

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PRAKTINĖS INFORMATIKOS KATEDRA

Irmantas Kažukauskas

**PROGRAMINIO PAKETO *AUTOCAD* BRĖŽIMO
KOMANDŲ MOKYMO IR KONTROLĖS
PROGRAMINIŲ PRIEMONIŲ SUDARYMAS IR
TYRIMAS**

Magistro darbas

Darbo vadovas
doc. A. Lenkevičius

KAUNAS, 2004

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
PRAKTINĖS INFORMATIKOS KATEDRA

TVIRTINU
Katedros vedėjas
doc. dr. D. Rubliauskas
2004-05-25

**PROGRAMINIO PAKETO *AUTOCAD* BRĖŽIMO
KOMANDŲ MOKYMO IR KONTROLĖS
PROGRAMINIŲ PRIEMONIŲ SUDARYMAS IR
TYRIMAS**

Informatikos mokslo magistro baigiamasis darbas

Kalbos konsultantė
Lietuvių kalbos katedros lektorė
dr. J. Mikelionienė
2004 05-25

Vadovas
doc. A. Lenkevičius
2004 05-25

Recenzentas
doc. V. Jusas
2004 05-25

Atliko
IFM–8/3 gr. stud.
I. Kažukauskas
2004-05-25

KAUNAS, 2004

TURINYS

1.	ĮVADAS	3
2.	ANALITINĖ DALIS	6
2.1	MOKOMOSIOS PROGRAMOS APIBRĖŽIMAS IR STRUKTŪRA	6
2.2	BENDRIEJI MOKOMOSIOS SISTEMOS KŪRIMO PRINCIPAI	9
2.3	MOKOMŪJŲ SISTEMŲ PARUOŠIMO REKOMENDACIJOS	10
2.4	AUTOCAD 2002 – MOKYMO SISTEMOS PAGRINDAS	11
2.5	PROGRAMAVIMO KALBOS VISUALLISP PANAUDOJIMAS MOKYMO SISTEMOJE	12
2.5.1	PAGRINDINĖS VISUALLISP CHARAKTERISTIKOS	12
2.5.2	VISUALLISP DARBO PRINCIPAS	14
2.6	DXF FORMATO PANAUDOJIMAS MOKOMOJOJE SISTEMOJE	15
2.6.1	PAGRINDINĖS DXF FORMATO CHARAKTERISTIKOS	15
2.6.2	DXF NAUDOJIMO PRINCIPAS	16
3.	PROGRAMINĖS ĮRANGOS PROJEKTAS	17
3.1	AUTOCAD 2002 MOKOMOSIOS SISTEMOS SCENARIJUS	17
3.2	FUNKCINIAI REIKALAVIMAI	17
3.3	UŽDAVINIO SPRENDIMAS	18
3.4	PROGRAMINIŲ MODULIŲ SPECIFIKACIJOS	19
3.4.1	MODULIO „MENIU“ APRAŠYMAS	20
3.4.2	MODULIO „DARBINIS BRĖŽINYS“ APRAŠYMAS	20
3.4.3	MODULIO „PATIKRINIMAS“ APRAŠYMAS	21
3.4.4	MODULIO „INFORMACIJA“ APRAŠYMAS	21
3.4.5	MODULIO „PAKARTOJIMAS“ APRAŠYMAS	22
3.4.6	MODULIO „TIKRINIMAS“ APRAŠYMAS	22
3.5	MOKOMOSIOS SISTEMOS VALDYMAS	22
3.5.1	MENIU STRUKTŪRA BEI VALDYMAS	22
3.5.2	ĮRANKIŲ JUOSTOS VALDYMAS	26
3.6	MENIU PUNKTŲ APRAŠYMAS	26
3.7	NEFUNKCINIAI REIKALAVIMAI	29
3.7.1	PROGRAMINĖS IR APARATŪRINĖS ĮRANGOS REIKALAVIMAI	29
3.7.2	PROJEKTO PATIKIMUMAS	30
3.7.3	PROJEKTO GRAFIKŲ IR IŠLAIDŲ PLANAS	31
3.8	VARTOTOJO REIKALAVIMAI	32
3.9	RIZIKOS ĮVERTINIMO IR MAŽINIMO PLANAS	33
3.10	TESTAVIMAS	35
3.11	SUKURTOS MOKOMOSIOS SISTEMOS PARAMETRAI	35
4.	VARTOTOJO DOKUMENTACIJA	37
4.1	PASKIRTIS	37
4.2	SISTEMOS VADOVAS	37
4.3	SISTEMOS INSTALIAVIMO DOKUMENTAS	40
5.	PRODUKTO KOKYBĖS ĮVERTINIMAS	44
6.	IŠVADOS	47
7.	LITERATŪRA	48
8.	TERMINŲ IR SANTRUMPŲ ŽODYNAS	50
9.	PRIEDAI	51

SANTRAUKA

Sukurta mokomoji sistema skirta padėti studentams išugdyti naudojimą AutoCAD 2002 grafine sistema, bendruosius praktinius įgūdžius.

Mokomoji priemonė skirta mokytis bei mokyti naudoti kompiuteriu. Žmogus gali ja naudotis mokslo įstaigoje, darbe, namuose, visur, kur yra įdiegta AutoCAD 2002 mokomoji sistema. Šioje sistemoje yra pateikti praktiniai darbai apie grafinių primityvų braižymo komandas, brūkšniavimo ir matmenų komandas, grafinių objektų redagavimo priemones. Mokomoji sistema suskirstyta dalimis pagal temas.

Naudojama: personalinis kompiuteris, Windows 95/98/Me/2000/NT/XP operacinė sistema, programinis paketas AutoCAD 2002. Mokomoji sistema sukurta naudojantis VisualLISP programavimo kalba.

SUMMARY

The created learning program is given for students to develop their general practical skills for the use of graphical system AutoCAD 2002.

The learning program is given for learning and teaching using computer. The man can use it anywhere, where is the possibility to work with computer and learn the learning program. There are practical exercises for drawing graphical primitives; hatch and dimension commands; tools for editing graphical objects in the AutoCAD 2002 learning system. The learning system is divided into different parts by AutoCAD 2002 themes.

The requirements: personal computer, Windows 95/98/Me/NT/2000/XP operating system and AutoCAD 2002. The learning system is created using VisualLISP programming language.

1. ĮVADAS

Šiuolaikinis inžinerinės grafikos lygis pasiektas išplėtojus šimtmečiais kauptas mokslo ir technikos idėjas. Buvo pasitelkti matematikos ir braižomosios geometrijos analatiniai metodai ir topologijos žinios.

Grafinių darbų automatizavimo idėja kilo jau senokai – tada, kai atsirado pirmosios elektroninės skaičiavimo mašinos, tačiau praktiškai ją realizuoti pavyko tik sukūrus kokybiškus, grafiniam vaizdui rodyti tinkamus monitorius bei specialias funkcinio programavimo kalbas. Tuojuo atsirado braižymo darbus palengvinančios programos, skirtos ne tiek išprususiam matematikui, kiek eiliniam vartotojui, o inžinerinė grafika kompiuteriu palaipsniui tapo prieinama kiekvienam braižytojui – specialistui, dėstytojui ir net studentui.

Kompiuterinė grafika yra neatsiejama šio modernaus pasaulio kūrimo dalis. Vienas iš profesionaliausių ir patogiausių kompiuterinės grafikos kūrimo įrankių yra programinis paketas AutoCAD 2002. Nors AutoCAD sistema yra labai universali, vis dėlto jos sumanytojai numatė galimybę pritaikyti sistemą konkreitiems vartotojo poreikiams.

Pavyzdžiui, galima pertvarkyti komandų meniu struktūrą, sukurti paketų bylas, kad būtų galima automatiškai vykdyti kelias komandas, sukurti savo šriftus, linijų tipus, brūkšniavimą, šablonus, skaidres ir dar daug ką. Profesionalus vartotojas gali perprogramuoti sistemą, modifikuodamas esamas ar įdiegdamas visiškai naujas funkcijas AutoLISP arba C kalbomis.

Tad jei sistema, kurią dabar naudojate, kuo nors Jums neįtinka, visada turite galimybę pritaikyti ją savo konkreitiems poreikiams. AutoCAD – kiekvienam vartotojui atvira sistema.

Vadinasi, labiau patyręs specialistas, laikydamasis tam tikrų taisyklių, turi galimybę pertvarkyti darbo langų arba meniu struktūrą, pašalindamas nevertojamas komandas, susikurdamas ir įdiegdamas savąsias, priskirdamas senosioms kitokią prasmę. Yra trys keliai:

- ✓ perrašyti arba pakeisti meniu bylas naujomis;
- ✓ sukurti makrokomandų rinkinius – scenarijus, vadinamuosius „*script-files*“;
- ✓ naudojantis programavimo metodais parašyti programas, automatiškai vykdančias visas vartotojui reikalingas funkcijas.

Apibendrinant galima teigti, kad praktiškai visa tai, ką ruošiamės braižyti ar rašyti kompiuterio darbo lauke, galime pavesti daryti susikurtai programai. Jeigu pavyktų

pasirinkti optimaliausią braižymo, redagavimo ir AutoCAD programavimo priemonių santykį.

Modifikuoti AutoCAD'ą galima įvairiai, žiūrint ko vartotojui reikia ir ką jis sugeba padaryti, naudodamasis šiomis priemonėmis:

- ✓ funkcinė programavimo kalba AutoLISP;
- ✓ AutoLISP'o versija – VisualLISP;
- ✓ operatorine kalba C;
- ✓ simboline kalba DIESEL;
- ✓ dialogine DCL;
- ✓ periferinių įrenginių valdymo kalbomis HPGL, DMPL, DXY ir kitomis.

Gana nesudėtinga ir pakankamai lanksti kalba AutoLISP ir jos interpretatorius yra pati AutoCAD darbo aplinka. Šios kalbos ištakos siekia 1961 m., kai JAV buvo sukurta funkcinė dirbtinio intelekto valdymo kalba LISP, galinti operuoti žemiau išvardytomis informacijos struktūromis:

- ✓ lentelėmis;
- ✓ sąrašais;
- ✓ objektų aprašymais;
- ✓ bylomis.

Vėliau kiek pakeistas kalbos versijos CommonLISP dialektas buvo specializuotas grafiinei aplinkai AutoCAD ir pavadintas XLISP, neilgai trukus – AutoLISP. Kadangi tai – funkcinė kalba, jos leksikos pamatą sudaro funkcijų ir jų argumentų aibė, surašyta pagal tam tikrus vartosenos ypatumus. Pavyzdžiui, turime vartotojo funkciją. Vartotojo apibrėžtoje funkcijoje visi skaičiavimai ir valdymo instrukcijos atliekamos naudojant vidines AutoLISP kalbos elementarias funkcijas, sujungiant mažesnes ir perimant jų reikšmes kaip argumentus, prijungiant kitas standartines ar specializuotas funkcijas. LISP programa, sudaryta iš vienos ar daugiau vartotojo funkcijų, savo ruožtu gali būti panaudota kaip funkcija kitoms programoms, kitaip tariant, bet kuri funkcija su savo argumentais gali būti įdėta į kitą, kaip pastarosios argumentas. Funkcijų interpretavimas prasideda nuo paties žemiausio lygio, nuo giliausiai įdėtų funkcijų argumentų apdorojimo.

Kadangi šis programinis paketas AutoCAD yra gana sudėtingas, bet kartu ir patrauklus savo galimybėmis, todėl neatsitiktinai jau pirmame Kauno technologijos universiteto (KTU) kai kurių fakultetų kurse laboratoriniai darbai yra atliekami naudojant šį paketą. Per paskaitas AutoCAD 2002 nenagrinėjamas, o studentai turi jį įsisavinti per trumpą

laiką, tai studentui tenka daugiau dirbti savarankiškai. Todėl kvalifikuota kompiuterizuota mokomoji sistema yra labai naudinga.

Ši sukurtoji sistema yra būtent tokio tipo. Įdiegus sukurta mokymo sistemą bus galima įtvirtinti darbo su AutoCAD 2002 praktinius įgūdžius.

- ✓ braižant grafinius primityvus;
- ✓ brūkšniuojant ir nustatant matmenis;
- ✓ redaguojant grafinius primityvus.

Ši mokomoji sistema parengta A. Lenkevičiaus ir J. Maticko [2]. Užduotys pateikiamos popierinėje formoje. Tai nėra labai patogu, nes prieš atliekant užduotį, reikia nubraižyti pradinius duomenis, o tai kai kuriais atvejais užima netgi daugiau laiko, nei pačios užduoties atlikimas.

Vadovėlį papildžius diskeliu ar kompaktiniu disku su grafine bei mokomąja medžiaga, studentui nebereikėtų atlikinėti parengiamųjų darbų ir jis galėtų skirti daugiau dėmesio ir laiko užduoties atlikimui.

Panašią automatizuotą mokymosi metodiką naudoja ir kitos aukštosios mokyklos, pvz.: Klaipėdos universitetas (KU) [4], Maskvos Baumano aukštoji mokykla [8].

2. ANALITINĖ DALIS

2.1 MOKOMOSIOS PROGRAMOS APIBRĖŽIMAS IR STRUKTŪRA

Mokomųjų programų kūrimo ir panaudojimo mokymo procesų pradžia siejama su 1950 metais. Programuojamo mokymo idėją pasiūlė Harvardo universiteto profesorius B. F. Skinner. Dėl netobulos kompiuterinės technikos ir padarytų klaidų, šis mokymo metodas gyvavo nepilną dešimtmetį ir apie 1960 metus jį pakeitė – CAL (*Computer-aided learning*) arba amerikietišškai CAI (*Computer-aided instructions*). CAL buvo panaudota visa teigiama ir neigiama programuojamo mokymo metodo patirtis. Pirmą mokomoji programa pasaulį išvydo 1958 metais. Ji buvo skirta dvejetaini aritmetikai mokytis.

Kadangi žmogus sunkiai suvokia išsistintą informacijos srautą, todėl tikslinga jį struktūrizuoti, t. y. suskaidyti į dalis, telpančias kompiuterio ekrane. Šios informacijos dalys vadinamos informacijos mokomaisiais kvantais arba mokomaisiais kadrais (freimais). Mokomieji kvantai gali būti:

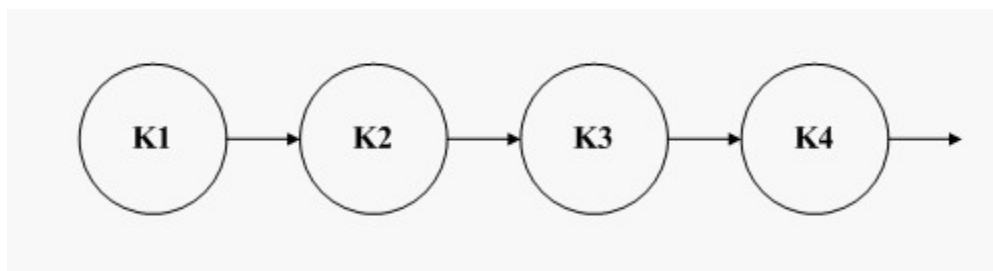
- ✓ statiniai – kai ekrane nėra nei vieno savo padėtį galinčio pakeisti objekto;
- ✓ dinaminiai – kai ekrane yra objektai, kurie gali pakeisti (ar keičia) savo padėtį;
- ✓ mišrūs – kai ekrane yra judantys ir nejudantys objektai.

Mokomoji programa – tai logiškai susietų mokomųjų kvantų visuma, skirta reikalaujamų mokinio žinių, mokėjimo įgūdžių lygiui formuoti.

Pagal tai, kokia tvarka mokiniui ar studentui yra pateikiami mokomieji kvantai, mokomosios programos yra suskirstomos į:

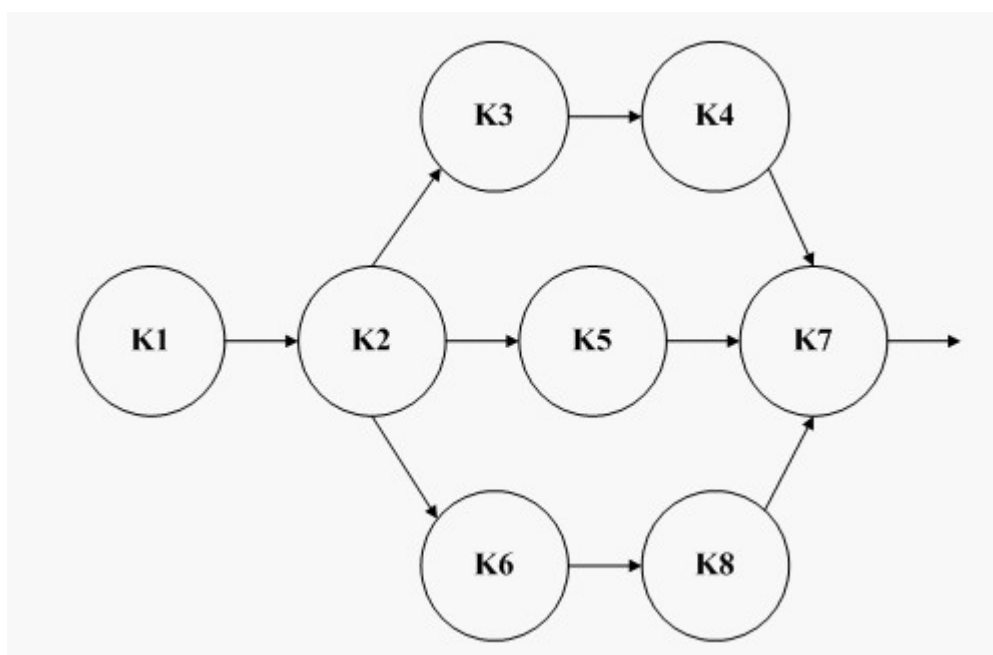
- ✓ tiesines;
- ✓ šakotas;
- ✓ daugialypes.

Tiesinėse mokomosiose programose mokomoji informacija pateikiama griežtai nustatyta tvarka, vienoda visiems mokiniams ar studentams.



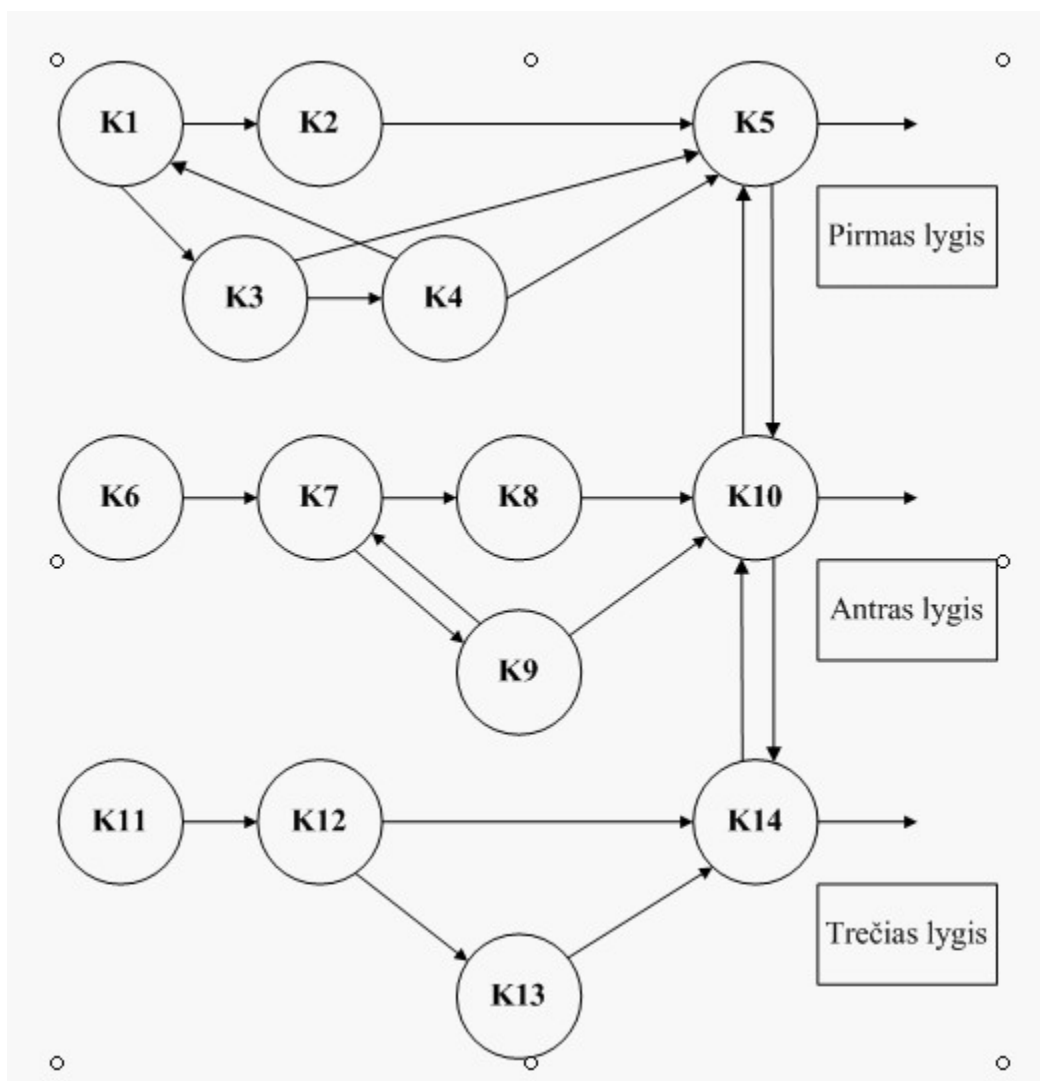
1 pav. Tiesinės mokomosios programos

Šakotose programose informacija gali būti pateikiama skirtinga tvarka. Ši tvarka priklauso nuo mokinio pasiruošimo lygio, kuris nustatomas klausimų-atsakymų metu (jei į pateiktą klausimą buvo atsakyta teisingai, tuomet kai kurie mokomieji kvantai gali būti praleisti, jeigu neteisingai – gali būti pateikti papildomi).



2 pav. Šakotosios mokomosios programos

Daugialypės mokomosiose programose ta pati mokomoji informacija pateikiama skirtingais lygiais. Kiekvienas informacijos lygis atitinka skirtingą mokinio paruošimo lygį. Mokomoji informacija skirtinguose lygiuose gali būti pateikiama tiek tiesine, tiek išsišakojusia tvarka.



3 pav. Daugialypės mokomosios programos

Pagal atliekamą funkciją mokomosios programos yra skirstomos į:

- ✓ treniravimo ir pratybų;
- ✓ modeliavimo;
- ✓ modeliavimo ir konstravimo.

Pasirenkant mokomąją programą, būtina atsižvelgti į tai, kokį žinių įsisavinimo lygį reikia pasiekti:

- ✓ Atsiminimo – treniravimo ir pratybų kontrolės programa;
- ✓ Supratimo – treniravimo ir pratybų kontrolės programos;
- ✓ Mokėjimo – modeliavimo;
- ✓ Pranešimo – konstravimo ir modeliavimo programos.

2.2 BENDRIEJI MOKOMOSIOS SISTEMOS KŪRIMO PRINCIPAI

Galima išskirti tokius mokomosios sistemos kūrimo etapus:

- ✓ **Scenarijaus aprašymas** (scenarijus – tai išsamus būsimosios sistemos, programos aprašymas);
- ✓ **Scenarijaus programavimas;**
- ✓ **Testavimas;**
- ✓ **Tiražavimas.**

Rašant mokomosios sistemos programos scenarijų ir jį programuojant, būtina atsižvelgti į daugelį psichologinių-pedagoginių veiksnių. Būtiniausi yra tokie:

1. Naujos informacijos įsiminimo užtikrinimas.

Jeigu mokomoji informacija visiškai nauja, tai įsimenant ją naudingas bet koks papildomas pastūmėjimas, pagalbinių informacija ir periodiškai naujos informacijos kartojimas.

Mokomosios sistemos, programos ir mokinio dialogas turi vykti korektiškai, vengiant užgaulių pranešimų. Atskirais atvejais, ypač įtvirtinant naują medžiagą, tikslinga naudoti padrąsinimo pobūdžio pranešimus.

2. Geras programos reakcijos laikas.

Optimalus mokomosios sistemos, programos reakcijos laikas į mokinio veiksmus, kai kurių autorių teigimu, neturi viršyti 3 sekundžių. Jei to pasiekti neįmanoma, kiekvienos reakcijos (viršijančios numatytą laiką) metu būtinas papildomas nuraminantis pranešimas.

Kiekvienoje mokomojoje sistemoje, programoje turi būti patenkinti ergonominiai faktoriai ir programos dizaino standartai. Kadangi skaityti ekrane esančią informaciją sunkiau negu iš knygos, reikia vengti ekrano perpildymo detalėmis. Pateikiama mokomoji informacija turi būti glausta ir gerai iliustruota, o raidės gerai įskaitomos. Svarbi informacija išdėstoma centrinėje ekrano dalyje ir, jeigu būtina, išskiriama kita spalva, įrėminama ar panašiai. Pranešimai, generuojami sistemos programos darbo metu, turi būti trumpi ir išsamūs. Tą pačią situaciją apibūdinantys pranešimai neturėtų kartotis. Jeigu mokomojoje programoje naudojamas meniu, tai:

- ✓ jis turi būti lengvai pasiekiamas bet kuriuo programos darbo metu;
- ✓ meniu punktų neturi būti daugiau kaip 9. Jei to pasiekti negalima, naudotinas hierarchinis (sudėtinis) meniu;

- ✓ meniu punktai turi būti išdėstyti taip, kad punktas, kurio pasirinkimas labiausiai tikėtinas, būtų pirmas, o mažiausiai tikėtinas – paskutinis.

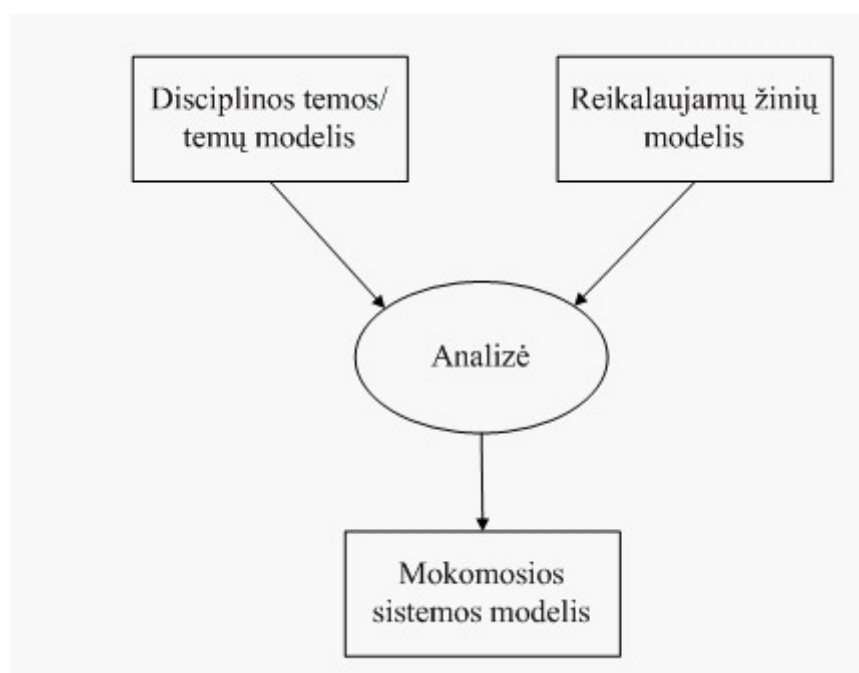
Būtina, kad mokomoji sistema, programa dirbtų stabiliai ir būtų apsaugota nuo korektiškų duomenų įvedimo.

Sistemos, programos scenarijus ir jų programinis realizavimas tikslinamas testavimo metu, kadangi tik tada išryškėja anksčiau nepastebėti jų trūkumai. Mokomosios sistemos, programos kūrimo procesas reikalauja kruopštaus darbo. Kai kurių stebėjimų rezultatai rodo, kad vidutinio sudėtingumo mokomajai sistemai, programai kurti autorių kolektyvas (pedagogas, psichologas, dailininkas, programuotojas) sugaišta ne mažiau kaip kelis mėnesius.

2.3 MOKOMŲJŲ SISTEMŲ PARUOŠIMO REKOMENDACIJOS

Prieš imantis mokomosios sistemos kūrimo, reikia atsakyti į šį klausimą, ką mokyti?

Sudaromos disciplinos, kurioms rašomos mokomoji sistema, temos arba temų modelis ir reikalaujamų mokinio žinių modelis. Tikrinama, ar abu šie modeliai yra suderinami.

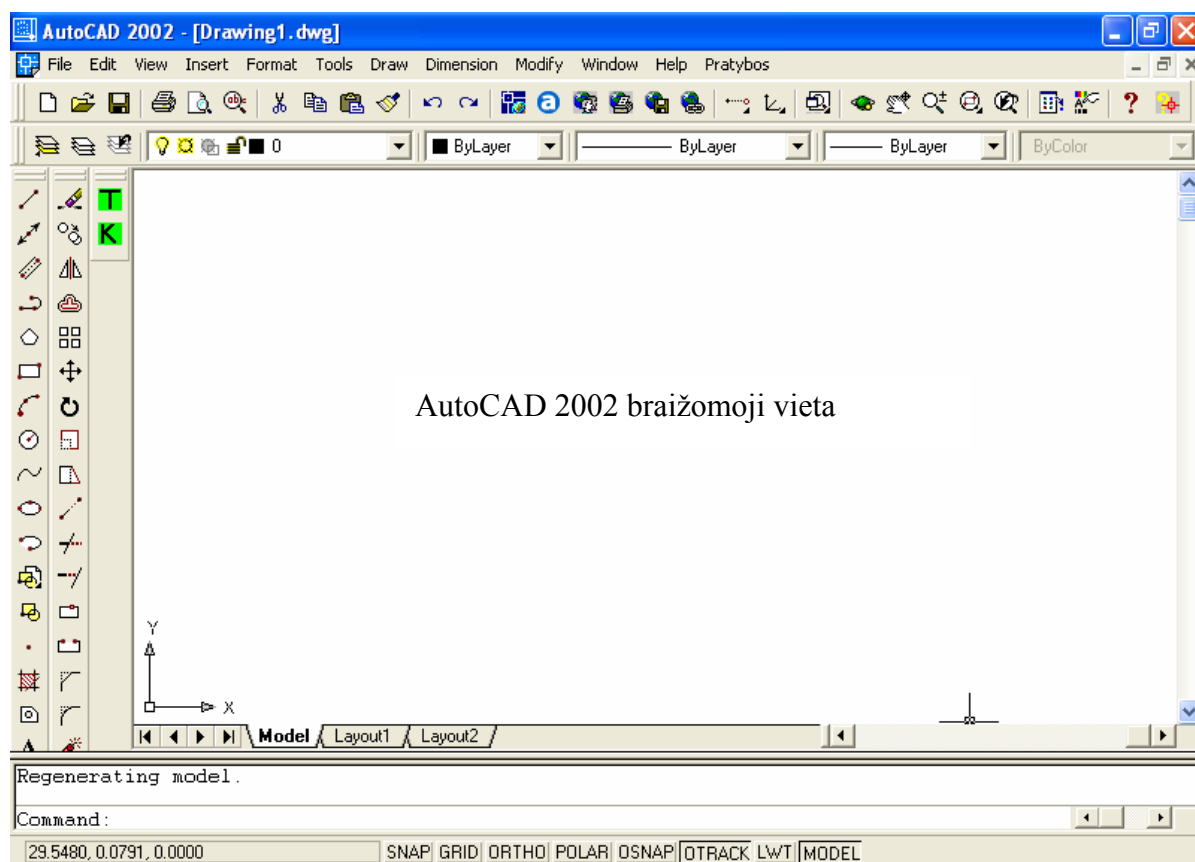


4 pav. Mokomųjų sistemų kūrimo schema

Ruošiant mokomąją sistemą, programą, reikia įvertinti, kad ja bus mokomas asmuo, kuris galbūt pirmą kartą mokomas tokio pobūdžio mokomąja sistema. Dėl to, turi būti įvertinta ir mokomojo asmens paruošimo mokomajai sistemai galimybė – pradedant atlikimo instrukcija su pavyzdžiu. Būtina išsami informacija apie konkrečią sistemą, taip bus pasiekti geresni mokymosi rezultatai.

2.4 AUTOCAD 2002 – MOKYMO SISTEMOS PAGRINDAS

Pirmoji AutoCAD versija buvo sukurta 1982 metais ir pavadinta MicroCAD. Ši versija buvo skirta Intel 8080 kartos kompiuteriams. MicroCAD buvo tikra braižymo ir projektavimo revoliucija. Dabar programinis paketas AutoCAD (5 pav.) yra išverstas į 18 kalbų ir naudojamas visame pasaulyje.



5 pav. Programinio paketo AutoCAD 2002 darbinis langas

Turbūt dėsninga yra tai, kad sparčiai greitėjant kompiuteriams, atitinkamai tobulėja programų lankstumas, pritaikomumas, universalumas. AutoCAD viena iš jų. AutoCAD komandų sistema yra paremta AutoLISP programavimo kalba, o objektų valdymas ir duomenys – DXF formatu. Šiomis priemonėmis galima nesudėtingai AutoCAD sistemą pertvarkyti pagal savo poreikius. Taip pat nesudėtingas yra meniu ir įrankių juostų redagavimas, dialogo lango kūrimas (panaudojant DCL (*Dialog control language*)). Šie langai taip pat yra AutoLISP valdomi. Visos šios minėtos charakteristikos leidžia sukurti patikimą mokomąją sistemą.

2.5 PROGRAMAVIMO KALBOS *VISUALLISP* PANAUDOJIMAS MOKYMO SISTEMOJE

2.5.1 PAGRINDINĖS *VISUALLISP* CHARAKTERISTIKOS

AutoLISP jau seniai yra viena iš AutoCAD sudėtinių dalių. Kartu su AutoCAD 2000 versija buvo sukurta ir nauja AutoLISP versija – VisualLISP (VLISP). Redaktorius VisualLISP yra parodytas 6 paveiksle.

```

Visual LISP for AutoCAD <Drawing1.dwg> - [Solidai.LSP]
File Edit Search View Project Debug Tools Window Help
car
(Defun DarbiniaiLaukai (/ vntTaskas1 vntTaskas2 objDarbinis vntDuomenuMasyvas objRezultatas)
  (setq vntTaskas1 (list -70 -90))
  (setq vntTaskas2 (list 165 110))
  (command "rectangle" vntTaskas1 vntTaskas2)
  (setq objDarbinis (entlast))
  (setq vntDuomenuMasyvas (list (cons 1000 "Darbinis") (cons 1000 "-70") (cons 1000 "-90") (cons
  PrikabintXDATA vntDuomenuMasyvas objDarbinis)
  ;265;-25 -> 470;110
  (command "color" "green")
  (command "rectangle" '(265 -90) '(500 110))
  (setq objRezultatas (entlast))
  (setq vntDuomenuMasyvas (list (cons 1000 "Rezultatas") (cons 1000 "265") (cons 1000 "-90") (cor
  PrikabintXDATA vntDuomenuMasyvas objRezultatas)

  (BreztiTeksta '(70 117) "Darbinis" 8)
  (BreztiTeksta '(275 117) "Rezultatas" 8)
  )

;
(defun ArGerasSablonas (/ layoutai iKiek i streilinis)
  (setq layoutai (layoutlist))
  (setq bYra nil)
  (setq iKiek (length layoutai))
  (setq i 0)
  (while (< i iKiek)
    (setq streilinis (nth i layoutai))
    (if (= (strcase streilinis) "VAIZDAS")
      (progn

```

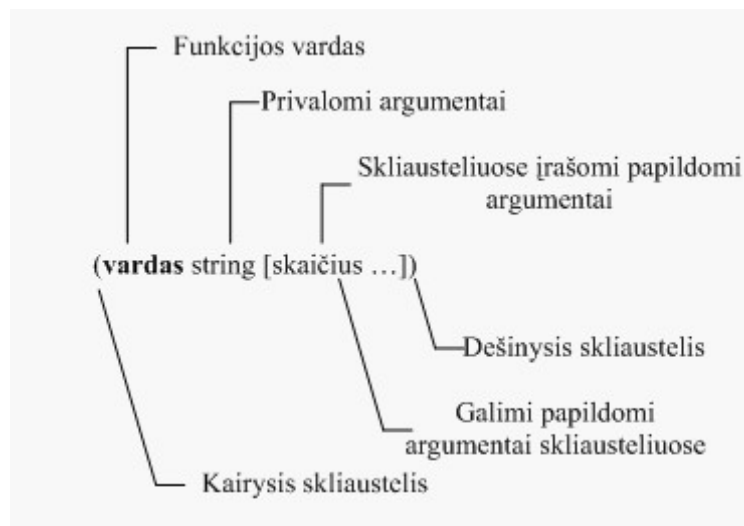
Edit: C:/Documents and Settings/Irmantas/Desktop/Diplominio darbo failai/Diplominis darbas/kurs/Irmantui/Solidai.L: L 00001 C 00001

6 pav. VisualLISP programų kūrimo įrankis

VLISP žymiai praplėtė programavimo kalbos AutoLISP galimybes:

- ✓ AutoLISP programų kūrimas pasidarė profesionalnesnis;
- ✓ objektų sąveika leidžiama Microsoft Active X technologijos būdu;
- ✓ integruota kūrimo aplinka (IDE) supaprastinta programos kodo kūrimą, modifikavimą, testavimą, transliavimą;
- ✓ suteikia galimybę sukurti autonominę programą.

VLISP funkcijos formatas yra nurodytas 7 paveiksle.



7 pav. VisualLISP funkcijos formatas

2.5.2 VISUALLISP DARBO PRINCIPAS

Programuojant VLISP kalba galima naudoti AutoCAD komandas, o dirbant AutoCAD tiesiogiai kviesti VLISP funkcijas. VLISP išraiškos gali būti rašomos tiesiogiai AutoCAD komandų eilutėje arba įrašomas į failą ir vykdomas vėliau.

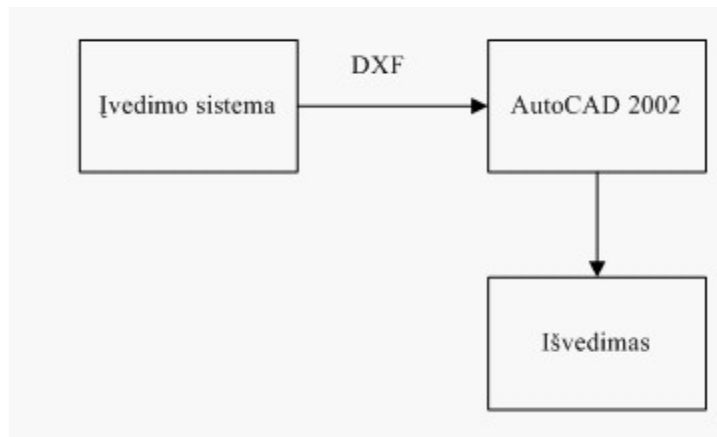
Pradedant darbą su VLISP atidarome darbinį langą, kuriame rašome programos kodą. Bylą išsaugome **.lsp* formatu. Tam, kad galėtume vykdyti savo sukurtą programą, ją turime transliuoti. Suaktyvinus AutoCAD langą, galime įvykdyti sukurtą funkciją.

Esant funkcijos vykdymui, paeiliui yra vykdoma kiekviena programinio kodo eilutė. Kiekvienos komandos pradžią identifikuoja atidarantis skliaustas, pabaigą – uždarantis. Esant skliaustų neatitikimui, funkcija bus nevykdoma.

2.6 DXF FORMATO PANAUDOJIMAS MOKOMOJOJE SISTEMOJE

2.6.1 PAGRINDINĖS DXF FORMATO CHARAKTERISTIKOS

DXF kaip ir VLISP yra neatsiejama AutoCAD dalis. Kaip pažymi kūrėjai, bet kokio vartotojo duomenys pavaizduoti ekrane, gali būti išreikšti DXF formatu. DXF formato panaudojimą matome 8 paveiksle.



8 pav. DXF formato panaudojimas

Šio formato pagrindinė charakteristika yra tokia, kad kiekvienas informacijos vienetas išreiškiamas dviejomis komponentėmis – kodu ir reikšme.

Pagrindinės kodų reikšmės yra 1 lentelėje.

1 lentelė. Pagrindinės DXF kodų reikšmės

Kodas	Reikšmė
0	Primityvo pavadinimas
1	Teksto reikšmė
5	Objekto identifikatorius
8	Sluoksnis
10, 20, 30	Pradžios koordinatės
11, 21, 31	Pabaigos koordinatės
40	Spindulys
42	Diametras
62	Spalva

2.6.2 DXF NAUDOJIMO PRINCIPAS

DXF naudojimas paremtas šio formato pagrindine charakteristika (kiekvienas informacijos vienetas išreiškiamas dvejomis komponentėmis: kodu ir reikšme). Remiantis šia charakteristika, galima manipuluoti objektų parametrais, juos tikrinti, pakeisti. Viena iš pažymėtinų savybių yra ta, kad DXF kodo nereikia kurti pačiam. AutoCAD sistema, braižant objektą, automatiškai sukuria DXF kodą. Be to, norint kodus panaudoti nėra būtina išsaugoti išorinėje byloje. Išsaugojus brėžinį DXF formatu, AutoCAD sistema jį visada atkurs. Tai patartina tiems vartotojams, kurių kietajame diske nėra daug laisvos vietos, kadangi darbas išsaugotas formatu *.dxf užima mažiau vietos nei formatu *.dwg.

3 PROGRAMINĖS ĮRANGOS PROJEKTAS

3.1 AUTOCAD 2002 MOKOMOSIOS SISTEMOS SCENARIJUS

Reikia sukurti mokomąją sistemą, programą, kuri būtų skirta kompiuterinės grafikos kompiuteriniam darbui automatizuoti, bei išmokti dirbti su AutoCAD 2002 grafiniams primityvams braižyti ir redagavimo komandomis. Dirbant su mokomąja sistema reikia remtis šiuo scenarijumi.

Scenarijus:

1. Vartotojas gali pasirinkti meniu punktą „Pratybos“, kuriame jis pasirenka darbo temą ir atitinkamą užduotį pagal ją. Naudojant meniu punktą „Darbinis brėžinys“ užkraunamas pratybų fragmentas su darbo atlikimo scenarijumi ir reikalaujama rezultatu.
2. Atidarytame pratybų lange vartotojas gali vykdyti nurodytas instrukcijas.
3. Atlikus užduotį vartotojas gali ją patikrinti. Atitinkamai pagal gaitą rezultatą jis gali pasirinkti naują užduotį arba pakartoti senąją.

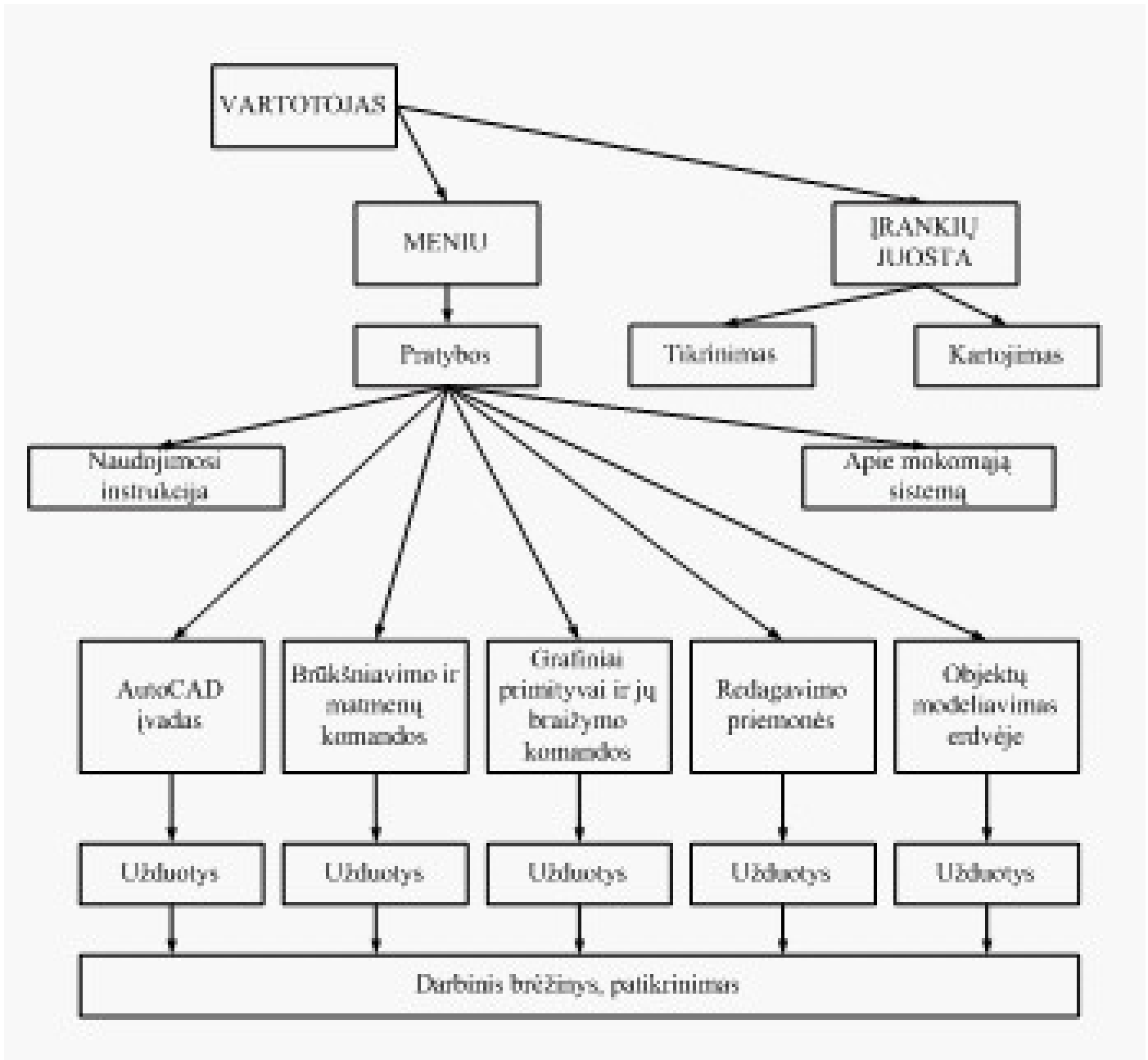
3.2 FUNKCINIAI REIKALAVIMAI

Pagrindiniai projekto tikslai yra šie:

- ✓ Sukurti nesunkiai įsisavinamą, nebrangią programinę įrangą, leidžiančią vartotojui išmokti valdyti ir realizuoti AutoCAD komandas;
- ✓ Tikrinimo algoritmas turi nepriklausyti nuo vartotojo daromų klaidų;
- ✓ Sudaryti lengvai valdomą ir programuojamą, lankstų programinį modelį, kuris įgalintų programuotoją greitai ir kokybiškai modifikuoti programinę įrangą tolimesniems tyrimams;
- ✓ Užtikrinti vartotojo užduoties vykdymo teisingumą.

3.3 UŽDAVINIO SPRENDIMAS

Uždavinį reikia spręsti taip, kad vartotojui būtų lengva naudotis sukurtu produktu. Todėl yra sudaroma funkcijų struktūrinė schema (9 pav.), pagal kurią bus kuriamas programinis produktas.



9 pav. Funkcijų struktūrinė schema

1. Vartotojas lengviau priima struktūrizuotą informaciją, todėl informacijos kvantai pateikiami daugialype sistema.
2. Mokomojoje sistemoje informacija pateikiama keturiais lygiais:
 - 1) meniu, įrankių juosta;
 - 2) darbų temos, informacija apie mokymo sistemą;
 - 3) darbų užduotys;
 - 4) darbinis brėžinys ir jo patikimumas.
3. Informacija skirtinguose lygiuose pateikiama tiek tiesine, tiek išsišakojusia tvarka.
4. Mokomojoje sistemoje naudojami mišrūs informacijos kvantai, t. y. judantys ir nejudantys objektai ir pan.
5. Darbo metu generuojami pranešimai generuojami yra trumpi ir išsamūs.
6. Informacija pateikiama glausta ir iliustruota, o raidės gerai įskaitomos. Svarbi informacija išdėstoma ekrano dalyje ir išskiriama kita spalva.
7. Darbo įvertinimas pateikiamas į ekraną išvedamu pranešimu.

Sukurta mokomoji sistema atlieka tokias funkcijas:

- ✓ supratimo treniravimo ir pratybų;
- ✓ atsiminimo treniravimo ir pratybų;
- ✓ kontrolės.

Pagal atliekamas funkcijas mokomoji sistema yra demonstracinė, treniravimo ir pratybų.

3.4 PROGRAMINIŲ MODULIŲ SPECIFIKACIJOS

Paleidus programos meniu modulį, jis suteikia galimybę iškviesti tokius modulius:

1. Darbinis brėžinys (darbo užduoties, instrukcijos programos);
2. Patikrinimas (tikrinimo programos);
3. Informacija (dialogo lango programa);
4. Pakartojimas (atnaujinimo programa iš įrankių juostos);
5. Tikrinimas (tikrinimo programa iš įrankių juostos).

3.4.1 MODULIO „MENIU“ APRAŠYMAS

„**Meniu**“ modulis saugomas faile **msistema.mnu**. Šiame faile taip pat saugoma įrankių juostos aprašai. Tai specialios sintaksės failas, kurį naudodama AutoCAD sistema sugeneruoja meniu ir įrankių juostas. Meniu sudarymas ir koregavimas sudaromas POP dalyje, naudojant žymės priešdėlius, kuriais suformuojamas pakopinis meniu.

Įrankių juosta aprašoma TOOLBARS dalyje.

Abiem atvejais vykdomos funkcijos nurodomos AutoLISP programavimo priemonėmis.

Programos kodas pateiktas **Priede**.

3.4.2 MODULIO „DARBINIS BRĖŽINYS“ APRAŠYMAS

„**Darbiniam brėžiniui**“ atvaizduoti yra suprogramuotos **93** funkcijos, kurios yra sudarytos pagal identišką algoritmą:

- ✓ standartinės funkcijos (ekrano išvalymas, teksto stiliaus nustatymas, OSNAP pasirinkties išjungimas, reikiamų sluoksnių sutvarkymas);
- ✓ darbo pavadinimas (visose funkcijose darbo pavadinimo koordinatės yra vienodos, todėl kituose moduluose jis naudojamas kaip atitinkamos užduoties identifikatorius);
- ✓ užduoties formuluotė;
- ✓ darbo atlikimo scenarijus (remiantis A. Lenkevičiaus ir J. Maticcko vadovėliu „Kompiuterinė grafika“ bei E. T. Romanyčevos (Э. Т. Романычева) knyga „Компьютерная технология инженерной графики в среде AutoCAD 2000“ suformuluojamas atskiras scenarijus);
- ✓ darbinės srities išskyrimas (nubraižomi darbinės srities kontūrai ir reikalingi pradiniai objektai);
- ✓ rezultato srities išskyrimas (nubraižomas rezultato brėžinys);
- ✓ koordinačių sistemos transformavimas konkrečiam brėžiniui.

Modulis yra skirtas atvaizduoti užduotis bei dirbti su užduotimis.

Programos kodas pateiktas **Priede**.

3.4.3 MODULIO „PATIKRINIMAS“ APRAŠYMAS

Kiekvienai „**Darbinio brėžinio**“ funkcijai yra reikalinga atitinkama „**Patikrinimo**“ funkcija. Todėl „Patikrinimui“ yra suprogramuotos **93** funkcijos, kurios yra sudarytos pagal identišką algoritmą:

- ✓ darbinės srities objektų nuskaitymas (išskiriami objektai ir jų skaičius iš darbinės srities);
- ✓ konkrečių parametrų išskyrimas, pasinaudojant DXF formato kodais (tam tikro tipo objektai identifikuojami skirtingais DXF parametrais);
- ✓ teisingų reikšmių priskyrimas (priskiriamos žinomos teisingos reikšmės, kurios bus naudojamos palyginimui);
- ✓ parametrų palyginimas (palyginami jau žinomi teisingi ir surasti parametrai);
- ✓ atsakymo paskelbimas (įvertinus teisingumą, pateikiamas pranešimas „Teisingai“ arba „Neteisingai, paspauskite mygtuką -Pakartoti-“).

Modulis skirtas atliktos užduoties teisingumui įvertinti.

Programos kodas pateiktas **Priede**.

3.4.4 MODULIO „INFORMACIJA“ APRAŠYMAS

Modulis „**Informacija**“ skirtas informacinio dialogo formavimui. Dialogas yra aprašomas DCL kalba, o jo vykdymas realizuojamas VisualLISP kalba. Pateiksiu šio modulio algoritmą:

- ✓ DCL kalbos priemonėmis yra aprašoma dialogo struktūra;
- ✓ AutoLISP kalba yra inicializuojamas dialogas;
- ✓ paveiksluko įkėlimas į dialogo langą (paveikslukas saugomas byloje „**ktu.sld**“);
- ✓ dialogo valdymas (reakcija į mygtuko paspaudimą).

Dialogo lange pateikiama informacija apie mokomąją sistemą ir jo autorius.

Programos kodas pateiktas **Priede**.

3.4.5 MODULIO „PAKARTOJIMAS“ APRAŠYMAS

Modulis „**Pakartojimas**“ skirtas nesėkmės atveju atnaujinti brėžinį. Šis modulis vykdomas iš įrankių juostos. Vykdytas yra paremtas tokiu algoritmu:

- ✓ objektų nuskaitymas (išskiriama tekstinė informacija iš ekrano);
- ✓ identifikuojama užduotis (identifikacija vykdoma pagal užduoties pavadinimą);
- ✓ reikiamo darbo suradimas ir naujo darbo užkrovimas (įvykdoma atitinkama „Darbinio brėžinio“ funkcija).

Programos kodas pateiktas **Priede**.

3.4.6 MODULIO „TIKRINIMAS“ APRAŠYMAS

Modulis „**Tikrinimas**“ skirtas įvertinti užduoties teisingumą. Šis modulis vykdomas iš įrankių juostos. Vykdytas yra paremtas tokiu algoritmu:

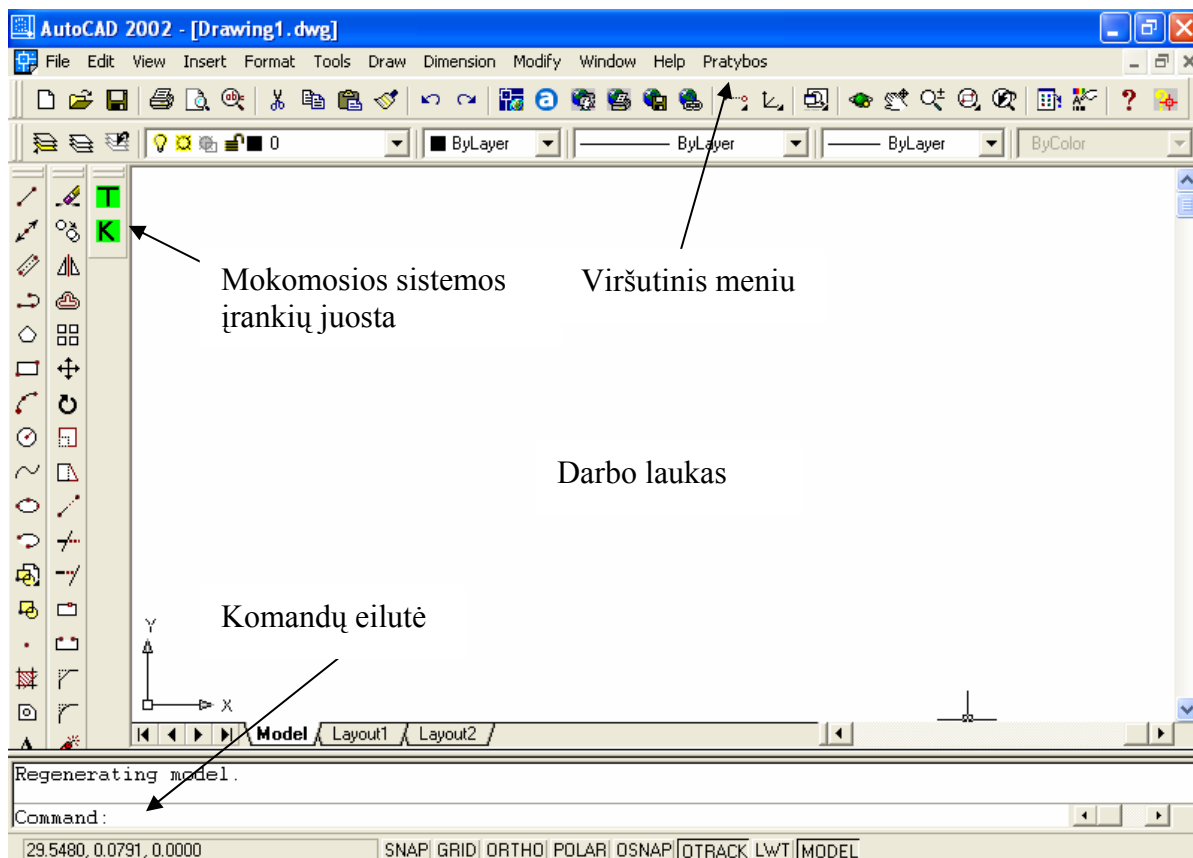
- ✓ objektų nuskaitymas (išskiriama tekstinė informacija iš ekrano);
- ✓ identifikuojama užduotis (identifikacija vykdoma pagal užduoties pavadinimą);
- ✓ reikiamo tikrinimo suradimas ir įvykdymas (įvykdoma atitinkama „**Patikrinimo**“ modulio funkcija).

Programos kodas pateiktas **Priede**.

3.5 MOKOMOSIOS SISTEMOS VALDYMAS

3.5.1 MENIU STRUKTŪRA BEI VALDYMAS

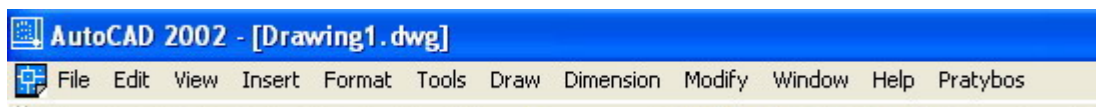
Pagrindinis meniu yra viršutinėje programos lango dalyje, o įrankių juostos meniu (du mokomosios sistemos mygtukai) yra kairėje lango pusėje (10 pav.).



10 pav. AutoCAD vartotojo darbo aplinka

1. Pagrindinis arba pirmo lygio meniu.

Pagrindinio arba pirmo lygio meniu turinys parodytas 11 paveiksle.



11 pav. Pagrindinis arba pirmo lygio meniu

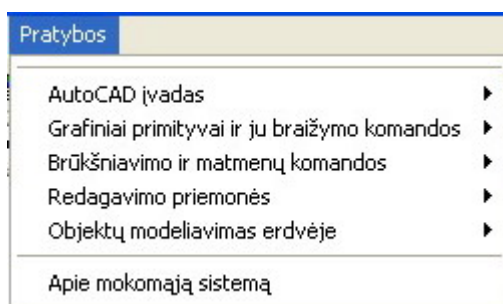
Pagrindinio meniu punktą galima pasirinkti tokiais būdais:

- ✓ **Pele.** Nuvedus pele kursorių ties meniu punktu, punktas sureaguos ir tada paspausti pelės kairįjį klavišą. Sureagavimas – tai trimačio efekto suteikimas punktui.
- ✓ **Alt + P klavišų kombinacija.** Išskleidžiamas pagrindinio meniu punktas „Pratybos“.

2. Papildomas meniu. Antras lygis.

Pagrindinio meniu punkto „Pratybos“ antrame lygyje yra šie punktai (12 pav.):

- ✓ „AutoCAD įvadas“;
- ✓ „Grafiniai primityvai ir jų braižymo komandos“;
- ✓ „Brūkšniavimo ir matmenų komandos“;
- ✓ „Redagavimo priemonės“;
- ✓ „Objektų modeliavimas erdvėje“;
- ✓ „Apie mokomąją sistemą“.



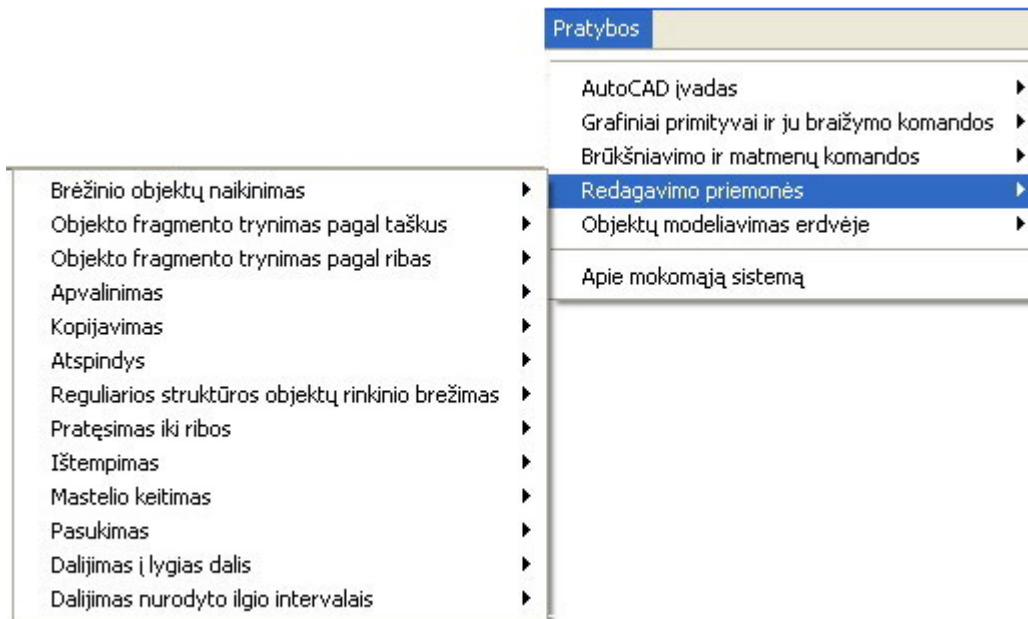
12 pav. Pagrindinio meniu punkto „Pratybos“ antras lygis

Antrame meniu lygyje punktus galime išsirinkti pele arba klavišais su rodyklėmis. Pažymėtas punktas antrame lygyje išskiriamas tamsia juosta.

3. Papildomas meniu. Trečias lygis.

Dauguma antro lygio meniu punktų turi trečią šoninio meniu pavidalo lygį (13 pav.). Šie punktai pažymėti dešinėje esančia rodykle. Į jį iš antro lygio galima pateikti taip:

- ✓ pelės rodyklę tiesiog nuvedus ant punkto, parodomas to punkto turinys;
- ✓ kursoriumi (klavišai su rodyklėmis), norimo punkto turinys bus parodomas, pažymėjus šį punktą ir paspaudus kursoriaus rodyklę „→“.

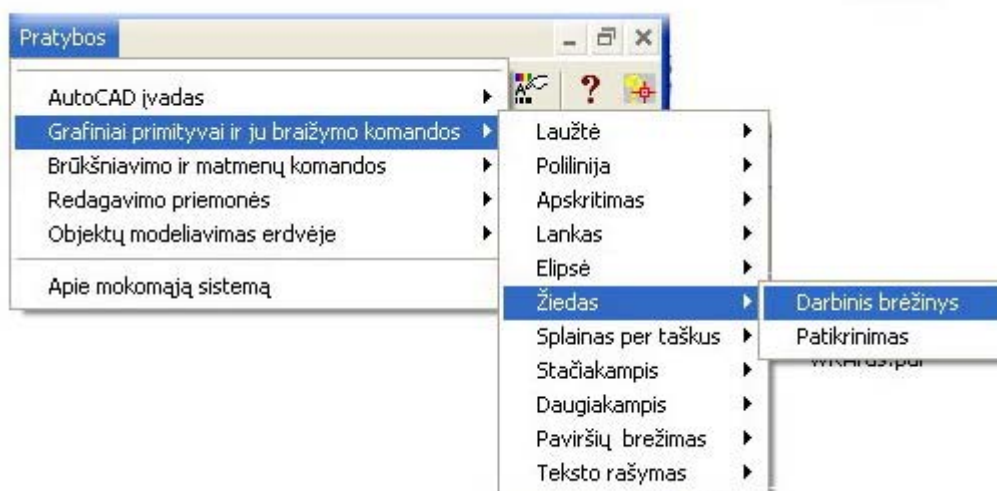


13 pav. Pagrindinio meniu punkto „Pratybos“ trečias lygis

4. Papildomas meniu. Ketvirtas lygis.

Visi trečio lygio meniu punktai turi ketvirtą šoninio meniu pavidalo lygį (14 pav.). Šie punktai pažymėti dešinėje esančia rodykle. Į jį iš trečio lygio galima patekti taip:

- ✓ pelės rodyklę tiesiog nuvedus ant punkto, parodomas to punkto turinys;
- ✓ kursoriumi (klavišai su rodyklėmis), norimo punkto turinys bus parodomas, pažymėjus šį punktą ir paspaudus kursoriaus rodyklę „→“.



14 pav. Pagrindinio meniu punkto „Pratybos“ ketvirtas lygis

3.5.2 ĮRANKIŲ JUOSTOS VALDYMAS

Įrankių juosta sudaro du mygtukai: „T“ („Patikrinimas“) ir „K“ („Kartojimas“). Nuvedus pele kursorių ties vienu iš dviejų mygtukų, mygtukas sureaguos ir tada paspausti pelės kairįjį klavišą. Sureagavimas – tai trimačio efekto suteikimas punktui. Paspaudus mygtuką vykdoma atitinkama funkcija. Įrankių juosta pavaizduota 15 paveiksle.



15 pav. Mokomosios sistemos įrankių juosta

3.6 MENIU PUNKTŲ APRAŠYMAS

Kaip buvo parodyta 12 paveiksle, į pagrindinį meniu įeina punktas „Pratybos“. Šį punktą sudaro tokie papunkčiai:

- ✓ „AutoCAD įvadas“;
- ✓ „Grafiniai primityvai ir jų braižymo komandos“;
- ✓ „Brūkšniavimo ir matmenų komandos“;
- ✓ „Redagavimo priemonės“;
- ✓ „Objektų modeliavimas erdvėje“;
- ✓ „Apie mokomąją sistemą“.

Papunktyje „AutoCAD įvadas“ pateikiamos A. Lenkevičiaus ir J. Maticko vadovėlio „Kompiuterinė grafika“ 6 skyriaus užduotys.

Papunktyje „Grafiniai primityvai ir jų braižymo komandos“ pateikiamos A. Lenkevičiaus ir J. Maticko vadovėlio „Kompiuterinė grafika“ 7 skyriaus užduotys (komandų „Lankai“ užduotys pateikiamos iš E. T. Romanyčevos (Э. Т. Романычева) knygos „Компьютерная технология инженерной графики в среде AutoCAD 2000“).

Papunktyje „Brūkšniavimo ir matmenų komandos“ pateikiamos A. Lenkevičiaus ir J. Maticko vadovėlio „Kompiuterinė grafika“ 8 skyriaus užduotys.

Papunktyje „Redagavimo priemonės“ pateikiamos A. Lenkevičiaus ir J. Maticko vadovėlio „Kompiuterinė grafika“ 9 skyriaus užduotys.

Papunktyje „Objektų modeliavimas erdvėje“ pateikiamos E. T. Romanyčevos (Э. Т. Романычева) knygos „Компьютерная технология инженерной графики в среде AutoCAD 2000“.

Papunktyje „Apie mokomąją sistemą“ dialogo lange pateikiama informacija apie mokomąją sistemą, jos autorius, sukūrimo datą ir įstaigą.

Kai kurie antro lygio meniu punktai taip pat yra skirstomi į papunkčius.

Punktas „**AutoCAD įvadas**“ skirstomas į tokius papunkčius:

- ✓ „Susiejimas su objektų traukos taškais“ (skirstomas į „Apskritimo centras, atkarpos sankirtos taškas“, „Statmuo iš apskritimo centro į atkarpą“, „Liestinė tarp dviejų apskritimų“ papunkčius);
- ✓ „Koordinačių filtrai“;
- ✓ „Tarpinis bazinis taškas“;
- ✓ „Geometrinė skaičiuoklė“.

Punktas „**Grafiniai primityvai ir jų braižymo komandos**“ skirstomos į tokius papunkčius:

- ✓ „Laužtė“ (skirstomas į „Absoliutinės koordinatės“, „Santykinės stačiakampio koordinatės“, „Santykinės polinės koordinatės“, „Mišrios koordinatės“ papunkčius);
- ✓ „Poliliniija“;
- ✓ „Apskritimas“ (skirstomas į „Centras, spindulys (C, R)“, „Centras, skersmuo (C, D)“, „Dvi liestinės, spindulys (T, T, R)“, „Du taškai (2P)“, „Trys taškai (3P)“, „Trys liestinės (T, T, T)“ papunkčius);
- ✓ „Lankas“ (skirstomas į „Nurodant pradžią, centrą, pabaigą“, „Nurodant pradžią, centrą, kampą“, „Nurodant pradžią, centrą, ilgį“, „Nurodant pradžią, pabaigą, kampą“, „Nurodant pradžią, pabaigą, kryptį“, „Nurodant pradžią, pabaigą, spindulį“, „Nurodant centrą, pradžią, pabaigą“, „Nurodant centrą, pradžią, kampą“, „Nurodant centrą, pradžią, ilgį“ papunkčius);
- ✓ „Elipsė“ (skirstomas į „Centras (C)“, „Ašis, Pabaiga (A, E)“, „Lankas (A)“ papunkčius);
- ✓ „Žiedas“;
- ✓ „Splainas per taškus“;
- ✓ „Stačiakampis“ (skirstomas į „Priešingų kampų koordinatės“, „Suapvalinti kampai“ papunkčius);

- ✓ „Daugiakampis“ (skirstomas į „5 kraštinės“, „6 kraštinės“, „Kraštinės ilgis“ papunkčius);
- ✓ „Paviršių brėžimas“ (skirstomas į „2D paviršius“, „3D paviršius“ papunkčius);
- ✓ „Teksto rašymas“ (skirstomas į „Nurodant vieną tašką“, „Nurodant du taškus“, „Stilius, pradinis taškas“ papunkčius).

Punktas „**Brūkšniavimo ir matmenų komandos**“ skirstomas į tokius papunkčius:

- ✓ „Sričių užbrūkšniavimas“ (skirstomas į „Normal stilius“, „Outer stilius“, „Ignore stilius“, „Srities užbrūkšniavimas, nurodant linijas per taškus“ papunkčius);
- ✓ „Matmenų rašymas“ (skirstomas į „Atstumas tarp dviejų taškų“, „Atkarpos ilgis“ papunkčius);
- ✓ „Greitasis matmenų rašymas“ (skirstomas į „Nuo pirmo taško“, „Nuo paskutinio taško“, „Lanko spindulys“, „Skersmuo tarp taškų“, „Matmuo, pasukant kampu“, „Kampas tarp atkarpu“, „Lygiagretūs matmenys“ papunkčius).

Punktas „**Redagavimo priemonės**“ skirstomas į tokius papunkčius:

- ✓ „Brėžinio objektų naikinimas“;
- ✓ „Objekto fragmento trynimas pagal taškus“ (skirstomas į „Nuo 2 iki 1 taško“, „Nuo 1 iki 2 taško“ papunkčius);
- ✓ „Objekto fragmento trynimas pagal ribas“ (skirstomas į „Išsikišimų trynimas“, „Figūros išėmimas“ papunkčius);
- ✓ „Apvalinimas“ (skirstomas į „Sujungimas objektu“, „Užapvalinimas tiesų susikirtime“, „Apskritimų sujungimas“ papunkčius);
- ✓ „Kopijavimas“;
- ✓ „Atspindys“;
- ✓ „Reguliarios struktūros objektų rinkinio brėžimas“;
- ✓ „Pratęsimas iki ribos“;
- ✓ „Ištempimas“;
- ✓ „Mastelio keitimas“;
- ✓ „Pasukimas“;
- ✓ „Dalijimas į lygias dalis“;
- ✓ „Dalijimas nurodyto ilgio intervalais“.

Punktas „**Objektų modeliavimas erdvėje**“ skirstomas į tokius papunkčius:

- ✓ „3D objektai“ (skirstomas į „Dėžė (Box)“, „Sfera (Sphere)“, „Cilindras (Cylinder)“, „Kūgis (Cone)“, „Pleištas (Wedge)“, „Torusai (Torus)“, „Sukurti ištemptą objektą (Extrude)“, „Sukurti ištemptą objektą su kampu (Extrude)“, „Sukurti objektą – „sukinį“ (Revolve)“ papunkčius);
- ✓ „3D objektų redagavimas“ (skirstomas į „Modifikuoti kubą (Chamfer)“, „Apjungimas (Union)“, „Atėmimas (Subtract)“, „Sukirtimas (Intersect)“, „Modifikuoti 3D objektą ištempiant (Extrude faces)“, „Modifikuoti 3D objektą pernešant sienelę (Move faces)“, „Modifikuoti 3D objektą perkeliant sienelę (Offset faces)“, „Modifikuoti 3D objektą pasukant sienelę (Rotate faces)“, „Nukopijuoti 3D objekto sienelę (Copy faces)“ papunkčius).

Visi šie papunkčiai skirti konkrečios užduoties atlikimui (menu pavadinimai atitinka užduoties temą).

Taip pat kiekviena čia išvardinta užduotis skirstoma į du papunkčius:

- ✓ „Darbinis brėžinys“;
- ✓ „Patikrinimas“.

Papunktis „**Darbinis brėžinys**“ skirtas išvesti užduotį, veiksmų planą ir darbinį lauką į ekraną.

Papunktis „**Patikrinimas**“ skirtas patikrinti atliktos užduoties teisingumą.

3.7 NEFUNKCINIAI REIKALAVIMAI

3.7.1 PROGRAMINĖS IR APARATŪRINĖS ĮRANGOS REIKALAVIMAI

Išnagrinėjus galimus duomenų apdorojimo poreikius, programos sąsajas bei jų reikalavimus sistemoms, planuojami programinės bei aparatūrinės įrangos reikalavimai būtų tokie:

Programinė įranga:

- ✓ Mokomosios sistemos greitis priklausys nuo turimos techninės įrangos;

- ✓ Planuojamas mokomosios sistemos dydis: iki 2 MB;
- ✓ Mokomoji sistema turi veikti AutoCAD 2002 (arba AutoCAD 2000) aplinkoje;
- ✓ OS: plačiausiai naudojamos operacinės sistemos Windows 9x, ME, NT, 2000, XP;

Techninė įranga yra aprašyta 2 lentelėje.

2 lentelė Techninės priemonės

Komponentai	Minimalūs reikalavimai	Rekomenduojama
Kompiuteris (AK)	Pentium 133	Pentium II 300
Operatyvioji atmintis (RAM)	32 MB	64 MB
Kompaktinio disko grotuvas (CD-ROM)	4 x	12 x
Kietasis diskas (HDD)	800 MB	2 GB
Garso plokštė	-	Suderinta su Windows
Grafinė plokštė	Video grafinis adapteris (VDA), sugebantis atvaizduoti 256 spalvas ir palaikantis 800x600 gebą.	VGA 32, 768 spalvos, palaikantis 1024x768 skiriamąją gebą.

3.7.2 PROJEKTO PATIKIMUMAS

Programa turi veikti nesukeldama vartotojo sistemos žlugimo. Programa turi veikti tiksliai kaip nurodyta užsakovo dokumentacijoje ir neklaidinti vartotojo.

Mokomosios sistemos darbinis brėžinys yra užrakintas kitame sluoksnyje, nei vartotojo darbo langas, todėl jis negali sugadinti pradinius duomenis. Net ir sugadinus darbo pavadinimą (identifikatorių), sistema neatsidurs aklavietėje, tokiu atveju vykdant patikrinimą, brėžinio atnaujinimą iš įrankių juostos, pasirodys pranešimas, kuris yra parodytas 16 paveiksle.



16 pav. Pranešimas nesant identifikatoriui

Be to šias funkcijas galima įvykdyti iš meniu. Patikrinimo funkcija turi būti ypatingai patikima, nes yra visų vartotojo pastangų įvertinimas. Atlikus viską pagal pateiktą darbo scenarijų, įvertinimas tikrai bus teigiamas, o atlikus klaidingai – įvertis bus neigiamas.

Nors reikalaujamas objektas nubrėžtas teisingai, bet rezultatas yra neigiamas, nes darbiname lange yra nereikalingų objektų.

Projekto patikimumui įvertinti naudosiu rodiklį POFOD (*Probability of failure on demand*): nesėkmingo bandymo pasinaudoti sistema tikimybė, t. y. tikimybė, kad bandant pasinaudoti sistema, įvyks klaida.

Mano projekto POFOD lygis 0,0005 tai reiškia, kad vienas iš dviejų tūkstančių bandymų pasinaudoti sistema gali būti nesėkmingas. Projektuojamos programos klaidos turi būti pašalintos projektavimo ir testavimo metu, bet lieka tikimybė operacinės sistemos sutrikimų ir kitų faktorių.

3.7.3 PROJEKTO GRAFIKŲ IR IŠLAIDŲ PLANAS

Projekto išlaidos yra minimalios, kadangi projektas vykdomas savarankiškai kaip magistrinis – mokslinis darbas. Pagrindines projekto išlaidas sudaro laikas skirtas projektui kurti bei programai rašyti. Projekto atlikimui naudojami nemokami informacijos šaltiniai ir kiti resursai.

Projekto grafikas.

Projektas buvo kuriamas su atskiromis pertraukomis nuo 2002 metų:

- ✓ 2002-09-01 – 2003-01-30: sugalvota projekto idėja. Padaryta uždavinio aplinkos analizė. Išskelti tikslai bei uždaviniai, nustatyti pradiniai reikalavimai.

- ✓ 2003-02-01 – 2003-06-30: pildomi reikalavimai. Apibrėžta pradinė programos struktūra. Apgalvoti keli galimi sprendimo algoritmai.
- ✓ 2003-09-01 – 2004-01-30: galutinai nustatytas brėžinio teisingumą nustatantis algoritmas ir programos struktūra.
- ✓ 2004-02-01 – 2004-05-25: programuojama vartotojo sąsaja, programuojami programos komponentai, atliekamas testavimas, rašoma programos dokumentacija.

3.8 VARTOTOJO REIKALAVIMAI

Vartotojo reikalavimai turi aprašyti funkcinis ir nefunkcinis reikalavimus taip, kad jie būtų suprantami sistemos vartotojų, kurie neturi detalių techninių žinių.

Vartotojo reikalavimus išgausime potencialiųjų užsakovų apklausos būdu, panaudojant scenarijus. Scenarijai – tai aprašymai, kaip sistema naudojama praktiškai. Jie padeda formuojant reikalavimus, kadangi žmonės reikalavimus geriau susieja su scenarijais, negu su abstrakčiais teiginiais, nusakančiais, ko jie reikalauja iš sistemos.

Programinė įranga yra skirta jos vartotojui, todėl pirmiausiai ir išsiaiškinsime jo poreikius.

Vartotojo poreikiai kuriamai programinei įrangai. Vartotojas nori programos, kuri leistų jam:

- ✓ Išsirinkti norimą užduotį;
- ✓ Patikrinti nubraižyto brėžinio teisingumą;
- ✓ Visą programinį paketą realizuoti lanksčiu modeliu, kad esant poreikiui, galima būtų greitai modifikuoti, papildyti sistemą (programos papildymą naujais sukurtais algoritmais turi daryti programuotojai).

Vartotojo reikalavimai vartotojo sąsajai.

- ✓ Patogus užduoties išvedimas į ekraną;
- ✓ Užduoties atvaizdavimas programos lange;
- ✓ Patogus ir informatyvus programos meniu;
- ✓ Pagalbos ir pranešimų sistema;

- ✓ Patogus valdymo mygtukų išdėstymas;
- ✓ Paryškinta svarbiausia informacija.

Vartotojo keliami reikalavimai užduoties modeliui.

- ✓ Vartotojas turi matyti užduoties pavadinimą, užduoties scenarijų, darbinį ir rezultato laukus;
- ✓ Rezultato lauke turi būti nubraižytas tikslus brėžinys, pagal kurį vartotojas braižys darbiniam lauke;
- ✓ Programa turi leisti nustatyti vartotojo nubraižyto brėžinio teisingumą.

Vartotojo reikalavimai produkto patikimumui.

- ✓ Programos patikimumo POFOD rodiklio lygis turi būti ne didesnis už 0,0005.

3.9 RIZIKOS ĮVERTINIMO IR MAŽINIMO PLANAS

Nustatomos pavojingos situacijos, kurios gali sukelti grėsmę sistemos saugumui, ir įvertinama su jomis susijusi rizika. Atliekama į klases suskirstytų pavojingų situacijų analizė. Programinės įrangos kūrimo procesas nuo specifikacijų iki realizacijos vykdomas, įvertinant šios analizės rezultatus.

Rizikos mažinimo būdai:

- ✓ Sistema projektuojama taip, kad pavojingos situacijos nesusidarytų visai arba bent jau netaptų nelaimingų atsitikimų priežastimi;
- ✓ Pavojingų situacijų vengimas;
- ✓ Sistemą reikia projektuoti taip, kad teisingai ją naudojant, pavojingos situacijos niekada nesusidarytų;
- ✓ Pavojingų situacijų atpažinimas ir pašalinimas;
- ✓ Sistema projektuojama taip, kad pavojingos situacijos būtų greit pastebėtos ir neutralizuotos dar prieš sukeldamos nelaimingą atsitikimą;
- ✓ Žalos ribojimas;
- ✓ Sistema projektuojama taip, kad nelaimingo atsitikimo pasekmės būtų minimalios.

Projekto rizikos:

1. **Reikalavimų pasikeitimas.** Programos pasikeitimas gali būti įtakotas programos kūrimo proceso. Gali išaugti programos kūrimo laikas, tačiau pagrindinė idėja nustatyta, todėl keistūsi tik atskiros detalės. Sprendimas: iš anksto būtina numatyti rezervinį laiką reikalavimų pasikeitimų realizavimui.
2. **Programinės įrangos atsisakymas.** Vartotojas gali atsisakyti programinės įrangos, tačiau projektas yra kuriamas moksliniais tikslais. Sprendimas: svarbiausia neapsiriboti tik vienu vartotoju. Taip pat programa gali būti panaudojama kitų programų sudėtyje arba tapti kitų programų pagrindu. Jei vartotojas atsisako produkto dėl kainos, reikia atlikti su juo derybas, pasiūlyti tolimesnį nemokamą programos atnaujinimą, pigesnes programinės įrangos aptarnavimo paslaugas, ieškoti kompromisinių sprendimų ir pan., t. y. naudoti papildomas marketingo bei kainodaros priemones.
3. **Naujų reikalavimų įvedimas bei senų keitimas.** Į galimą sistemos praplėtimą atsižvelgiama jau sistemos projektavimo procese, kurio metu siekiama, kad sistema būtų kiek galima lankstesnė ir lengvai išplečiama. Programoje taip pat numatyti tolimesni patobulinimai (naujų tikrinimo algoritmų tyrimas, vartotojo sąsajos modifikavimas), tad papildomi reikalavimai gali pareikalauti papildomo laiko jiems įdiegti. Žinoma, vartotojas turi būti informuotas apie tai, kad įdiegimams reikės papildomai laiko ir lėšų.
4. **Papildomi darbai.** Papildomi darbai labai apsunkintų projekto baigimą laiku. Sprendimas: paaiškinti vartotojui apie galimą projekto uždelimą, bandyti derėtis su juo, priešingu atveju, jam nesutikus, pirmiausiai dirbti prie svarbiausių projekto vietų, kad vartotojui atrodytų, jog praktiškai visas funkcionalumas egzistuoja ir projektas bus pristatytas laiku, o likusį funkcionalumą įdiegti vėliau.
5. **Techninės rizikos.**

Kompiuterių gedimas. Nors šiuolaikinė kompiuterinė technika ganėtinai patikima, tačiau visada išlieka tikimybė, kad kompiuteris gali sugesti. Kompiuterio komponentių (išskyrus kietąjį diską) gedimas labai didelės įtakos projekto eigai neturės, kadangi nėra sunku sugedusią komponentę pakeisti nauja, o projekto duomenims, pavyzdžiui, pagrindinės plokštės gedimas – nepakenktų. Sprendimas: pakeisti sugedusia detalę.

Kietojo disko gedimas. Tai labai įtakotų projekto eigą. Sprendimas: periodiškai susikurti projektui svarbios informacijos kopijas tiek į perrašomus kompaktinius diskus

(CD-R), tiek ir į kitus kietuosius diskus, tokiu atveju prarastos informacijos kiekis būtų nedidelis, rizika minimizuota.

Projektuojamos programinės įrangos sutrikimai. Sprendimas: visi programinės įrangos trūkumai turi būti pašalinti programos testavimo metu.

3.10 TESTAVIMAS

1. **Testavimo metodika.** Nuodugnus testavimas buvo atliktas kiekvienai funkcijai atskirai, vėliau viską sujungus į bendrą sistemą, funkcijos buvo testuojamos kaip visuma. Jokių didelių neatitikimų nebuvo pastebėta, neskaitant keleto smulkių netikslumų, kurie buvo pataisyti. Norėdamas viską pilnai patikrinti, sistema buvo duota išbandyti keletui studentų ir stebėjau ar neatsiras nenumatytų atvejų, kurių neįvertinau. Visos sistemos apsaugos veikė patikimai, tik buvo bendras nusiskundimas, jog esant tuščiam darbiniam langui ir pasirinkus tikrinimą iš įrankių juostos, nėra jokio pasipriešinimo, kad jie dar nepasirinko darbo. Šis nepatogumas buvo pašalintas.
2. **Vartotojo sąsajos testavimas.** Tikrinamas meniu veikimas. Atlikau pirminį vartotojo sąsajos testavimą. Tolimesnį testavimą vykdė visiškai su projektu nesusiję asmenys. Šis būdas man pasirodė tinkamiausias ir patikimiausias, kadangi tik realus vartotojas gali pastebėti trūkumus.

3.11 SUKURTOS MOKOMOSIOS SISTEMOS PARAMETRAI

Užduotį vartotojas pasirenka savo nuožiūra. Jos atlikimo scenarijus pateikiamas AutoCAD darbiniam lauke.

Mokomąją sistemą sudaro **93** užduotys. Kiekvienai užduočiai yra sukurta:

- ✓ „Darbinis brėžinys“;
- ✓ „Patikrinimas“.

Taip mokomąją sistemą sudaro **93** tikrinimo failai, 3 valdymo bylos *.lsp formatu, viena dialogo lango byla *.dcl formatu ir viena dialogo valdymo byla *.lsp formatu.

Visi failai užima 3,25 MB.

4. VARTOTOJO DOKUMENTACIJA

4.1 PASKIRTIS

Vartotojo instrukcija skirta AutoCAD 2002 mokomosios sistemos vartotojams (studentams ir dėstytojams). Instrukcijoje pateiktas pilnas sistemos aprašymas, reikalingas prieš pradėdant darbą su sistema.

4.2 SISTEMOS VADOVAS

1. Jeigu vartotojas yra ir administratorius, tai patariame iš pradžių skaityti 4.3 dalį – „Sistemos instaliavimo dokumentas“. Perskaityę instaliavimo instrukcijas, skaitykite 4.2 dalį – „Sistemos vadovas“.

2. Mokomoji sistema užkraunama kartu su AutoCAD 2002 programa.

3. Renkantis darbą iš meniu, atliekamas jo vykdymas.

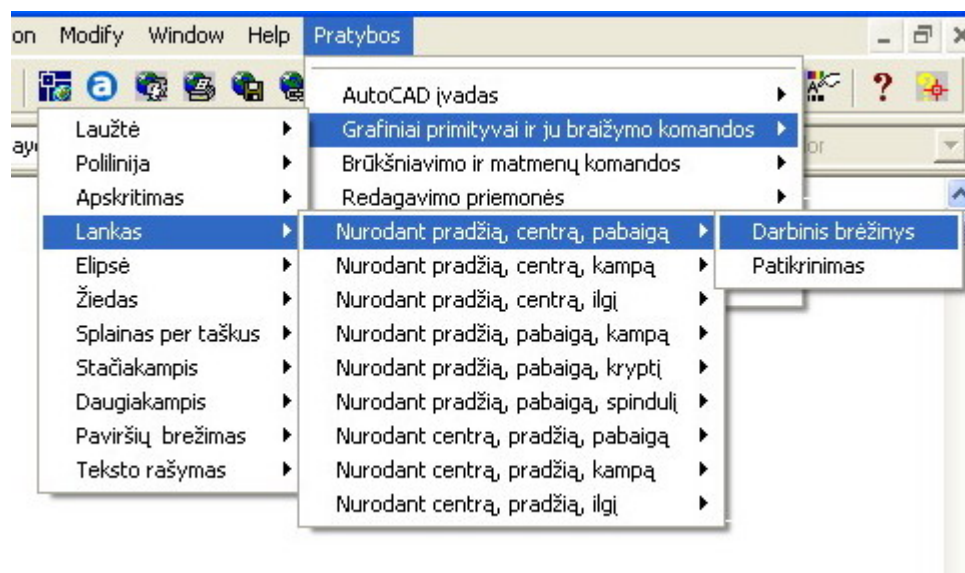
Naudojantis mokomąja sistema vartotojas turi vadovautis meniu struktūra (žr. 4.5 dalį – „Meniu punktų aprašymas“).

Suaktyvinus meniu punktą „Apie mokomąją sistemą“ galima susipažinti su mokomosios sistemos paskirtimi, autoriais, įstaiga ir sukūrimo data (17 pav.).



17 pav. Dialogo langas „Apie mokomąją sistemą“

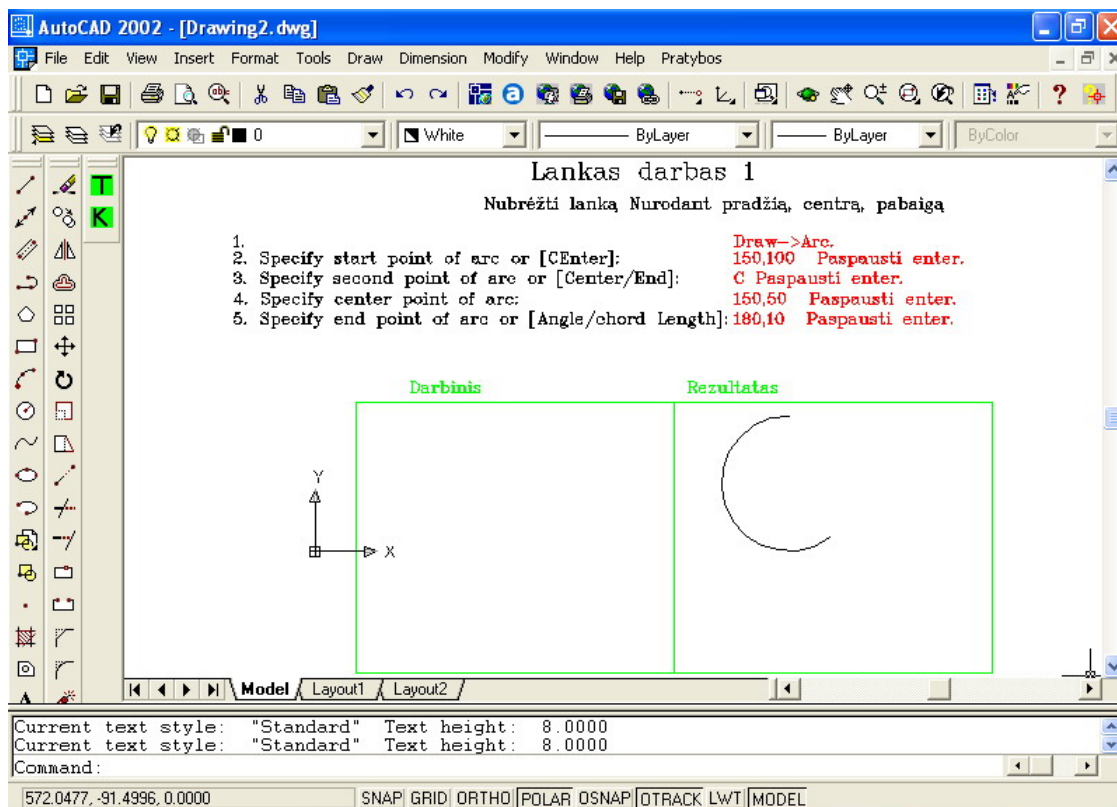
Toliau viskas priklauso nuo vartotojo pageidaujamos užduoties temos. Iš meniu pasirenkama atitinkama užduočių tema, pavyzdžiui, iš dalies „Grafiniai primityvai ir jų braižymo komandos“ temos pasirenkame „Lankas“, „Nurodant pradžią, centrą, pabaigą“, „Darbo brėžinys“ (18 pav.).



18 pav. Darbinio brėžinio pasirinkimas

Pasirinkus šį punktą (18 pav.), atidaromas pageidaujamos užduoties darbinis langas (19 pav.). Visų darbinių brėžinių struktūra yra vienoda. Galima išskirti tokias dalis:

- ✓ užduoties pavadinimas;
- ✓ užduotis;
- ✓ užduoties atlikimo scenarijus;
- ✓ darbinis langas;
- ✓ reikalaujamo rezultato langas.



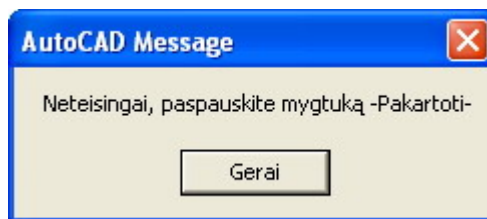
19 pav. Darbinis langas, brėžinys

Atlikus patikrinimą, jūsų darbas yra įvertinamas. Jei įvertinimas bus teigiamas, tokiu atveju Jūs galite pereiti prie naujos užduoties. Teisingai bus atliktas darbas, jei patikrinimo metu atsiras pranešimas „Teisingai“ (20 pav.).



20 pav. Užduoties įvertinimo pranešimas „Teisingai“

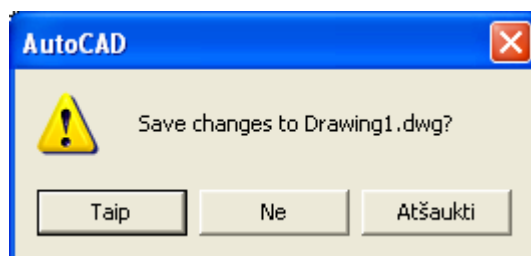
Jei įvertinimas bus neigiamas (21 pav.), jūs galite pakartoti užduotį, paspaudę mygtuką „K“, kuris reiškia „Pakartoti“ arba pasirinkti kitą darbą iš meniu.



21 pav. Užduoties įvertinimo pranešimas „Neteisingai, paspauskite mygtuką -Pakartoti-,,“

Norint baigti darbą su AutoCAD 2002 mokomąja sistema, reikia paspausti išėjimo mygtuką, kuris yra dešiniajame viršutiniame kiekvieno brėžinio lango kampe.

Iš programos išeiti galima tada, kai ekrane nėra jokio dialogo lango ar pranešimo. Paspaudus išėjimo mygtuką, atsiranda pranešimo langas (22 pav), kurioje sistema paklaus ar norite išsaugoti pakeitimus (*Save changes to Drawingx.dwg?*). Atsakius į tai teigiamai (paspausti mygtuką „**Taip**“ (*Yes*)), sistema baigs darbą ir darbas bus išsaugotas jūsų nurodytoje vietoje. Atsakius neigiamai (paspausti mygtuką „**Ne**“ (*No*)), sistema baigs darbą ir neišsaugos jūsų darbo. Jei persigalvojote ir nenorite baigti darbą, paspauskite mygtuką „**Atšaukti**“ (*Cancel*).



22 pav. Sistemos pranešimas dėl darbo baigimo

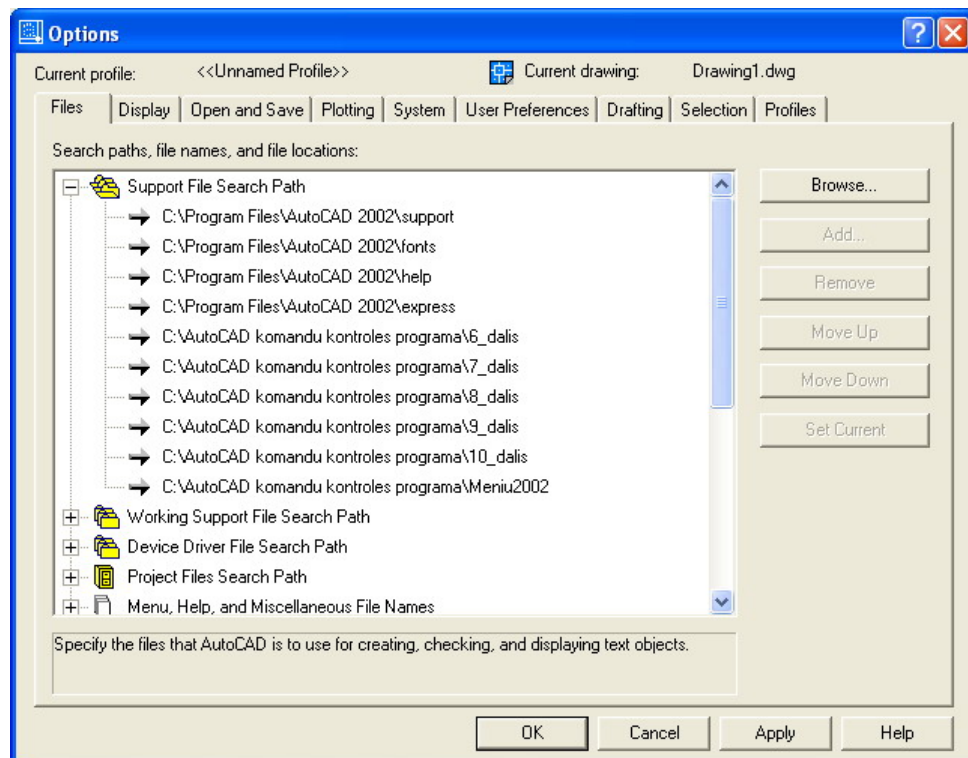
4.3 SISTEMOS INSTALIAVIMO DOKUMENTAS

Sukurtas produktas įrašytas į kompaktinį diską. Sukurtas produktas yra patalpintas į „AutoCAD komandu kontrolės programa“ katalogą. Į šį katalogą įeina tokie katalogai: „6_dalis“, „7_dalis“, „8_dalis“, „9_dalis“, „10_dalis“, „Meniu2000“, „Meniu2002“. Taip pat įeina labai reikalingos bylos: „Romancl.shx“ (skirtas atpažinti lietuviškus rašmenis) ir „Sablonas3D.dwt“ (skirtas braižyti 3D erdvėje trimačius objektus).

Visus katalogus, esantys kompaktiniame diske, perkopijuoti į kietojo disko vieną katalogą (pavyzdžiui, katalogo „AutoCAD komandu kontrolės programa“ (į kurią būtų įkelti tie katalogai) kelias būtų toks: *C:\AutoCAD komandu kontrolės programa*).

Užkrovus programą AutoCAD 2002 reikia nurodyti visų katalogų kelius (23 pav.): **Tools** → **Options**. Atsiras lentelė „**Options**“, pasirinkti kortelę „**Files**“. Šioje kortelėje išskleidžiame paieškos medį „**Support File Search Path**“. Vėliau spaudžiame mygtukus „**Add**“ ir „**Browse**“, nurodome katalogą (pavyzdžiui, „6_dalis“). Kiekvieną katalogą „prikabiname“ atskirai, t. y. teks atlikti keletą operacijų su mygtukais „**Add**“ ir „**Browse**“. Reikalingi katalogai:

- ✓ 6_dalis (AutoCAD įvadas);
- ✓ 7_dalis (Grafiniai primityvai ir jų braižymo komandos);
- ✓ 8_dalis (Brūkšniavimo ir matmenų komandos);
- ✓ 9_dalis (Redagavimo priemonės);
- ✓ Meniu2000 (menu byla, valdymo funkcijos, dialogo langas, ikonos skirtas programiniam paketui AutoCAD 2000);
- ✓ Meniu2002 (menu byla, valdymo funkcijos, dialogo langas, ikonos skirtas programiniam paketui AutoCAD 2002).

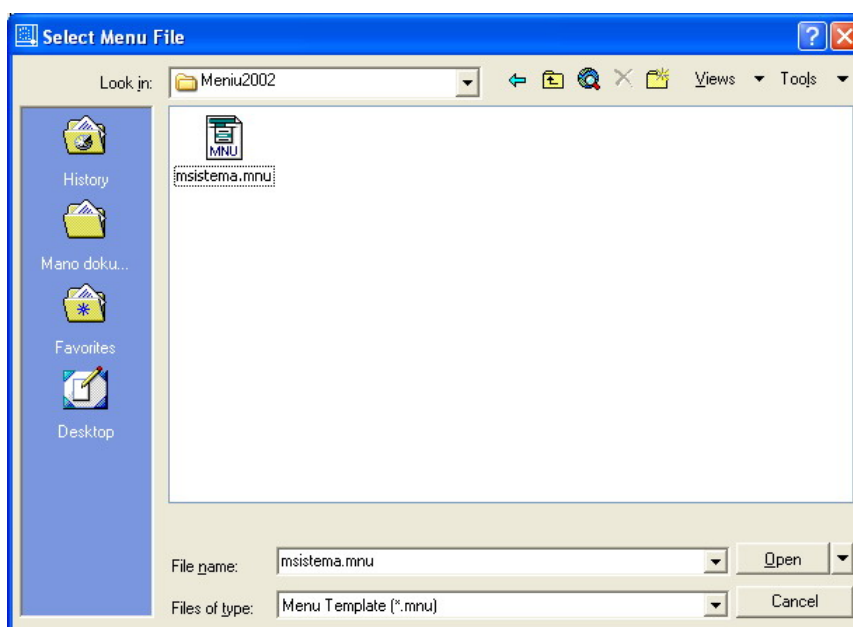


23 pav. Katalogų kelių nuorodos

Bylą „Romancl.shx“ perkelti į programinio paketo AutoCAD katalogą „**Support**“ (pavyzdžiui, *C:\Program Files\AutoCAD 2002\Support*).

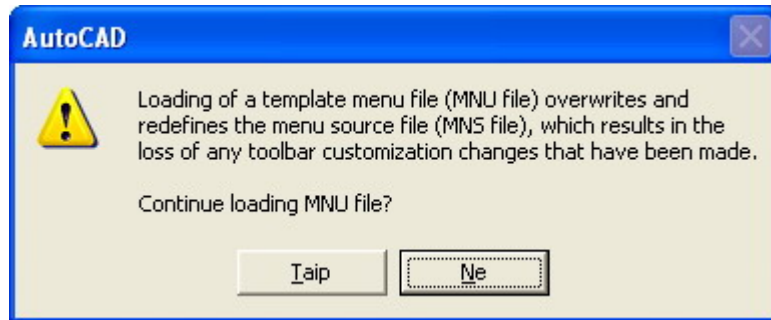
Bylą „Sablonas3D.dwt“ perkelti į programinio paketo AutoCAD katalogą „**Template**“ (pavyzdžiui, *C:\Program Files\AutoCAD 2002\Template*).

Dabar reikia užkrauti meniu: AutoCAD komandinėje eilutėje įrašyti žodį **menu** ir paspausti **Enter**. Pasirodžius bylos paieškos dialogui („**Select Menu File**“) pasirenkame bylos tipą („**Files of Type**“) „**Menu Template *.mnu**“, po to viršutiniame dialogo dalyje nurodome kelią į bylą „**msistema.mnu**“ (jis yra kataloguose „Menu200“ ir „Menu2002“; priklausomai nuo įdiegto programinio paketo išsirinkti vieną iš tų katalogų). Kelias į reikiamą bylą yra nurodytas 24 paveiksle.



24 pav. Bylos paieškos dialogo lentelė

Suradus reikiamą bylą pažymime ją ir nuspaužiamo mygtuką „**Open**“. Į pranešimą „**Continue loading MNU file?**“ (25 pav.) reikia atsakyti teigiamai, t. y. paspausti mygtuką „**Yes**“.



25 pav. Meniu užkrovimo pranešimas

Meniu bus užkrautas ir mokomoji sistema bus parengta darbui.

Jeigu bus braižoma dvimatėje erdvėje, tai maketą („**Template**“) imame „**acad.dwt**“, jei bus braižoma trimatėje erdvėje – „**Sablonas3d.dwt**“.

5. PRODUKTO KOKYBĖS ĮVERTINIMAS

Programinės įrangos kokybės įvertinimas nustato ar:

- ✓ sukurta programa atitinka vartotojo dokumentaciją;
- ✓ programa sukurta pagal standartus.

Sukurta programinė įranga buvo patikrinta projekto vadovo dr. A. Lenkevičiaus. Patikrinimo metu buvo patikrintos sukurtos programų sistemos funkcijos, tiriama, ar sukurta programa atitinka vartotojo reikalavimų specifikaciją. Buvo nustatyta, kad visi vartotojo reikalauti funkciniai bei nefunkciniai reikalavimai yra realizuoti. Prieita išvados, kad programa atitinka vartotojų reikalavimų specifikaciją.

Vartotojo reikalavimų specifikacijoje nebuvo reikalavimų, kad programa turi atitikti tam tikriems standartams. Todėl programinė įranga atitikimas standartams nebus nagrinėjamas.

Interviu su vadovu dr. A. Lenkevičiumi buvo vykdomi:

- ✓ apibrėžiant programos užduotį;
- ✓ nustatant vartotojo reikalavimus;
- ✓ sudarant sistemos architektūros specifikaciją;
- ✓ atliekant literatūros šaltinių analizę;
- ✓ tikrinant sukurtos programos funkcionalumą bei atitikimą vartotojo reikalavimams;
- ✓ sudarant programos vartotojo dokumentaciją.

Programos funkcionalumo tikrinimo metu atlikti tokie reikalingi patobulinimai: nustatyti brėžinių tikrinimo algoritmų kiekis ir tipai, patogus valdymas.

Produkto kokybę charakterizuoja programos funkcionalumas, patikimumas, panaudojamumas, efektyvumas, pernešamumas, atlikto darbo ir darbo našumo įvertinimas.

Funkcionalumas.

Tinkamumas. Sukurta programa atitinka vartotojo reikalavimų specifikaciją. Ši programa leidžia patikrinti nubraižyto brėžinio teisingumą.

Tikslumas. Sukurta programa atitinka sistemos struktūros modelį.

Sąveika su kitomis sistemomis. Programos kūrimo ir testavimo metu nebuvo pastebėta problemų programai sąveikaujant su kitomis programomis.

Saugumas. Ypatingų reikalavimų programų sistemos saugumui nebuvo keliami, kadangi sistemoje nebus saugomi duomenys, kurie turėtų būti neprieinami jokiems kitiems asmenims. Programa kurta mokymo tikslams.

Patikimumas.

Užbaigtumas. Programą galime laikyti neteisingai veikiančia, jei ji netikrina nubraižytą brėžinį. Apie brėžinio teisingumo nustatymą programa informuoja pranešimais. Nustatyta, kad programa atitinka reikalavimuose nurodytą patikimumą.

Kadangi programa kurta mokymo tikslais, ją galima tobulinti pridėdant daugiau funkcijų.

Panaudojamumas.

Suprantamumas. Sukurti vartotojo sąsajos. Sukurta programos vartotojo dokumentacija, kurioje aprašytos programos panaudojimo galimybės bei funkcijos.

Išmokstamumas. Yra sukurta programos vartotojo dokumentacija, kur aiškiai aprašytos visos programos funkcijos bei galimybės.

Vykdymo savybės. Sukurta patogi valdymo vartotojo sąsaja.

Patrauklumas. Programa skirta naudojimui įprastoje Windows aplinkoje, programos valdymas gana paprastas.

Efektyvumas.

Laiko parametrai. Programos veikimo greitis priklauso nuo techninės įrangos, kurioje yra įdiegta programa. Kuo geresnės techninės įrangos charakteristikos, tuo greitesnis programos veikimas. Minimalūs reikalavimai techninei įrangai pateikti vartotojo vadovo dokumentacijoje.

Resursų naudojimas. Kadangi yra tik tikrinamas vartotojo nubraižytas brėžinys, tai naudojama gana mažai operatyviosios atminties.

Palaikomumas.

Analizės savybės. Programos kodas yra prieinamas. Analizuojant programą, reikalingos AutoLISP programavimo kalbos žinios.

Pakeičiamumas. Kadangi programos kodas yra lengvai prieinamas, galima keisti programos funkcionalumą. Programa buvo kurta moksliniams tyrimams ir mokymo tikslams, taigi priėjimas prie programų sistemos kodo nėra apsaugotas.

Stabilumas. Programos stabilumas siejasi su operacinės sistemos Windows stabilumu. Kadangi nėra apsaugotas priėjimas prie programos kodo, pakeitus programos kodą, pakis ir programos funkcionalumas.

Pernešamumas.

Pritaikymas kitoje aplinkoje. Programa skirta tik Windows operacinei sistemai visoms versijoms.

Įdiegimo savybės. Programos įdiegimas aprašytas sistemos instaliavimo dokumente. Programos funkcionavimui turi būti instaliuotas programinis paketas AutoCAD 2002 (arba AutoCAD 2000).

Atlikto darbo kainos ir darbo našumo įvertinimas.

Programa buvo kurta mokymo tikslams ir moksliniams tyrimams, taigi reali programos kaina nėra nustatyta. Darbo našumas, įvertinant atlikto darbo apimtį ir naudą, yra pakankamai didelis.

6. IŠVADOS

1. Programa skirta Kauno technologijos universiteto studentams, kurių modulis „Kompiuterinė grafika“.
2. Programinio paketo AutoCAD 2002 mokomąją sistemą gali naudoti įstaigos ar pavieniai asmenys (dėstytojai, studentai ir t.t.).
3. Sukurta mokomoji sistema supažindina su AutoCAD 2002 grafine sistema, vartotojui suteikia galimybę atlikti pagrindines AutoCAD funkcijas, kurių atlikimas yra tikrinamas.
4. Sistemoje realizuotos užduotys sudarytos pagal A. Lenkevičiaus ir J. Maticko vadovėlį „Kompiuterinė grafika“ ir E. T. Romanučevos (Э. Т. Романычева) knyga „Компьютерная технология инженерной графики в среде AutoCAD 2000“.
5. Sukurtoje mokomojoje sistemoje yra pateikiamos užduotys **93** užduotys.
6. Mokomoji sistema nėra labai paprasta, nes reikia pagrindinių kompiuterinių įgūdžių.
7. Ši sistema suteikia galimybę sutaupyti laiką ir dirbti savarankiškai.

7. LITERATŪRA

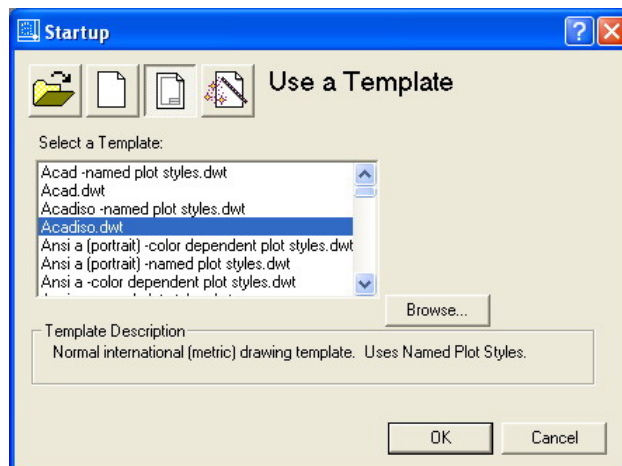
1. Čiuprinas V., Nenorta V. Grafinių vaizdų programavimas AutoLISP kalba. Kaunas Technologija, 1994.
2. Lenkevičius A., Matickas J. Kompiuterinė grafika. Kaunas: Technologija, 2000.
3. Ramašauskas O. AutoCAD'o pradžiamokslis. Vilnius: TEV, 1998. – 80 p.
4. Ramašauskas O. Automatizuoto projektavimo ir inžinerinės grafikos kompiuteriui laboratoriniai darbai. Klaipėda: Klaipėdos Universiteto leidykla, 2000.
5. Sinkevičius V. AutoCAD 2000 nuo nulio... Kaunas: Smaltija, 2002. – 736 p.
6. Smitas B. AutoCAD 14 žaliems. Kaunas: Smaltija, 1998. – 346 p.
7. Кудрявцев Е. М. AutoLISP. Основы программирования в AutoCAD 2000. Москва: ДМК прес, 2000. – 416 с.
8. Романычева Э.Т., Соколова Т.Ю. Компьютерная технология инженерной графики в среде AutoCAD 2000. Москва: ДМК прес, 2001. – 656 с.
9. AutoCAD 2000. AutoLISP Reference (elektroninė knyga): Autodesk, Inc, 1999. – 426 p.
10. AutoLISP Release 10. AutoLISP Programmer's Reference: Autodesk, Inc, 1998.
11. Allison T. D. AutoLISP Application Documentation (elektroninė knyga). – 19 p.
12. Finkelstein E. AutoCAD 2002 Bible (elektroninė knyga): Autodesk, Inc. – 1363 p.
13. Kompiuterinės grafikos paskaitos, pran. doc. Lenkevičius (2002 m.).
14. Programavimo inžinerijos paskaitos, pran. dr. Šeinauskas (2003 m.).
15. An Introduction and Tutorial for Common Lisp [interaktyvus]. Prieiga per internetą: <http://www.apl.jhu.edu/~hall/lisp.html>
16. AutoCAD users forum [interaktyvus]. Prieiga per internetą: <http://groups.msn.com/AutoCADusersforum/programming.msnw>
17. CAD Help by PG [interaktyvus]. Prieiga per internetą: <http://www.cadhlp.kulichki.com/>
18. CADchat Forums [interaktyvus]. Prieiga per internetą: <http://www.cadchat.com/forums/index.php?act=idx>
19. CAD/CAM/CAE Russian Community Forum [interaktyvus]. Prieiga per internetą: <http://sapr2000.ru/invision/index.php>
20. comp.cad.autocad AutoLISP FAQ [interaktyvus]. Prieiga per internetą: <http://faqs.org.ru/progr/graph/autolisp.htm>

21. АВТОЛИСП – язык графического программирования в системе AutoCAD [interaktyvus]. Prieiga per internetą:
http://kappasoft.narod.ru/info/acad/lisp/a_lisp.htm#3
22. Программирование: LISP [interaktyvus]. Prieiga per internetą:
<http://www.autocad.ru/cgi-bin/fl/board.cgi?p=23&sort=4a&page=1>
23. Программирование на AutoLISP [interaktyvus]. Prieiga per internetą:
http://aco.ifmo.ru/~nadinet/html/lectures/lect_lisp.html
24. Руководство пользователя по АВТОЛИСПу [interaktyvus]. Prieiga per internetą:
<http://www.codenet.ru/progr/alisp/>

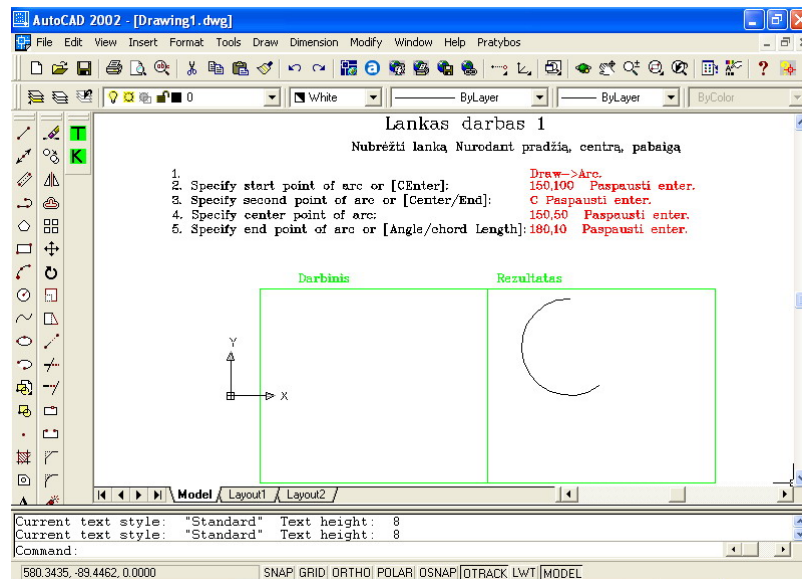
8. TERMINŲ IR SANTRUMPŲ ŽODYNAS

Pavadinimas	Paaiškinimas
AutoLISP	Programavimo kalba
CAD-sistemos	Automatizuoto projektavimo sistemos
CAI	Automatizuotos funkcijos (<i>Computer-aided instructions</i>)
CAL	Automatizuotas mokymas (<i>Computer-aided learning</i>)
CD-R	Perrašomas kompaktinis diskas
CD-ROM	Kompaktinių diskų grotuvas (<i>Compact Disc Read Only Memory</i>)
DCL	Dialogų valdymo kalba (<i>Dialog control language</i>)
DXF	AutoCAD duomenų formatas
HDD	Kompiuterio kietasis diskas (<i>Hard Disc Drive</i>)
IDE	Integruota kūrimo aplinka
POFOD	Nesėkmingo bandymo pasinaudoti sistema tikimybė (<i>Probability of failure on demand</i>)
RAM	Operatyvioji atmintis (<i>Random Access Memory</i>)
Toolbars	Įrankų juosta
VisualLISP	Programavimo kalba, aplinka
VLISP	Programavimo kalba, aplinka
*.dxf	AutoCAD duomenų formatas
*.dwg	AutoLISP dokumento formatas
*.lsp	AutoLISP ir VisualLISP dokumentų formatas

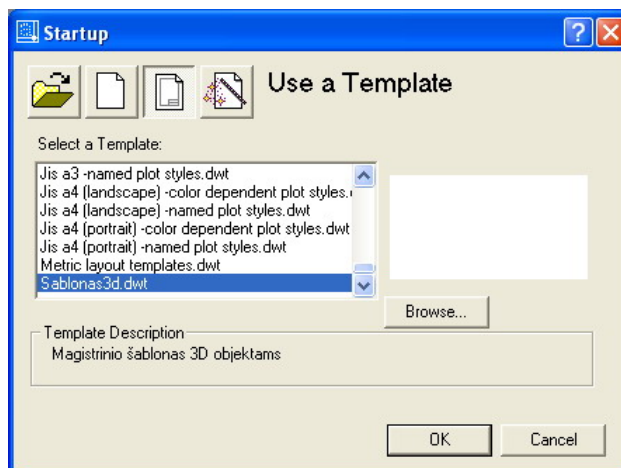
9. PRIEDAI



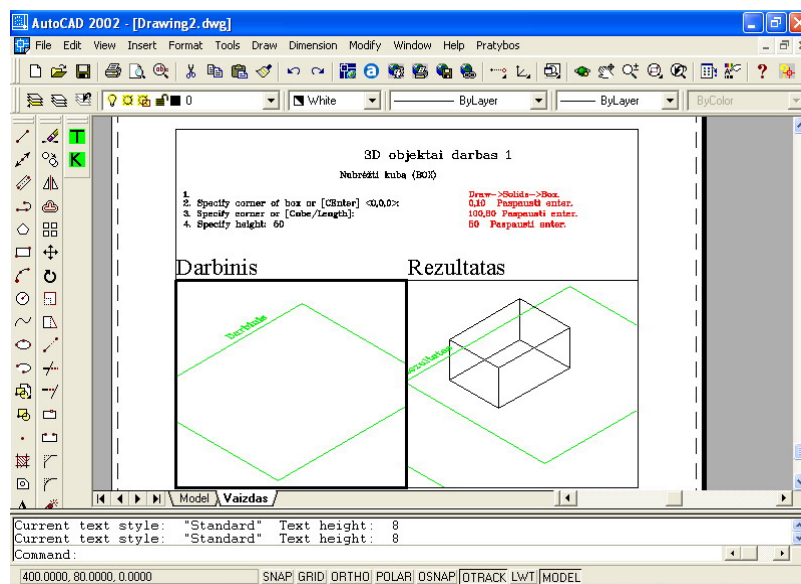
26 pav. Trafaretas „acadiso.dwt“ skirtas braižti 2D objektams



27 pav. 2D objektų braižymas



28 pav. Trafaletas „Sablonas3D.dwt“ skirtas braižti 3D objektams



29 pav. 3D objektų braižymas