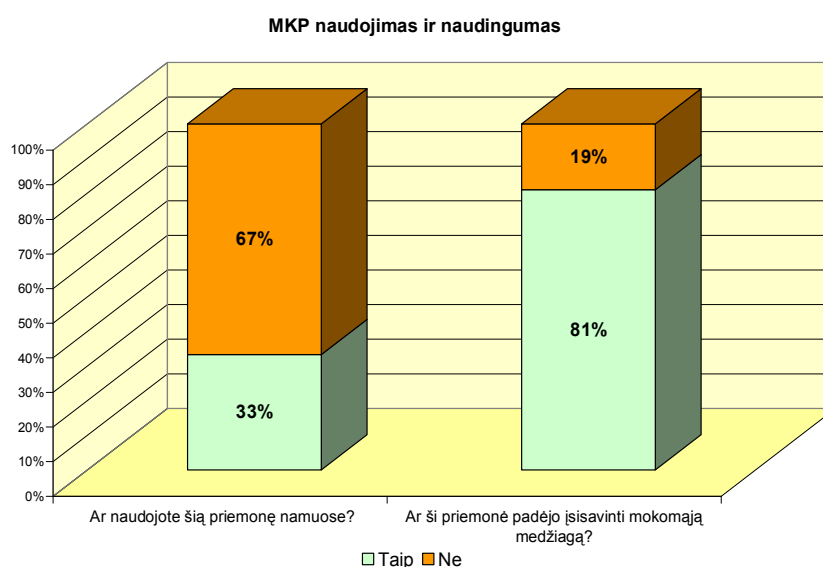
42 pav. MKP „Algoritmai“ mokomosios medžiagos vertinimo diagrama

MKP „Algoritmai“ teorinę medžiagą ir pateiktus testus (žr. 42 pav.) dauguma apklaustųjų įvertino gerai. Blogai įvertinusiems teorinės medžiagos pateikimą nepatiko, kad reikėjo daug skaityti. 57% mokinių labai patiko aktyvūs pavyzdžiai.



43 pav. MKP „Algoritmai“ valdymo vertinimo diagrama

MKP „Algoritmai“ valdymą (žr. 43 pav.) dauguma apklaustųjų įvertino gerai.



44 pav. MKP „Algoritmai“ naudojimo ir naudingumo diagrama.

Nors mokomąją priemonę namuose naudojosi tik 33% apklaustųjų (žr. 44 pav.), net 81% nurodė, kad ši priemonė padėjo įsisavinti mokomąją medžiagą.

Apibendrinimai

Dauguma mokinių turi galimybes naudotis mokomosiomis kompiuterinėmis priemonėmis namuose.

Iš gautų duomenų galime pamatyti, kad mokiniai retai naudojasi vadovėliais ir mieliau renkasi mokomasias priemones, kurios padaro pamoką įdomesne. Priimtinausia mokymosi forma mokiniams yra mokomųjų programų naudojimas kartu su mokytojo paaiškinimais.

MKP „Algoritmai“ mokinių įvertinta buvo gerai. Dalis apklaustųjų ja naudojosi namuose savo žinių įtvirtinimui.

5.3. MKP „ALGORITMAI“ ĮVERTINIMAS PAGAL PASIRINKTUS VERTINIMO KRITERIJUS

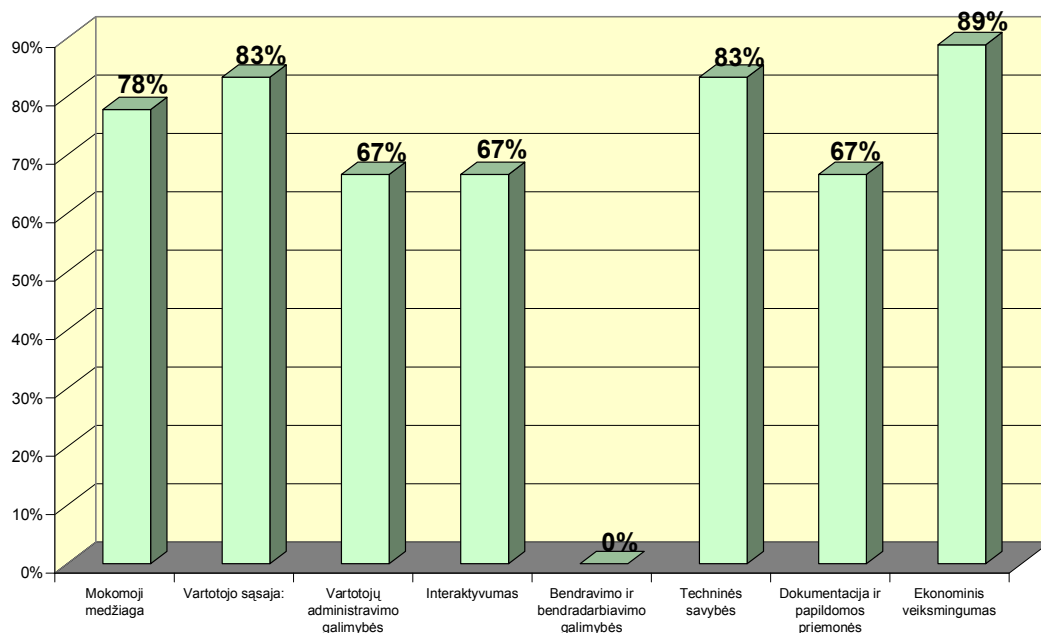
Pagal 1 lentelėje pateiktus kriterijus atlikta MKP „Algoritmai“ kokybės vertinimo analizė. Buvo pasirinktas ekspertinis vertinimas. Mokamąją priemonę vertino tie patys vertintojai, tik dabar pateiktą įvertinimą jie dar privalėjo pakomentuoti. Vertinimo rezultatai pateikti 16 lentelėje.

16 lentelė. MKP „Algoritmai“ įvertinimas.

Kriterijai	Balas	Komentarai
Mokomoji medžiaga		
Teorinė dalis	3	Pateikta teorinė medžiaga detali ir lengvai suprantama. Pavyzdžiai atitinka teorinę medžiagą. Savikontrolės klausimuose pateiktos nuorodos į teisingus atsakymus. Mokomosios užduotys atitinka pasirinktą temą. Kontrolinėse užduotyse programa pateikia žinių įvertinimus. Padarius klaidą pateikiama taisyklė, kaip mokinys privalėjo atlikti užduotį. Yra sudaryta galimybė grafinėje sąsajoje modeliuoti algoritmus. TestTool testuose yra galimybė kontroliuoti užduočių įvairovę, sudėtingumo lygius.
Pavyzdžiai	2	
Savikontrolės klausimai	2	
Mokomosios užduotys	2	
Kontrolinės užduotys	3	
Programų modeliavimas	3	
Klaidų analizė	1	
Užduočių įvairovė (užduotys skirtingos)	2	
Klaidos (pateikus klaidingą atsakymą, programa jį užskaito kaip teisingą)	3	
Vartotojo sąsaja:		
Patogumas	3	Vartotojo sąsaja yra grafinė. Valdymas vykdomas pele.
Ergonomiškumas	2	
Vartotojų administravimo galimybės		
Vartotojų administravimo priemonės	2	TestTool programoje sprendžiant sukurtus testus yra apskaitos apie vartotojus galimybė. Galima individualizuoti mokymosi procesą.
Mokymosi proceso individualizavimo galimybės	2	
Asmeninių duomenų ir informacijos saugumas.	2	
Interaktyvumas		
Grižtamasis ryšys	2	MKP „Algoritmai“ yra pritaikyta savarankiškam darbui. Mokytojas gali matyti mokinių rezultatus, analizuoti klaidas.
Pranašumas prieš tradicines mokymo priemones	2	
Savarankiško darbo galimybės	3	
Papildomos mokymosi priemonės	1	
Mokymosi organizavimo priemonės	2	

Bendravimo ir bendradarbiavimo galimybės		
Sinchroninės priemonės	0	MKP „Algoritmai“ bendravimo ir bendradarbiavimo priemonių nėra.
Asinchroninės priemonės	0	
Techninės savybės		
Veikimo stabilumas	2	Programa veikia stabiliai. Nereikalaujama didelių kompiuterio resursų.
Darbo sparta	2	
Nesudėtingas programos įdiegimas	3	
Suderinimas su esama technine ir sistemine įranga	3	
Dokumentacija ir papildomos priemonės		
Bendra informacija	2	Pateikta mokomosios priemonės bendra informacija. Yra įdiegimo, naudojimo ir valdymo galimybių aprašymas.
Įdiegimo aprašymas	3	
Naudojimo ir valdymo galimybių aprašymas	3	
Metodinės rekomendacijos ir papildomos metodinės priemonės mokiniui	1	
Metodinės rekomendacijos ir papildomos metodinės priemonės mokytojui	1	
Taisyklingas stilius ir kalba	2	
Ekonominis veiksmingumas		
Kainos prieinamumas	3	Ši programa yra nemokama. Sudėtingų papildomų išteklių programos įdiegimas nereikalauja.
Papildomos sąnaudos ir ištekliai programos įdiegimui	3	
Programą galima nuolat naudoti ir atnaujinti esant reikalui	2	

MKP „Algoritmai“ įvertinimas



45 pav. MKP „Algoritmai“ įvertinimas procentais

MKP „Programavimo pradmenys“ kriterijus „Mokomoji medžiaga“ įvertintas 70%, o MKP „Algoritmai“ – 78%. Kriterijus „Vartotojo sąsaja“ MKP „Programavimo pradmenys“ įvertintas 100%, o MKP „Algoritmai“ – 83%. Kriterijus „Interaktyvumas“ MKP „Programavimo pradmenys“ įvertintas 60%, o MKP „Algoritmai“ – 67%.

5.4. EKSPERTŲ APIBENDRINIMAI

Mokytojai, vertinę MKP „Algoritmai“, pateikė tokius atsiliepimus:

Privalumai:

- Programoje pateiktos temos glaudžiai susietos su informacinių technologijų ugdymo programomis.
- Sukurta patraukli, tikslinga vizualizacija, kelianti mokinių mokymosi motyvaciją.
- Programoje sukurtos interaktyvumo galimybės leidžia individualiai organizuoti mokymosi temas ir tempą.
- Nuorodos į papildomus informacinius langus sudaro galimybę individualiai pasirinkti mokymosi gylį.
- Programoje aiškiai ir tikslingai organizuotas grįžtamasis ryšys.
- Galimybė pasirinkti mokymosi vietą ir laiką (*programa patalpinta web serveryje*) leidžia organizuoti papildomą mokymąsi.

Trūkumai:

- Reikėtų į programą įtraukti daugiau pavyzdžių.
- Reikėtų programoje pateikti daugiau saitų į alternatyvių temų mokymo(si) resursus internete.

6. DARBO REZULTATAI, IŠVADOS

1. Remiantis LR Švietimo ir mokslo ministerijos mokyklų aprūpinimo mokomosiomis kompiuterinėmis priemonėmis (MKP) tvarkos aprašu suformuluoti vertinimo kriterijai, kurie apima mokamąją medžiagą, vartotojo sąsają, administravimo galimybes, interaktyvumą ir technines savybes. Pagal šiuos kriterijus įvertintos Lietuvos mokyklose informacinių technologijų pamokose dažniausiai naudojamos 5-ios MKP, kriterijai panaudoti magistriniame darbe atliktų realizacijų įvertinimui.
2. Mokymo priemonė turėtų turėti platų galimybių spektrą, tačiau atlikus analizę pastebėta, kad vienos priemonės yra daugiau orientuotos praktinėms užduotims, kitose pateikiama daugiau teorinės medžiagos.
3. Darbe realizuotos dvi programinės priemonės:
 - a. MKP „Algoritmai“ skirta kuo platesnio galimybių spektro sudarymui ir tyrimui. Jos komponentų įgyvendinimui pasirinktos technologijos, kurios leidžia kurti HTML dokumentus, animacijos objektus, filmuoti kompiuterio ekrane vykstančius veiksmus, kurti testus, diagramas, blokines schemas, apdoroti grafinius elementus ir technologija interaktyvių internetinių puslapių realizacijai.
 - b. Algoritmų modeliavimo užduočių įgyvendinimui panaudotos 3 technologinės priemonės: HTML principas, CourseBuilder Interaction ir TestTool Author programos.
4. MKP „Algoritmai“ struktūra apima šias temas: algoritmai, algoritmų tipai, pavyzdžiai, testai, užduotys. Joje sukurtos galimybės individualiai rinktis mokymosi sudėtingumo lygį ir tempą. Mokomosios priemonės dokumentacijoje aprašytos galimybės, įdiegimas ir reikalavimai techninei įrangai.
5. Ekspertiniame tyrime MKP „Algoritmai“ įvertinti pagal pasirinktus kriterijus. Palyginus su kitomis priemonėmis, ji išsiskyrė paprastumu, intuityviai suprantama vartotojo sąsaja, informatyvumu, aiškumu, glaustai pateikta informacija. MKP „Algoritmai“ eksperimentinio taikymo pamokose metu 81% respondentų pastebėjo, kad priemonė padėjo įsisavinti mokamąją medžiagą. Pastebėtas motyvacijos pagerėjimas bendradarbiaujant su mokytojais bei kitais moksleiviais.
6. Algoritmavimo praktikos informacinio modelio sudarymui atlikti šie tyrimai:
 - a. Išanalizuotos mokinių daromos klaidos;

- b. sudaryti šakoto ir ciklinio sakinių informaciniai modeliai remiantis kontekstiniais grafais;
7. Pagrindinė algoritmų modeliavimo užduočių kūrimo priemonė pasirinkta programa Testtool Author, kuria realizuotos trijų sudėtingumo lygių užduotys. Algoritmų modeliavimo užduočių kūrimas TestTool Author programa pareikalavo mažiausiai darbo ir laiko išteklių ir pavyko pasiekti gerą modelių kokybę. Kurti algoritmų modeliavimo užduotis HTML principu ar CourseBuilder Interaction priemonėmis nėra tikslinga, nes tai reikalauja didelių laiko ir darbo sąnaudų.

LITERATŪRA

1. *Bendrosios ir mokomosios atvirosios programos / Parengė Valentina Dagienė ir Tatjana Jevsikova. V.: Žara, 2004. 88 p.*
2. *Blonskis J., Baniulis K. ir kiti. Programavimas. K.: Technologija, 2000. 380 p.*
3. *Burgis B., Butleris R., Jurgaitis M., Kulikauskas A., Leitanas R., Limanauskas I., Listopadskis N., Lukšys G., Marcinkevičius R., Starkus B., Šakys V., Toleikis Š. Kompiuterija. K.: Naujas lankas, 2000. 587 p.*
4. *Dagienė V. Informacinės technologijos IX–X klasėms. 2 dalis. V.: TEV, 2003. 255 p.*
5. *Dagienė, V. Informatikos pradžios. II dalis. V.: TEV. 1998. 104 p.*
6. <http://aldona.mii.lt/pms/Logo/tekstai/pamokos/nulinis.html> (žiūrėta 2008 vasario 5)
7. http://iktgu.projektas.lt/Konf_2006papers/iktgu_konf06_095Railiene.pdf (žiūrėta 2008 kovo 28)
8. http://ktk.kauko.lt/~esplk/images/stories/mkrd_metodika.doc (žiūrėta 2008 kovo 29)
9. <http://lt.wikipedia.org/wiki/JavaScript> (žiūrėta 2008 vasario 12)
10. http://www.daukantas.lt/edu/Info/Vadovelis/ALGOR/ALGO_00.HTML (žiūrėta 2007 vasario 24)
11. <http://www.emokykla.lt/admin/file.php?id=381> (žiūrėta 2008 vasario 5)
12. <http://www.emokykla.lt/admin/file.php?id=338> Mokslinio tyrimo darbo mokomųjų kompiuterinių priemonių ir virtualiųjų mokymosi aplinkų profesinio mokymo srityse diegimas. Ataskaita. Darbo vadovė: Valentina Dagienė. V.: 2005
13. <http://www.leidykla.vu.lt/inetleid/inf-mok/25/str12.html> (žiūrėta 2007 kovo 25)
14. *J.Adomavičius, M.Pečiukonienė, V.Sekliuckis. Informatika 2. Algoritmai ir jų įdiegimas. K.: Technologija, 2002. 84 p.*
15. *Pagrindinio ugdymo informacinių technologijų bendroji programa. Patvirtinta Lietuvos Respublikos švietimo ir mokslo ministro 2005 m. liepos 14 d. įsakymu Nr. ISAK-1432*
16. *Rajeckas V. Mokymo organizavimas. K.: Šviesa, 1999 – 382.*
17. *Rutkauskienė D., Lenkevičius A., Targamadžė A., Volungevičienė A., Pociūtė E., Dėmenienė A., Kelmienė V. Nuotolinio mokymosi dėstytojo vadovas.– K.: „Technologija“, 2007. – 188 p.*
18. *Tumasonis V. Paskalis ir Turbo Paskalis 7.0. – V.: Aušra, 1993. – 381 p.*

PRIEDAI

Priedas Nr. 1. MKP priemonių aprašymas.

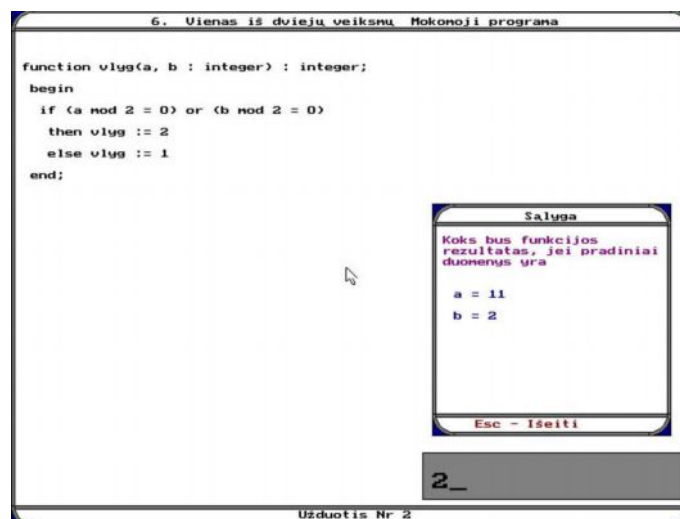
PASKMOK

Autoriai: S.Rupeikis, A.Verseckas. Kuršėnai 1995.

Programa pateikia mokiniams atlikti keletą algoritmų arba jų fragmentų ir patikrina gautų rezultatų teisingumą. Nagrinėjama 14 temų:

1. Reiškinių reikšmės.
2. Priskyrimas.
3. Procedūra.
4. Funkcija.
5. Loginiai duomenys.
6. Vienas iš dviejų veiksmų.
7. Vienas iš kelių veiksmų.
8. Sudėtinis sakiny.
9. Ciklas WHILE.
10. Ciklas FOR.
11. Ciklas cikle.
12. Rekursija.
13. Programa.
14. Masyvas.

Pasirinkus norimą temą kompiuterio ekrane atsiranda pasiūlymas pasirinkti mokomąjį arba kontrolinį temos variantą. Kiekvienu atveju mokiniui pateikiami nagrinėti 5 pratimai, kuriuose reikia atlikti algoritmus arba bent kelis sakinius, tai yra nustatyti rezultatus, turint pradinis duomenis.



46 pav. PaskMok programa. Užduotis

Mokomuosiuose programos variantuose, algoritmai ar jų fragmentai yra vienodi visiems mokiniams, tačiau pradiniai duomenys parenkami kaip atsitiktiniai skaičiai (savaime suprantama atitinkantys tam tikras ribas ar reikalavimus). Tai, kad pradiniai duomenys kiekviename kompiuteryje skirtingi, verčia mokinius pratimus atlikti savarankiškai, nesižvalgant į draugus. Užduotys išdėstytos sunkėjimo tvarka.

Nr	Teisingas	Pateiktas
1.	1	1
2.	0	0
3.	25	4
4.	19	2
5.	1	5
	2	5
	2	4

Ivertinimas : 4

Atliko Jonas Jonaitis

47 pav. PaskMok programa. Kontrolinės užduoties įvertinimas

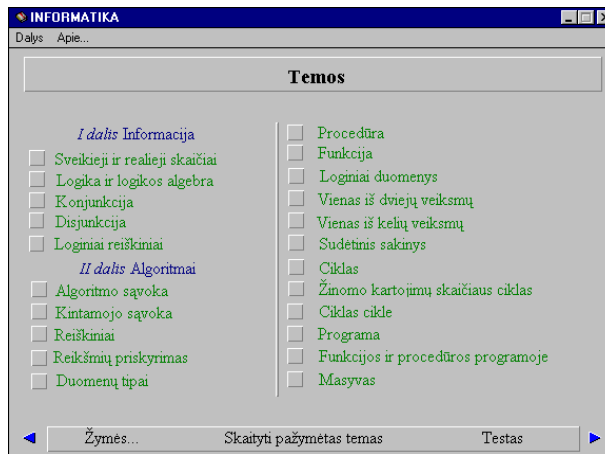
Mokiniai atlikę algoritmą, įveda rezultatus, o kompiuteris patikrina, ar įvesti duomenys teisingi. Jei mokinys suklydo, ekrane apie tai atsiranda pranešimas, ir parodomas rezultatas, kurį turėjo gauti. Po to jam siūloma dar kartą atlikti tą patį algoritmą, tik jau su kitais pradiniais duomenimis. Ir taip tol, kol mokinys supras algoritmo esmę ir jį atliks gerai. Naudojant kompiuterinę programą mokinys gali mokytis ir mąstyti savarankiškai, jam tinkamu tempu.

Programa tikrai naudinga mokant pirmųjų temų, padeda gerai suprasti ir įsisavinti mokiniams neįprastas operacijas div ir mod, priskyrimą. Palengvina ir sunkesniųjų temų, pradedant ciklu while, mokymąsi.

INFORMATIKA

Autorius A.Bačiulis. 1997

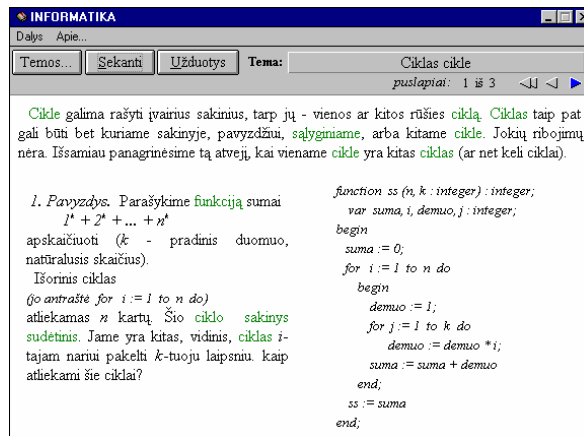
Programa skirta bendrojo lavinimo mokyklų 10–12 klasių mokiniams. Tai kompiuterizuotas V.Dagienės ir G.Grigo vadovėlio „Informatika“ variantas – elektroninė knyga su testu žinioms patikrinti.



48 pav. Programa „Informatika“. Temos.

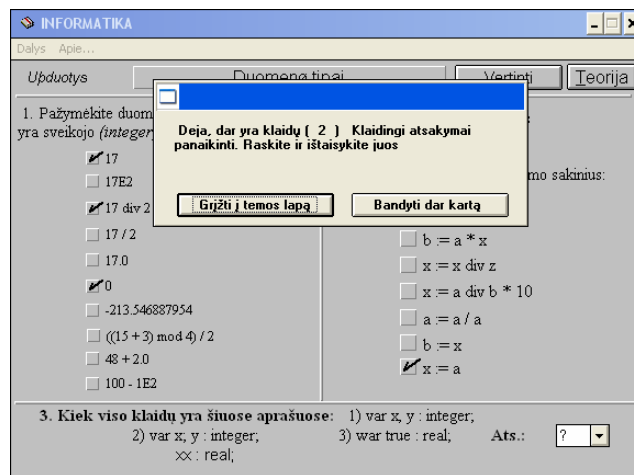
Programos teorinę dalį sudaro 22 temos:

1. Sveikieji ir realieji skaičiai.
2. Logika ir logikos algebra.
3. Konjunkcija.
4. Disjunkcija.
5. Loginiai reiškiniai.
6. Algoritmo sąvoka.
7. Kintamojo sąvoka.
8. Reiškiniai.
9. Reikšmių priskyrimas.
10. Duomenų tipai.
11. Procedūra.
12. Funkcija.
13. Loginiai duomenys.
14. Vienas iš dviejų veiksmų.
15. Vienas iš kelių veiksmų.
16. Sudėtinis sakiny.
17. Ciklas.
18. Žinomo kartojimų skaičiaus ciklas.
19. Ciklas cikle.
20. Programa.
21. Funkcijos ir procedūros programoje.
22. Masyvas.



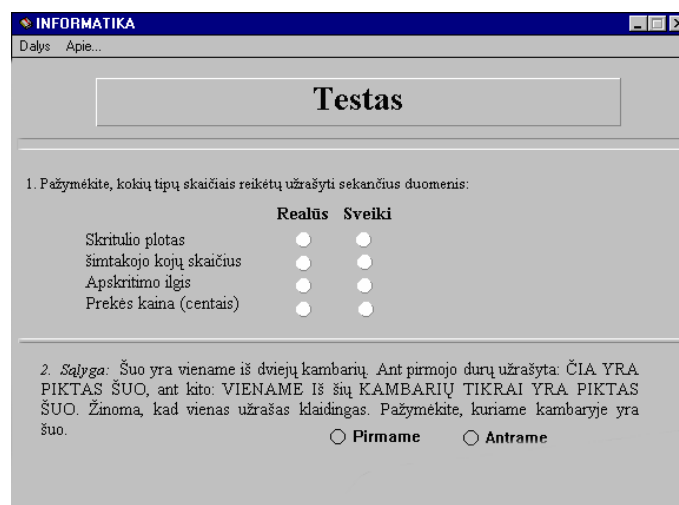
49 pav. Programa „Informatika“. Teorija.

Kiekvieną teorinę temos dalį atitinka praktinė. Tai keletas į dalis skaidomų klausimų paketas, skirtas savarankiškai patitikrinti, kaip įsisavinta pastaroji teorija. Vienintelė tema, neturinti uždavinių – „Programa“.



50 pav. Programa „Informatika“. Savarankiškos užduotys.

Kiekvienai teorinei temai yra sukurtos savikontrolės užduotys, kurias įvykdžius yra pateikiamas klaidingų atsakymų kiekis.



51 pav. Programa „Informatika“. Testas.

Testas. Jį sudaro 20 klausimų, apimančių visas programoje siūlomas temas. Radusi klaidas, programa atlieka šiuos veiksmus:

- ✓ Turinyje pažymės temas, atitinkančias klaidingai atsakytus klausimus. (Sudarys temų, kurias dar reikia skaityti, grupę).
- ✓ Praneš klaidų skaičių.
- ✓ Perkels į turinio lapą. Čia mygtuko „Skaityti pažymėtas temas“ pagalba galima pereiti prie pažymėtų temų nuoseklaus skaitymo.
- ✓ Tuo atveju, jei klaidų nėra, programa siūlo baigti darbą.

PROGRAMAVIMO PRADMENYS

Autoriai: A. Verseckas, A. Jarockis Kuršėnų L.Ivinskio gimnazija.

E-vadovėlį sudaro 11 skyrių:

1. Programa
2. Rašymo sakiniai
3. Programų rašymas
4. Skaičių sekos
5. Operacijos div ir mod
6. Skaičiaus skaitmenys
7. Sąlyginis sakiny
8. Loginiai duomenys
9. Vienas iš kelių veiksmų
10. Ciklas For
11. Ciklas cikle

Kiekviename skyriuje yra aiškinamoji tekstinė ir animacinė medžiaga.

Animacija paaiškina programos veikimą (programos veikimo modelis), aiškina programavimo kalbos Pascal konstrukcijas (programavimo kalbos konstrukcijų modeliai) ir duotą uždavinį (uždavinių aiškinimo modeliai).

2. Rašymo sakiniai.

Naudojami du rašymo sakiniai **writeln** ir **write**. **Writeln** atspausdina ir peršoka į kitą eilutę, o **write** spausdina vienoje eilutėje ir pereina į kitą tik tada, kai duomenys netelpa.

Užduotys

Nr. 1

Parašyta programa, kuri tvarkingai atspausdina skaičiaus **a n** kartotinių. Pvz. jeigu **a = 5, n = 11**, programa turi spausdinti:

```
Skaičiaus 5 kartotiniai:
10 15 20 25 30 35 40 45 50 55 60
```

Vietoje daugtaškių įrašykite reikiamas reikšmes.

```
programos antrašte akartotiniai yra programos vardas
```

```
program akartotiniai;
var a, ..., ..., ... : integer;
begin
  writeln('Įveskite a ir n');
  read(...);
```

52 pav. Programa „Programavimo pradmenys“. Teorija.

Taip pat yra pateikiami sukurtų programų komentarai. Atvedę pelės žymeklį ant bet kokios programos eilutės, virš programos esančioje juostelėje, pamatysite tos eilutės komentarą.

Pvz. 4.1. Programos veikimo modelis

Parašykite programą, kuri rastų duotas skaičių sekos sumą. Sekos pabaigos požymis yra nulis, pagal jį kompiuteris sužino, kad seka jau baigėsi.

Programos tekstas

```
program sekossuma;
var n, sum : integer;
begin
  read(n);
  sum := 0;
  while n <> 0 do
  begin
    sum := sum + n;
    read(n);
  end;
  writeln('Sum = ', sum);
end.
```

Vykdomas sakinyje

```
read(n);
```

Klaviatūra įvedama skaičių seka, kurią baigiame skaičiumi 0

Atmintinė

```
n = 8
sum =
```

Įvesta skaičių seka: 8 10 3 4 0

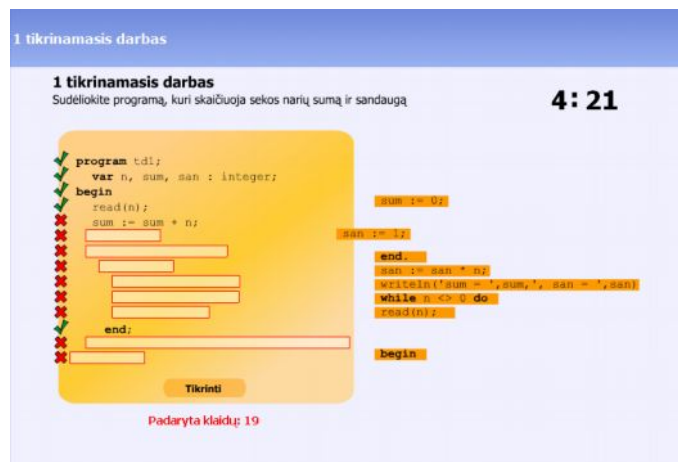
53 pav. Programa „Programavimo pradmenys“. Programos veikimo modelis.

Dar geriau programos veikimą paaiškina animuotas jos modelis. Paspaudę nuorodą, prasidedančią žodeliu „pvz.“, galėsite interaktyviai stebėti ir valdyti nagrinėjamą programą.



54 pav. Programa „Programavimo pradmenys“. Animuotas uždavinio aiškinimas.

Paspaukę kitokią nuorodą, tarkim, prasidedančią žodeliu „Nr.“, arba atskiram uždavinio atvežui, galėsite žiūrėti animuotą uždavinio aiškinimą ar animuotą sprendimo idėją.



55 pav. Programa „Programavimo pradmenys“. Tikrinamasis darbas.

Kiekviename skyrelyje yra vienas arba du tikrinamieji darbai. Paspaudus nuorodą gaunate užduotį sudėlioti programą ir apskaičiuoti rezultatus su duotais pradiniais duomenimis.

ELEKTRONINIS INFORMATIKOS VADOVĖLIS.

<http://www.daukantas.lt/edu/Info/Vadovelis/>

Kompiuterinis „Informatikos vadovėlis“ sukurtas Alytaus Vidzgirio vidurinėje mokykloje. Autorius – informatikos vyr. mokytojas V.Žemaitis. Vadovėlyje mokomoji medžiaga pateikta pagal Valentinos Dagienės vadovėlius INFORMATIKOS PRADMENYS.

Antrą dalį algoritmai sudaro šios temos:

1. Algoritmo sąvoka.
2. Algoritmo savybės.
3. Algoritmų vaizdavimo būdai.

4. Algoritmas ir programa.
 5. Kintamieji ir jų reikšmės.
 6. Reikšmių priskyrimas.
 7. Loginiai duomenys.
 8. Pasirinkimas: vienas iš dviejų veiksmų.
 9. Pasirinkimas: vienas iš kelių veiksmų.
 10. Ciklas.
 11. Algoritmų testavimas.
 12. Žinomo kartojimų skaičiaus ciklas.
 13. Ciklas cikle.
 14. Algoritmų skaitymas.
 15. Algoritmavimo stilius ir kultūra.
 16. Uždavinių sprendimo etapai.
 17. Programuotojas, kompiuteris, vartotojas.
- Kiekvieną skyrių sudaro teorija bei pratimai ir užduotys.

ALGORITMAS: sveikųjų skaičių dalyba
PRADINIAI DUOMENYS: du natūralieji skaičiai a ir b
Dahmenį prilyginame 0: $dalmuo := 0$
KOL: $a >= b$
Kiekvieną kartą iš skaičiaus a atimame skaičių b , t.y. $a := a - b$
Padidiname dahmenį vienetu: $dalmuo := dalmuo + 1$
Liekana bus skaičiaus a likutis: $liekana := a$
REZULTATAS: $dalmuo$ ir $liekana$

Urašysime tą patį algoritmą Paskalio kalba. Ciklo schema Paskalio kalboje pateikta dešinėje esančiame paveikslėlyje. Šis ciklas dažnai vadinamas nežinomo kartojimų skaičiaus ciklu (arba **while** ciklu: **while** - "kol"). Lietuviškai jį galima būtų vadinti **KOL** ciklu.

Ciklą pradeda antraštė: pirmas žodis **while**, toliau rašoma sąlyga (loginis reiškinys), o po jos - žodis **do** ("atlikti, daryti"), kuriuo užbaigiama antraštė. Toliau nurodomi ciklą sudarantys veiksmai - sakiny, kuris kartojamas tol, kol tenkinama antraštėje nurodyta sąlyga.

Ciklo antraštė valdo tik vieno (po jos einančio) saknio kartojimą. Jeigu reikia kartoti kelių saknius, tai jie jungiami į sudėtinį sakinį (apgaubiami žodžiais **begin** ir **end**, taip pat kaip ir sąlyguose sakiniuose).

2 pavyzdžio ciklas Paskalio kalba būtų užrašomas šitaip:

```

while a >= b do
begin
a := a - b;
dalmuo := dalmuo + 1
end

```

WHILE
KOL
DARYK

while **sąlyga** **do**

sakiny

Ciklo KOL schema

56 pav. Elektroninis informatikos vadovėlis. Teorija.

Kai kuriuose pratimų ir užduočių skyriuose galima rasti testus pagal išeitą temą.

10 skyriaus pratimai ir užduotys

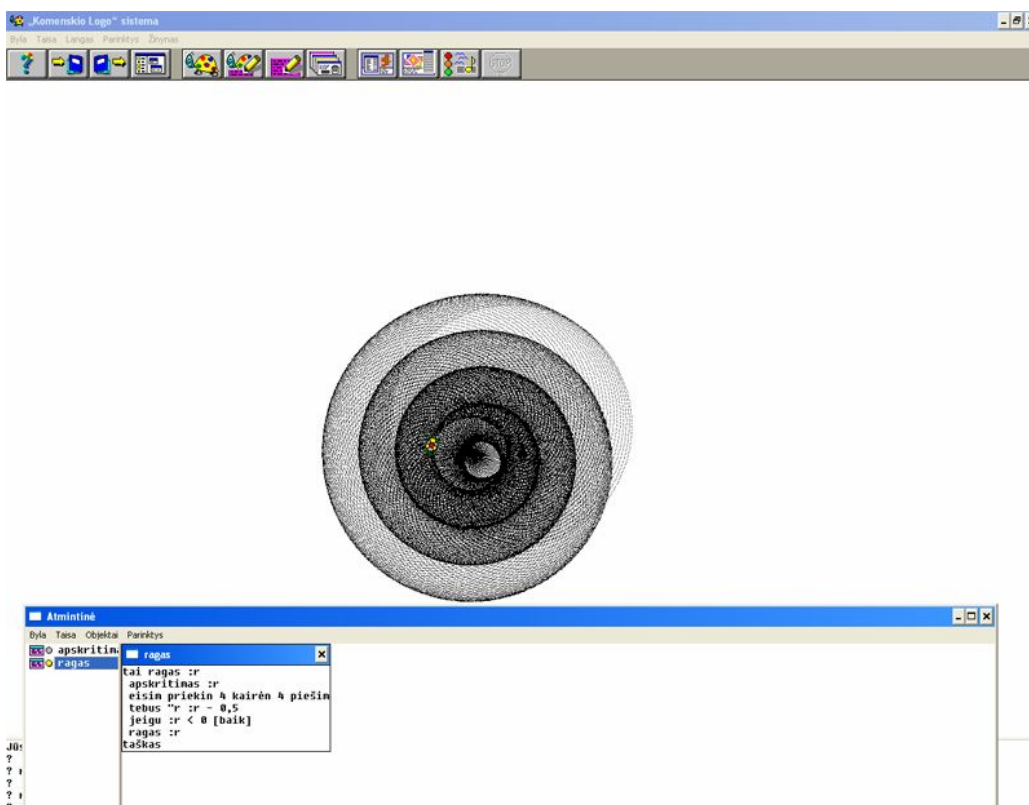
Pasirinkite	
01	
02	
03	
04	
05	5. Parašykite algoritmą natūraliojo skaičiaus skaitmenų sumai apskaičiuoti.
06	
07	
08	
09	
10	
Testas	

57 pav. Elektroninis informatikos vadovėlis. Užduotys.

Tai modifikuotos A.Versecko PaskMok kontrolinių testų užduotys, kurių skaičius sumažintas iki 8.

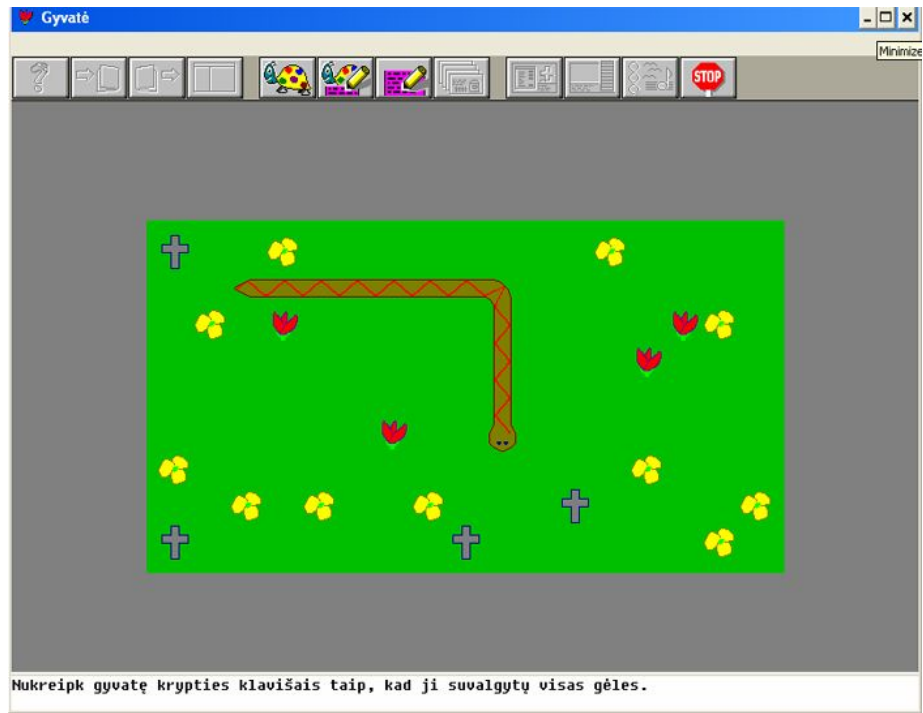
KOMENSKIO LOGO PROGRAMA

Komenskio Logo programa labiausiai tinka mokyti perprasti darbo kompiuteriu (ypač programavimo) principus. Šia programa galime atlikti įvairias užduotis: piešti paveikslus ir kurti melodijas, aprašyti geometrines figūras, išreikšti matematinius skaičiavimus, modeliuoti įvairius fizikos, chemijos, biologijos, kalbos procesus (paprastai sakoma: kurti projektus), mokytis panaudoti multimedijos principus ir galimybes. *Komenskio Logo* sistema visiškai suderinta su operacinės sistemos *Windows* darbu, todėl, jei yra poreikis, galima naudotis kitų programų rezultatais, o svarbiausia – lengvai įgyjami darbo *Windows* terpėje bei programavimo įgūdžiai. [7]



58 pav. Komenskio logo programa. Procedūros ir rezultato langai

Taigi *Komenskio Logo* programa labiausiai tinka modeliuoti ir projektuoti įvairias užduotis, pasitelkus šiuolaikinio programavimo idėjas. Šį darbą nesunkiai gali atlikti netgi jaunesniojo amžiaus vaikai. Programa nesudėtinga, greitai perprantami jos pagrindiniai veiksmai, vaizdžiai išreiškiami rezultatai, kita vertus, joje daug įvairių šiuolaikinių programavimo idėjų bei konstrukcijų, patogių priemonių kompiuterio programinei įrangai projektuoti bei kurti.



59 pav. Komenskio logo programa. Sukurti projektai.

Komenskio Logo – tai ne tik programavimo kalba skirta mokymuisi, bet drauge ir didaktinė sistema. Ji priklauso vertingiausių dvidešimtojo amžiaus pabaigos mokomųjų programų grupei – tokių, kurios leidžia tenkinti pagrindinius dabartinio žmogaus poreikius, būtent, nuolatinio mokymosi ir savojo kūrybiškumo ugdymo. Šios sistemos autoriai sukūrė tokią sistemą, kuri atitinka mokinių, mokytojų ir kompiuterinių mokymo priemonių kūrėjų poreikius.

Priedas Nr. 2. MKP vertinimo kriterijai

MKP VERTINIMO KRITERIJAI

MKP vertinimo kriterijai	MKP vertinimo kriterijų požymiai
Mokomoji medžiaga	<p>Informacija: mokomosios medžiagos gylis, pateikimo formų įvairovė.</p> <p>Mokomosios medžiagos profesionalumas: dalykinis tikslumas ir aiškumas; pristatomų sąvokų, teorijų, interpretacijų šiuolaikiškumas; profesionalumas bei originalumas; naudojamų šaltinių patikimumas.</p> <p>Mokomosios medžiagos pateikimas: struktūros aiškumas, nuoseklumas ir tvarkingumas; mokomosios medžiagos tinkamumas, pritaikomumas ir suprantamumas pasirinktoms auditorijoms; įvairių pateikimo formų derinimas ir pateikimo naujumas; taisyklingas stilius ir kalba.</p> <p>Socialiniai, kultūriniai ir doriniai principai: humaniškumas; demokratiškumas; nacionalumas; lyčių ir amžiaus grupių lygybė; mažumų, kitų valstybių, tautų, rasių, religijų nediskriminavimas ir etikos normų laikymasis.</p> <p>Autorių teisės: Lietuvoje galiojančio Autorių teisių ir gretutinių teisių įstatymo laikymasis (Žin., 2003, Nr.28-1125).</p>
Psichologiniai ir pedagoginiai aspektai	<p>Psichologinis tinkamumas: MKP sudėtingumo atitiktis pasirinktai auditorijai; mokomosios medžiagos, užduočių ir kitos informacijos orientavimas į mokinių patirtį, socialinį ir kultūrinį kontekstą; vaizdumas; psichologinis ergonomiškumas.</p> <p>Pedagoginis tinkamumas: sistemingumas, dermė su kitų dalykinių sričių, klasių, amžiaus grupių vadovėliais, MKP bei kitomis mokymosi priemonėmis ir naudojamais ugdymo metodais.</p> <p>Ugdymo metodų įvairovė ir dermė: dėstymas, imitavimas, modeliavimas, praktinės užduotys, darbas grupėse, projektų darbas ir kt.</p> <p>Didaktinių principų šiuolaikiškumas: konstruktyvistiniai metodai; kūrybiškumo, savarankiškumo ir pasirinkimo skatinimas; atviro teksto užduotys ir kt.</p> <p>Semantinis integralumas: mokomosios medžiagos, ugdymo</p>

	<p>metodų, formų, strategijų, didaktinių principų ir kt. dermė.</p>
Mokymosi valdymas ir interaktyvumas	<p>Interaktyvumo galimybių panaudojimas: grįžtamasis ryšys; MKP pranašumas lyginant su vadovėliais, pratybų sąsiuviniais, atlasais, uždavinynais ir kitomis tradicinėmis mokymo priemonėmis.</p> <p>Pritaikomumas savarankiškam darbui: galimybės automatizuoti ir/ arba pačiam kontroliuoti mokymosi procesą; mokymosi eigos diferencijavimas ir individualizavimas; pažangos ir pasiekimų įvertinimas bei įsivertinimas; informacija apie padarytas klaidas ir klaidų analizė; rekomendacijos.</p> <p>Papildomos mokymosi priemonės: pagalbos sistema; žinynai ir žodynai; paieška; vartotojo vadovas.</p> <p>Mokymosi organizavimo priemonės: kurso kalendorius; užrašų knygelė; studijų gidas ir kt.</p>
Vartotojo sąsaja	<p>Kokybė: fono ir teksto dermė; harmoningas išdėstymas ekrane; stilinga grafika, šriftai, spalvos, garsas, vaizdo ir kiti daugialypės terpės elementai.</p> <p>Patogumas: valdymo elementų paprastumas, patogumas, semantinis aiškumas; aiški navigacija; logiškas leidimas arba draudimas pasirinkti meniu punktus, valdymo elementus; vartotojo informavimas apie darbo eigą ir būseną.</p> <p>Individualizavimas: garso, grafikos ir vaizdo valdymas; galimybė keisti ir pritaikyti savo poreikiams visus vartotojo sąsajos elementus; pritaikomumas vartotojams, turintiems regėjimo, klausos ir judėjimo negalią*.</p> <p>Ergonomiškumas: vaizdo virpėjimas, raibuliavimas, aiškumas, ryškumas ir kontrastas; kitų daugialypės terpės elementų kokybė.</p>
Vartotojų administravimo galimybės	<p>Vartotojų administravimo priemonės: registravimo ir išregistravimo galimybės; informacijos apie vartotojus apskaitos galimybės.</p> <p>Mokymosi proceso administravimo ir individualizavimo galimybės: vartotojų profiliavimas; mokymosi eigos diferencijavimas; informacijos apie mokymąsi kaupimas ir apskaita (suvestinės); stebėjimas (mokymosi kontrolė).</p> <p>Vartotojų apsauga: autorizavimas; asmeninių duomenų ir informacijos saugumas; apsauga nuo išorinio neigiamo poveikio.</p>

	<p>Informacijos saugumas: turinio, vartotojo duomenų ir kitos informacijos dubliavimas.</p>
<p>Instrumentikos priemonės (rengimo galimybės)</p>	<p>Mokymosi objektų kūrimo galimybės. Mokymosi objektų kaupimo ir paieškos galimybės. Galimybės parengti (sudaryti) kursą. Standartai: mokymosi objektų, metaduomenų, kursų ir kitos informacijos duomenų formatų suderinamumas su tarptautiniais standartais (pvz., SCORM).</p>
<p>Bendravimo ir bendradarbiavimo galimybės bei priemonės</p>	<p>Sinchroninės priemonės: pokalbiai, vaizdo konferencijos. Asinchroninės priemonės: elektroninis paštas, diskusijos, pasikeitimas rinkmenomis, naujienų grupės ir kt.</p>
<p>Techninės savybės</p>	<p>Kokybė: veikimo stabilumas; darbo sparta; prieinamumas (23,5 h per parą) ir kt. Eksportas ir importas: suderinamumas su dažniausiai vartojamais rinkmenų tipais bei su tarptautiniais standartais (pvz., SCORM). Technologinis lankstumas ir suderinamumas: nesudėtingas programos įdiegimas ir automatinis suderinimas su senesne ir naujausia technine bei sisteminė įranga; palaikymas ir atnaujinimas; minimalus sistemos išteklių panaudojimas; optimizuota grafika ir kt.; pagrindinių Lietuvos ir tarptautinių informacinių technologijų standartų atitikimas **.</p>
<p>Dokumentacija ir papildomos priemonės</p>	<p>Bendra informacija: MKP koncepcija; paskirtis ir svarbiausios savybės; reikalavimai operacinei sistemai, programinei ir techninei įrangai; kitų galimų MKP komplekto dalių aprašas. Vartotojo vadovas: įdiegimo aprašymas; išsamus naudojimosi ir valdymo galimybių aprašymas; MKP mokomosios medžiagos apibūdinimas. Metodinė medžiaga: metodinės rekomendacijos ir papildomos</p>

	<p>metodinės priemonės mokiniui; metodinės rekomendacijos ir papildomos priemonės mokytojui.</p> <p>Dokumentacijos ir papildomų priemonių taisyklingas stilius ir kalba.</p>
<p>Ekonominis veiksmingumas</p>	<p>Kaina.</p> <p>Įdiegimo sąnaudos: pedagogų mokymui reikalingos lėšos; įrangai atnaujinti arba įsigyti reikalingos lėšos; kitos papildomos sąnaudos ir išteklių.</p> <p>MKP palaikymo sąnaudos: nuolatinis naudojimas ir palaikymas; MKP atnaujinimas; kt.</p>

Priedas Nr. 3. Algoritmų struktūrizavimo dokumentacija

Duoti a, b ir c. Apskaičiuokite $S = a + b * y$, kai $y = a * c + b$.

17 lentelė. Algoritmų analizė. Priskyrimo sakiny 1.

<pre> graph TD Pradžia([Pradžia]) --> Pabaiga([Pabaiga]) </pre>		
	Blokai	Komentarai (taisyklės)
		Duomenų įvedimas. Tam, kad galėtume atlikti veiksmus, turime įvesti pradinį duomenį
		1 priskyrimo sakiny. Svarbi priskyrimo sakinių tvarka. Tam, kad sėkmingai gautume rezultatą, pirmiausiai turime apskaičiuoti tarpinių kintamųjų reikšmes (konkrečiu atveju turime rasti kintamojo y reikšmę).
		2 priskyrimo sakiny. Apskaičiuojamas programos rezultatas.
	Rezultatu išvedimas. Atlikus veiksmus būtina pateikti rezultatus (konkrečiu atveju yra pateikiama kintamojo S reikšmė).	

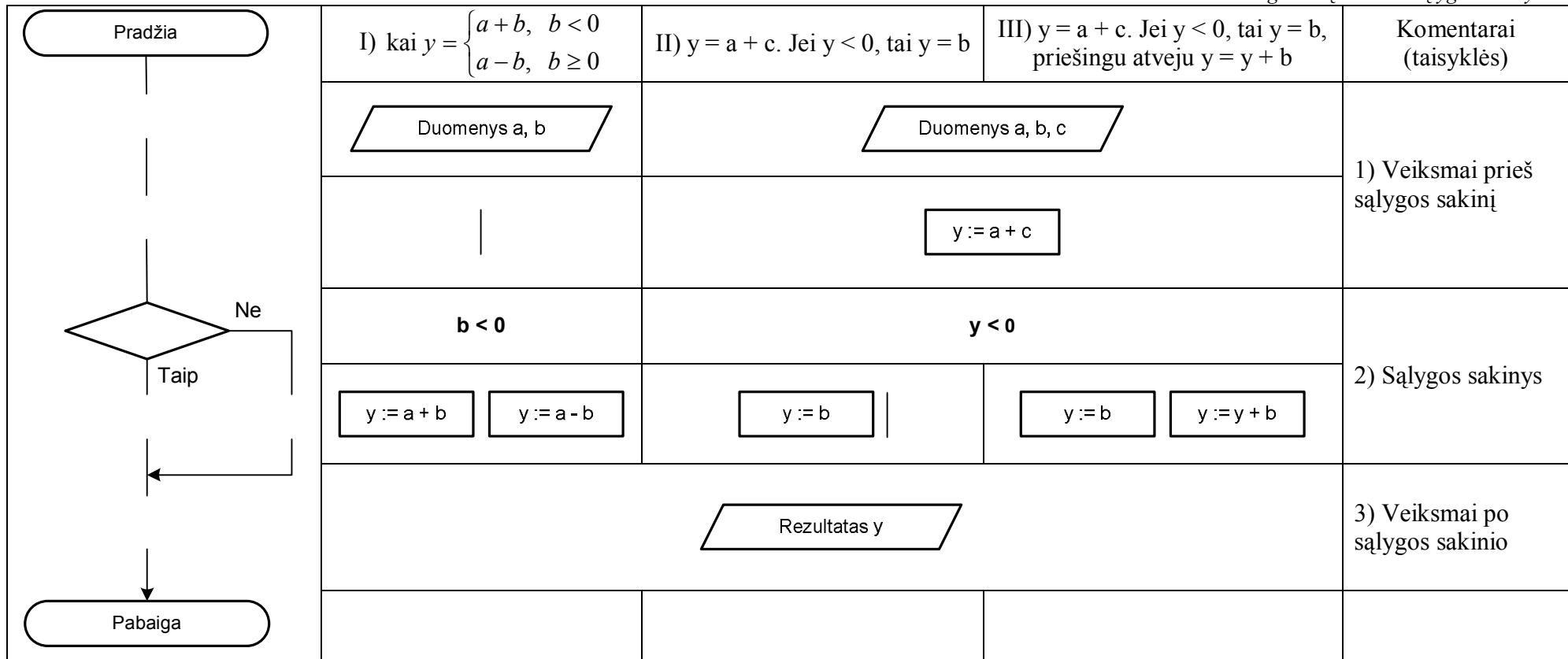
Apskaičiuoti dviejų skaičių sumą. Spausdinti duotus dydžius ir sumą.

18 lentelė. Algoritmų analizė. Priskyrimo sakiny 2

	Taisyklės	1. Kai a ir b duoti.	2. Kai $b = 2$, o a duotas.	3. Kai b duotas.	4. Kai $a = 2$.
	1				
	2				
	3				
	4				

Duoti a, b ir c. Apskaičiuokite ir spausdinkite funkcijos reikšmę.

19 lentelė. Algoritmų analizė. Sąlygos sakiny.



- 1) Veiksmai prieš sąlygos sakinį:** Įvedami pradiniai duomenys (atliekami papildomi veiksmai pvz. randamos tarpinių kintamųjų reikšmės $y := a + c$).
- 2) Sąlygos sakiny:** Nurodoma teisinga sąlyga. Gali būti paprastas sąlygos algoritmas (kai atliekami nepriklausomai nuo to ar sąlyga teisinga ar klaidinga). Sutrumpintas sąlygos algoritmas (sakiny), kuriame yra veiksmai atliekami tik vienu atveju kai sąlyga teisinga arba klaidinga.
- 3) Veiksmai po sąlygos sakinio:** Dažniausiai išvedamas rezultatas. Gali būti atliekami papildomi veiksmai.

20 lentelė. Algoritmų analizė. Sumos, sandaugos ir kiekio skaičiavimas.

	Apskaičiuoti funkcijos $f(x)$ reikšmes, kai x kinta nuo x_p iki x_g žingsniu h_x ir rasti jų:			Apskaičiuoti duoto masyvo $X(n)$ elementų:			Komentarai (taisyklės)
	1. Sumą.	2. Sandaugą	3. Kiekį	4. Sumą.	5. Sandaugą	6. Kiekį	
	Duomenys x_p, x_g, h_x			Duomenys $n, X(n)$			1) Veiksmai prieš ciklo sakinį
	$x := x_p$			$x := 1$			
	$S := 0$	$P := 1$	$k := 0$	$S := 0$	$P := 1$	$k := 0$	
	$x \leq x_g$			$i \leq n$			2) Ciklo sakiny
	$S := S + f(x)$	$P := P * f(x)$	$k := k + 1$	$S := S + x_i$	$P := P * x_i$	$k := k + 1$	
	$x := x + h_x$			$i := i + 1$			
	Rezultatas S	Rezultatas P	Rezultatas k	Rezultatas S	Rezultatas P	Rezultatas k	3) Veiksmai po ciklo sakinio
Pabaiga							

1 Veiksmai prieš ciklo sakinį:

- ✓ Pradinių duomenų įvedimas.
- ✓ Nurodoma pradinė ciklo kintamojo reikšmė.
- ✓ Jei ieškome sumos ar kiekio, tai pradinė reikšmė 0, jei sandaugos – pradinė reikšmė lygi 1.

2 Ciklo sakiny:

- ✓ Teisingai nurodoma ciklo pabaigos sąlyga (priešingu atveju galimas amžinas ciklas, arba ciklas veiks blogai). Jei algoritme atlikinėjami veiksmai su funkcija, tai dažniausiai ciklas vykdomas tol, kol pasiekiamas galutinė intervalo reikšmė. Jei veiksmai atliekami su masyvo elementais, tai ciklas vykdomas tiek, kiek masyve elementų.
- ✓ Atliekami sumos, sandaugos ar kiekio skaičiavimai. Dažniausiai svarbu, kad šie skaičiavimai būtų atliekami prieš ciklo kintamojo reikšmės pakeitimą.
- ✓ Keičiama ciklo kintamojo reikšmė. Jei reikšmė nebus keičiama arba keičiama neteisingai, galima situacija „Amžinas ciklas“.

3 Veiksmai po ciklo sakinio:

- ✓ Išvedamas programos rezultatas.

21 lentelė. Algoritmų analizė. Sumos, sandaugos ir kiekio skaičiavimas (teigiamo arba neigiamo).

	Apskaičiuoti funkcijos $f(x)$ teigiamų (neigiamų) reikšmių $***$, kai x kinta nuo x_p iki x_g žingsniu h_x .			Apskaičiuoti duoto masyvo $X(n)$ teigiamų (neigiamų) elementų:			Komentarai (taisyklės)
	1. Sumą.	2. Sandaugą	3. Kiekį	4. Sumą.	5. Sandaugą	6. Kiekį	
	Duomenys x_p, x_g, h_x			Duomenys $n, X(n)$			1) Veiksmai prieš ciklo sakinį
	$x := x_p$			$i := 1$			
	$S := 0$	$P := 1$	$k := 0$	$S := 0$	$P := 1$	$k := 0$	
	$x \leq x_g$			$i \leq n$			2) Ciklo sakiny
	$f(x) > 0 \quad f(x) < 0$			$x_i > 0 \quad x_i < 0$			
	$S := S + f(x)$	$P := P * f(x)$	$k := k + 1$	$S := S + x_i$	$P := P * x_i$	$k := k + 1$	
	$x := x + h_x$			$i := i + 1$			
	Rezultatas S	Rezultatas P	Rezultatas k	Rezultatas S	Rezultatas P	Rezultatas k	3) Veiksmai po ciklo sakinio

1) **Veiksmai prieš ciklo sakinį:**

- ✓ Pradinių duomenų įvedimas.
- ✓ Nurodoma pradinė ciklo kintamojo reikšmė.
- ✓ Jei ieškome sumos ar kiekio, tai pradinė reikšmė 0, jei sandaugos – pradinė reikšmė lygi 1.

2) **Ciklo sakiny:**

- ✓ Teisingai nurodoma ciklo pabaigos sąlyga.
- ✓ Nustatomas funkcijos ar masyvo elemento ženklas. Ir jei ženklas atitinka užduotyje keliamas sąlygas, atliekami sumos, sandaugos ar kiekio skaičiavimai. Jei neatitinka – atliekami sekantys programos sakiniai.
- ✓ Keičiama ciklo kintamojo reikšmė.

3) **Veiksmai po ciklo sakinio:**

- ✓ Išvedamas programos rezultatas.

22 lentelė. Algoritmų analizė. Minimumo, maksimumo radimas.

	Rasti didžiausią (mažiausią) funkcijos $f(x)$ reikšmę, kai x kinta nuo x_p iki x_g žingsniu h_x .		Duotas masyvas $X(n)$. Rasti didžiausią (mažiausią) jo elementą.		Komentarai (taisyklės)
	1. Max	2. Min	3. Max	4. Min	
	Duomenys x_p, x_g, h_x		Duomenys $n, X(n)$		1) Veiksmai prieš ciklo sakinį
	$\max := f(x_p)$	$\min := f(x_p)$	$\max = x_1$	$\min = x_1$	
	$x := x + x_p$		$i := 2$		
	$x \leq x_g$		$i \leq n$		2) Ciklo sakiny
	$\max < f(x)$	$\min > f(x)$	$\max < x_i$	$\min > x_i$	
	$\max := f(x)$	$\min := f(x)$	$\max := x_i$	$\min := x_i$	
	$x := x + x_p$		$i := i + 1$		
	Rezultatas max	Rezultatas min	Rezultatas max	Rezultatas min	
				3) Veiksmai po ciklo sakinio	

1) Veiksmai prieš ciklo sakinį:

- ✓ Pradinių duomenų įvedimas.
- ✓ Prielaida, kad pirmoji funkcijos reikšmė (pirmas masyvo elementas) yra didžiausia (mažiausia).
- ✓ Nurodoma pradinė ciklo kintamojo reikšmė.
- ✓ Jei ieškome sumos ar kiekio, tai pradinė reikšmė 0, jei sandaugos – pradinė reikšmė lygi 1.

2) Ciklo sakiny:

- ✓ Teisingai nurodoma ciklo pabaigos sąlyga.
- ✓ Lyginama tariamoji mažiausioji (didžiausioji) funkcijos reikšmė su sekančia funkcijos reikšme. Jei reikia atliekamas atitinkamas pakeitimas. Jei neatitinka – atliekami sekantys programos sakiniai.
- ✓ Keičiama ciklo kintamojo reikšmė.

3) Veiksmai po ciklo sakinio:

- ✓ Išvedamas programos rezultatas.

Priedas Nr. 4. Anketa mokiniams

Anketa mokiniams

1. Ar turite namuose kompiuterį?

- Taip Ne

2. Ar galite naudotis internetu?

- Taip Ne

3. Ar savarankiškai naudojate mokomasias kompiuterines priemones namuose?

- Taip Ne

4. Kokios mokymosi formos jums atrodo priimtinausios, kad pamoka būtų įdomi? Įvertinkite balais nuo 1 iki 4 (4 – geriausias įvertinimas, 1 – prasčiausias įvertinimas).

Vadovaujant mokytojui

Vadovėlio pagalba

Tik mokomosios programos

Reikia ir mokomųjų programų ir mokytojo aiškinimų

5. Kaip vertinate MKP „Algoritmai“ teorinės medžiagos pateikimą?

- Puikiai Gerai Patenkinamai Blogai

6. Kaip vertinate MKP „Algoritmai“ pateiktus pavyzdžius?

- Puikiai Gerai Patenkinamai Blogai

7. Kaip vertinate MKP „Algoritmai“ pateiktus testus?

- Puikiai Gerai Patenkinamai Blogai

8. Kaip vertinate mokomosios priemonės valdymą?

- Puikiai Gerai Patenkinamai Blogai

9. Ar naudojote šią priemonę namuose?

- Taip Ne

10. Ar ši priemonė padėjo įsisavinti mokomąją medžiagą?

- Taip Ne

11. Parašykite savo pastebėjimus ir pastabas.