

Latvijas Universitāte
Latvijas Universitātes Matemātikas un informātikas institūts

Māris Vītiņš

VISPĀRIZGLĪTOJOŠO VIDUSSKOLU INFORMĀTIKAS PAMATKURSA
DAŽU TĒMU PROGRAMMNODROŠINĀJUMS

Promocijas darbs (disertācija) datorzinātņu doktora grāda iegūšanai

Rīga 1993

Dis/8048

Promocijas darba (disertācijas) aizstāvēšana notiks LU Datorzinātņu nozares habilitācijas un promocijas padomes atklātā sēdē 1993.g. 7.oktobrī plkst.15⁰⁰ Latvijas Universitātes Matemātikas un informātikas institūta (LU MII) 413.telpā Raiņa bulvārī 29.

Recenzenti:

Dr.h.mat. R.Freivalds, Dr.ped. L.Burlakovs, fiz.-mat.zin.kand. G.Zlatkus (Lietuva).

Anotācija

Darbā ir apkopoti autora pētījumu un izstrāžu rezultāti par vispārizglītojošās vidusskolas informātikas pamatkursa saturu un dažu tēmu programmnodrošinājumu.

Darbs satur ievadu, četras nodaļas, nobeigumu, publicēto darbu sarakstu un pielikumu.

Teksta izklāsts ir uz 84 lappusēm.

Darba rezultāti ir aprobēti un tā vai citādi ieviesti.

Par darba tematiku ir 27 publikācijas. Ir referēts 4 starptautiskās un vairāk kā 20 lokāla rakstura konferencēs, semināros un sanāksmēs.

Ir publicētas grāmatas, izstrādātais programmnodrošinājums ir nodots nozares programmu un algoritmu fondā un tiek lietots skolās, Latvijas Universitātes Fizikas un matemātikas fakultātes studentiem tiek lasīts kurss "Informātikas pasniegšanas metodika".

Pielikumā ir pievienots akts par programmu kompleksa "Uzdevumi atkārtošanaī" ieviešanu Latvijas skolās.

Saturs

levads	7
1. Izglītības standarts informātikā un informātikas pamatkursa programma	9
1.1. Vidējās izglītības standarts informātikā	9
1.1.1. Informātikas kursa mācību mērķi	9
1.1.2. Informātikas kursa saturs	9
1.1.3. Zināšanas un prasmes, kas skolēnam jāapgūst informātikas pamatkursā	10
1.1.4. Skolēnu zināšanu un prasmju pārbaudes formas un tehnoloģija	11
1.2. Vispārīzglītojošās vidusskolas informātikas pamatkursa programma	12
1.2.1. Informācija. Informātika	12
1.2.2. Datora uzbūves un darbības principi	12
1.2.3. Datortehnikas attīstības vēsture un nozīme mūsdienu sabiedrībā	13
1.2.4. Algoritmi	13
1.2.5. Programmēšanas valoda BASIC	14
1.2.6. Informācijas apstrāde ar datoru	15
1.2.6.1. Lietišķo programmu paketes	15
1.2.6.2. Teksta redaktors	15
1.2.6.3. Grafiskais redaktors	15
1.2.6.4. Datu bāze	15
1.2.6.5. Elektroniskā tabula	16
2. Mācību materiāls par algoritmiem un programmēšanas valodām	17
3. Gausa metodes vispārināts algoritms	25
3.1. Risinājumu piemēri	25
3.2. Algoritma pieraksts algoritmiskajā valodā	29
3.3. Algoritma programmrealizācija MSX BASIC	38
4. Mācību datu bāze	43
4.1. Vispārējās ziņas	43
4.1.1. Datu bāzes vadības sistēmas darbošanās vide	43
4.1.2. Datu bāzes vadības sistēmas palaišana	43
4.1.3. Datu bāzes struktūra un parametri	43
4.1.4. Datu bāzes vadības sistēmas funkcijas	44

4.1.5. Lietotāja un sistēmas dialogs	46
4.1.6. Programmas	46
4.2. Datu bāzes vadības sistēmas funkciju apraksts	48
4.2.1. Datu bāzes ielāde	48
4.2.2. Datu bāzes aprakste	48
4.2.3. Datu bāzes aizpilde un labošana	50
4.2.4. Datu bāzes ierakstu apskate	55
4.2.5. Datu bāzes saglabāšana	56
4.2.6. Visu ierakstu dzēšana	57
4.2.7. Datu bāzes dzēšana	58
4.2.8. Statistiskā apstrāde	58
4.2.9. Ierakstu atlase (iezīmēšana)	59
4.2.10. Pārskata veidošana	61
4.2.10.1. Pārskata teksta faila sagatavošana izvadei uz drukas iekārtas	61
4.2.10.2. Pārskata izdruka	65
4.3. Datu bāzes vadības sistēmas lietošana	65
4.3.1. Uzdevuma "Ārstniecības augi" apraksts	65
4.3.1.1. Datu bāzes struktūras plānošana	66
4.3.1.2. Datu bāzes "AUGI" aprakste	66
4.3.1.3. Datu bāzes "AUGI" aizpilde	68
4.3.1.4. Datu bāzes apraksta mainīšana	70
4.3.1.5. Ziņu iegūšana no datu bāzes "AUGI"	70
4.3.1.6. Pārskatu sagatavošana, izmantojot datu bāzi "AUGI"	76
Nobeigums	79
Publicēto darbu saraksts	81
Pielikumā	85
Akts par programmu kompleksa "Uzdevumi atkārtšanai" ieviešanu Latvijas skolās	

Ievads

Kopš 1985./1986.mācību gada, pateicoties akadēmiķa Andreja Jeršova aktivitātēm, PSRS vidējās izglītības mācību iestādēs ir ieviests mācību kurss "Informātikas un skaitļošanas tehnikas pamati". Kursa uzdevums ir veidot jaunās paaudzes datorprasmi.

Kursa pirmā versija, ņemot vērā skolu slikto apgādi tajā laikā ar datoriem, bija orientēta uz "bezmašīnu" apguves variantu. Laika gaitā, uzlabojoties skolu apgādei ar datoriem, kurss pakāpeniski transformējās "mašīnvariantā". Tas kļūst apjomīgāks saturā un tam tiek atvēlēts vairāk stundu. Kursā būtisku vietu ieņem programmēšanas elementi un darbs ar lietišķo programmu paketēm.

Latvija bija pirmā no PSRS republikām, kas visās vidējās izglītības mācību iestādēs realizēja kursa mašīnvariantu, un šobrīd ir vienīgā no bijušām PSRS republikām, kur katrā vidējās izglītības mācību iestādē ir datoru klase. Tāpēc Latvijā mašīnvarianta kursa satura un tā atbalsta programmnodrošinājuma izstrādes problēma ir īpaši aktuāla un nozīmīga; vēl vairāk - tā no teorētisku pētījumu statusa ir pārgājusi neatliekami risināmu praktisku jautājumu kārtā.

Saskaņā ar pašlaik valstī spēkā esošo vispārīzglītojošo skolu mācību plānu informātika ir to izvēles priekšmetu grupā, kas skolai ir jāpiedāvā. Skolēni var apgūt informātiku kā pamatkursu (70 st.) vai kā profilkursu (210 st.).

Darbā ir apkopoti autora (ari kopā ar līdzautoriem) pētījumu un izstrāžu rezultāti par vispārīzglītojošās vidusskolas informātikas pamatkursa saturu un dažu tēmu programmnodrošinājumu.

Pirmā nodaļa satur vidējās izglītības standartu informātikā un tam atbilstošu vispārīzglītojošās vidusskolas informātikas pamatkursa programmu. Izglītības standarts ir reglamentējošs dokuments, kas nosaka priekšmeta mācīšanas mērķus un uzdevumus, saturu, sasniedzamo galarezultātu - zināšanas, prasmes un iemaņas, kā arī to pārbaudes formas un tehnoloģiju. Standarts ir kolektīvi radīts, Izglītības ministrijas Konsultatīvajā padomē apspriests un ministra apstiprināts dokuments. Autora piedāvātajai informātikas pamatkursa programmai, atbilstoši standartā formulētajam, ir rekomendējošs raksturs, taču, ņemot vērā to, ka tā tika izstrādāta, ieteikta un lietota vairākus gadus pirms standarta publicēšanas un nav pretrunā ar standarta prasībām, tad tā ir pamatkursa programma "de fakto". /Raksti: 4., 5., 11., 12. Tēzes: 3., 5., 7./

Otrās nodaļas saturs sniedz ieskatu autora sagatavotajā mācību materiālā par algoritmu un programmēšanas valodu apguvi. Ir izstrādāti risinājumu algoritmi gandrīz visiem uzdevumiem, kuri ietverti Latvijas skolās biežāk lietotajās mācību grāmatās, kā arī 35 FOKAL programmu komplekss "Uzdevumi atkārtošanai", kas realizē A.Jeršova un V.Monahova redakcijā izdotās "Informātikas un skaitļošanas tehnikas pamatu" mācību grāmatas nodaļas "Uzdevumi atkārtošanai" uzdevumu risinājumus. /Grāmatas: 2., 3., 4., 5., 6. Raksti: 6., 8., 9., 13. Tēzes: 4./

Trešajā nodaļā ir ievietots materiāls, kas noder, iepazīstinot skolēnus ar vienkāršām skaitliskajām metodēm. Ir izklāstīta Gausa metode (pakāpeniskās izslēgšanas metode) lineāru vienādojumu sistēmu risināšanai un dots autora izstrādāts Gausa metodes vispārināts algoritms (arī programmrealizācija MSX BASIC), kas ir vienlīdz rezultatīvi lietojams gan sistēmām ar vienu atrisinājumu, gan sistēmām bez atrisinājuma, kā arī sistēmām ar bezgala daudziem atrisinājumiem. /Grāmatas: 1. Raksti: 1., 2., 3., 7. Tēzes: 1., 2./

Ceturtajā nodaļā ir aprakstīta īpaši mācību mērķiem līdzautoru izstrādāta datu bāzes vadības sistēma, kas darbojas KYBT-86 (ДБК, БК-0010 (BASIC)) vidē, un tās lietošana. /Grāmatas: 7. Raksti: 10. Tēzes: 6./

Darba rezultāti ir aprobēti un tā vai citādi ieviesti.

Par darba tematiku ir 27 publikācijas. Ir referēts 4 starptautiskās un vairāk kā 20 lokāla rakstura konferencēs, semināros un sanāksmēs.

Ir publicētas grāmatas, izstrādātais programmnodrošinājums ir nodots nozares programmu un algoritmu fondā un tiek lietots skolās, Latvijas Universitātes Fizikas un matemātikas fakultātes studentiem tiek lasīts kurss "Informātikas pasniegšanas metodika".

Pielikumā ir pievienots akts par programmu kompleksa "Uzdevumi atkārtošanai" ieviešanu Latvijas skolās.

Autors ir Izglītības, kultūras un zinātnes ministrijas Konsultatīvās padomes informātikā loceklis.

Autors ir Latvijas vidējo mācību iestāžu informātikas olimpiāžu (1991.g., 1992.g., 1993.g.) rīcības komitejas priekšsēdētājs un Latvijas valstsvienības vadītājs starptautiskajās skolēnu informātikas olimpiādēs (1992.g. Bonnā, 1993.g. Buenosairesā).

1. Izglītības standarts informātikā un informātikas pamatkursa programma

1.1. Vidējās izglītības standarts informātikā

Vidējās vispārējās izglītības standarts informātikā ir reglamentējošs dokuments, kas nosaka priekšmeta pamatkursa mācīšanas mērķus un uzdevumus, saturu, sasniedzamo galarezultātu - zināšanas, prasmes un iemaņas, kā arī to pārbaudes formas un tehnoloģiju.

Vidējās vispārējās izglītības standarts informātikā jāsāk ieviest no 1993./94. mācību gada.

Katram skolotājam ir tiesības un pienākums organizēt mācību procesu patstāvīgi, t.i. strādāt pēc paša veidotām mācību programmām, pēc saviem ieskatiem izvēloties izglītības standartā paredzētā satura apguves secību un apguvei nepieciešamo laiku. Tai pašā laikā skolotājs ir atbildīgs par to, lai katram skolēnam būtu nodrošināta iespēja iegūt izglītības standartam atbilstošu izglītību.

Izglītības ministrijas piedāvātajām informātikas mācību programmām ir rekomendējošs raksturs.

1.1.1. Informātikas kursa mācību mērķi

Iepazīt informācijas datorapstrādes tehnoloģiju, gūt priekšstatu par svarīgākajiem informātikas jēdzieniem, datoru uzbūves un darbības principiem, iemācīties strādāt ar dažādām programmām (teksta redaktoru, grafisko redaktoru, datu bāzi, elektronisko tabulu), gūt priekšstatu par datoru izmantošanas iespējām dažādās jomās.

1.1.2. Informātikas kursa saturs

Informācija un informātiskie procesi mūsdienu sabiedrībā:

- informācijas iegūšana, pārveidošana, pārsūtīšana, uzglabāšana;
- datori informācijas apstrādē.

Priekšstats par algoritmiem un programmēšanu:

- algoritms kā darbības apraksts;

- algoritmu pierakstu veidi;
- jēdziens par programmēšanu;
- programma kā algoritma pieraksts izpildītāja komandu valodā.

Tekstu redaktori tekstu sagatavošanai un apstrādei datorā.

Grafiskie redaktori attēlu veidošanai.

Datu bāzes informācijas uzkrāšanai:

- datu bāzes vadības sistēmas funkcijas;
- informācijas ievadišana un meklēšana datu bāzē.

Elektroniskās tabulas un lietišķā grafika.

Datoru lietošanas ekonomiskie, sociālie un ētiskie aspekti:

- datoru izmantošana izglītībā, medicīnā u.c.;
- informācijas un komunikāciju tīkli;
- elektroniskais pasts;
- ekspertu sistēmas;
- autortiesību ievērošana.

1.1.3. Zināšanas un prasmes, kas skolēnam jāapgūst informātikas pamatkursā

Skolēnam jāzina:

- datora pamatbloki, to funkcijas;
- datora programmvadības princips;
- operāciju sistēmas jēdziens, tās funkcijas;
- algoritma un programmas jēdziens;
- galvenie teksta un zīmējumu apstrādes etapi ar datoru (sagatavošana, rediģēšana, noformēšana, izdrukāšana, uzglabāšana), tekstu analīzes iespējas datorā;
- datu bāzes pamatjēdzieni, datu bāzes vadības sistēmas pamatfunkcijas, pieprasījumu valodas pamatkonstrukcijas, datu bāžu lietojumi izglītībā, kultūrā, sadzīvē;
- elektroniskās tabulas pamatjēdzieni, lietišķā grafika un tās lietošanas iespējas;

- informācijas tīklu nozīme, lietojumi dažādās nozarēs un attīstības perspektīvas.

Skolēnam jāprot:

- strādāt ar datoru informātikas kabineta datoru tīklā (sagatavot darbam datoru, strādāt ar datora klaviatūru);
- strādāt ar teksta redaktora un grafiskā redaktora programmām: ielādēt datora atmiņā tekstu un zīmējumu no ārējās atmiņas, saglabāt gatavu tekstu un zīmējumu datora ārējā atmiņā, izdarīt teksta un zīmējuma sagatavošanas, labošanas un rediģēšanas operācijas, izdrukāt sagatavoto tekstu un zīmējumu;
- izveidot, aizpildīt un labot vienkāršu datu bāzi, strādāt ar sagatavotu datu bāzi apskates režīmā, sastādīt pieprasījumu valodā atlasēs nosacījumus, izveidot un izdrukāt pārskatu;
- risināt tipveida uzdevumus ar elektronisko tabulu: ievadīt elektroniskajā tabulā skaitļus, tekstus, formulas, atbilstoši izstrādātam algoritmam, iegūt rezultātus uz monitora un izdrukāt, attēlot skaitlisku informāciju ar lietisķās grafikas līdzekļu palīdzību.

1.1.4. Skolēnu zināšanu un prasmju pārbaudes formas un tehnoloģija

Pamatkursa apguves līmeņa pārbaudes formas:

- eksāmens;
- noslēguma ieskaite.

Eksāmenu kārtību katrā mācību gadā nosaka izglītības ministrs.

Jautājumus un uzdevumus rakstveida eksāmenam un biļetes mutvārdu eksāmenam apstiprina izglītības ministrs.

Sagatavojot tematisko ieskaīšu vai noslēguma ieskaīšu saturu, skolotājam tajās jāietver vairāki jautājumi un uzdevumi, lai skolēnam būtu iespējams pilnībā parādīt iegūtās zināšanas un prasmes.

Skolēnu prasmju pārbaude:

- prasmi strādāt ar klaviatūru un ievadīt vienkāršu informāciju pārbauda ar testu programmām;

- prasmi strādāt ar teksta redaktoru un grafisko redaktoru pārbauda ar iepriekš sagatavotiem vingrinājumiem par atsevišķām programmu komandām (operācijām), ar kompleksiem uzdevumiem par programmām kopumā (sagatavot, rediģēt, noformēt un izdrukāt tekstu vai zīmējumu);
- prasmi strādāt ar datu bāzēm pārbauda ar uzdevumiem par atlases nosacījumu sastādīšanu pieprasījumu valodā, ar uzdevumiem par darbu datu bāzes apskates režīmā, par vienkāršu bāzu veidošanu, aizpildīšanu un labošanu, par pārskata veidošanu un izdrukāšanu;
- prasmi strādāt ar elektroniskajām tabulām pārbauda ar uzdevumiem par aizpildītu tabulu lietošanu, par elektronisko tabulu sagatavošanu (skaitļu un tekstu ievadīšana, formulu lietošana u.tml.), par elektroniskās tabulas datu grafisku attēlošanu, elektroniskās tabulas satura izdrukāšanu.

1.2. Vispārīzglītojošās vidusskolas informātikas pamatkursa programma (10.klase; 70 st.)

1.2.1. Informācija. Informātika (2 st.)

Skolēnam jāprot:

- minēt piemērus par informācijas uztveršanu, uzglabāšanu, pārraidīšanu un apstrādi;
- raksturot informātiku kā zinātni;
- pastāstīt par informātikas un datortehnikas savstarpējo saikni.

1.2.2. Datora uzbūves un darbības principi (2 st.)

Datora galvenās sastāvdaļas, to funkcijas un savstarpējā saikne. Drošības tehnikas noteikumi darbā ar datoru. Datora programmvadības jēdziens. Operāciju sistēma. Dators skolā, lokālais tīkls.

Skolēnam jāzina:

- datora galvenās sastāvdaļas un to funkcijas;
- drošības tehnikas noteikumi darbā ar datoru;
- programmvadības jēdziens.

Skolēnam jāprot:

- lietot datora klaviatūru.

Praktiskie darbi:

- darbs ar klaviatūras trenāžieri.

1.2.3. Datortehnikas attīstības vēsture un nozīme mūsdienu sabiedrībā (4 st.)

Datortehnikas attīstības vēsture. Datoru paaudzes, elementu bāzes un programmnodrošinājuma attīstība. Dators kā skaitļošanas ierīce, modelēšanas instruments, informācijas uzglabāšanas un sistematizācijas līdzeklis, vadības ierīce.

Ražošanas automatizācija, inženieraprēķini un ekonomiskie aprēķini, matemātiskā modelēšana, automatizētās vadības sistēmas, informatīvās uzziņu sistēmas, automatizētās projektēšanas sistēmas.

Dators tautas saimniecībā, zinātnē, medicīnā, apkārtējās vides aizsardzībā, izglītībā, kultūrā, sadzīvē.

Skolēnam jāzina:

- datoru un programmnodrošinājuma attīstības galvenie etapi.

Skolēnam jāprot:

- minēt datoru lietošanas piemērus;
- izskaidrot datortehnikas nozīmi mūsdienu sabiedrības attīstībā;
- darboties ar vienkārša dialoga programmām.

Praktiskie darbi:

- darbs ar vienkārša dialoga programmām.

1.2.4. Algoritmi (12 st.)

Algoritma jēdziens. Algoritmu īpašības. Algoritma izpildītājs. Algoritma pieraksta veidi.

Lineāri algoritmi. Algoritmi ar sazarošanos. Algoritmi ar atkārtošanos.
Palīgalgoritmi.

Lielumi. Lielumu nosaukums un vērtība. Mainīgi lielumi un konstanti lielumi.
Lielumu tabulas.

Piešķires komanda.

Argumenti un rezultāti.

Skolēnam jāzina:

- algoritma jēdziens, tā īpašības;
- algoritma pieraksta veidi (vārdiskais apraksts, blokshēmas, algoritmiskā valoda, tabulas).

Skolēnam jāprot:

- izstrādāt un pierakstīt vienkāršus algoritmus;
- izpildīt vienkāršus algoritmus;
- strādāt ar izpildītāju.

Praktiskie darbi:

- vienkāršu algoritmu izstrāde;
- vienkāršu algoritmu izpilde;
- darbs ar izpildītāju.

1.2.5. Programmēšanas valoda BASIC (16 st.)

Programmēšanas valoda kā algoritmu pieraksta veids datoram.

Alfabēts. Skaitļu pieraksts. Mainīgie un konstantes. Piešķires komandas.
Skaitliskas izteiksmes. Standartfunkcijas. Darbības ar simboliskajiem
mainīgajiem. Datu ievade, labošana, izpilde un saglabāšana. Komentāri.

Programmas rindu izpildes secības maiņas komandas. Sazarojuma komanda.
Atkārtojuma komanda. Programmā ielasīto datu "ielases" komanda.

Apakšprogrammas. Programmētāja definētas funkcijas.

Grafiskie līdzekļi.

Skolēnam jāzina:

- datora kā algoritmu izpildītāja iespējas un īpašības.

Skolēnam jāprot:

- sastādīt vienkāršas programmas.

Praktiskie darbi:

- vienkāršu programmu sastādīšana.

1.2.6. Informācijas apstrāde ar datoru (34 st.)

1.2.6.1. Lietišķo programmu paketes (4 st.)

Informācijas apstrādes (uzdevumu risināšanas) tehnoloģijas jēdziens.

Uzdevuma risināšanas etapi. Uzdevuma nostādne, nosacījumu analīze un datoru tehnikas pielietojuma iespēju noteikšana.

Skaitlisko metožu, lineārās programmēšanas, statistiskās analīzes paketes, matemātiskās modelēšanas programmu paketes. Pedagoģiskie programmlīdzekļi. Programmdokumentācija.

1.2.6.2. Teksta redaktors (8 st.)

Teksta apstrāde. Teksta struktūra. Teksta redaktora pamatjēdzieni un pamatfunkcijas. Teksta izdruka. Teksta redaktors sadzīvē, lietvedībā, izdevējdarbībā.

1.2.6.3. Grafiskais redaktors (6 st.)

Attēlu konstruēšana un pārveidošana. Grafiskā redaktora pamatjēdzieni un pamatfunkcijas. Attēlu izdruka. Grafiskais redaktors sadzīvē, projektēšanā.

1.2.6.4. Datu bāze (8 st.)

Informācijas masivi. Operācijas ar datiem. Datu bāzes pamatjēdzieni. Datu struktūras. Datu bāzes vadības sistēmas. Pieprasījumu valoda. Datu izdruka. Datu bāzes sadzīvē, lietvedībā, zinātnē.

1.2.6.5. Elektroniskā tabula (8 st.)

Tabulas struktūra un elementu saturs. Elektroniskās tabulas pamatjēdzieni un pamatfunkcijas. Rezultātu izdruka. Tipveida uzdevumi, kas risināmi, izmantojot elektronisko tabulu.

Skolēnam jāzina:

- informācijas apstrādes uzdevumu risināšanas tehnoloģijas etapi;
- lietišķo programmu pakešu veidi.

Skolēnam jāprot:

- lietot programmdokumentāciju;
- izmantot lietišķo programmu paketes skolas kursa uzdevumu risināšanā;
- lietot teksta redaktoru;
- lietot grafisko redaktoru;
- lietot datu bāzes vadības sistēmu;
- lietot elektronisko tabulu.

Praktiskie darbi:

- skolas kursa uzdevumu risināšana, izmantojot lietišķo programmu paketes;
- darbs ar skolas kursa mācību programmām;
- darbs ar teksta redaktoru;
- darbs ar grafisko redaktoru;
- darbs ar datu bāzi;
- darbs ar elektronisko tabulu.

2. Mācību materiāls par algoritmiem un programmēšanas valodām

Viens no informātikas pamatkursa uzdevumiem ir iepazīstināt skolēnus ar vienkāršu algoritmu un to realizējošu programmu izstrādi.

Informātikas kursa pasniedzēji parasti nav programmētāji. Viņi lielākoties ir fizikas un matemātikas skolotāji, un tāpēc viņiem ir jāpapildinās algoritmizēšanas un programmēšanas jomā.

Autors ar kolēģiem nonāca pie atziņas, ka vislietderīgāk būtu izstrādāt un publicēt risinājumu algoritmus un programmrealizācijas tiem uzdevumiem, kuri ietverti Latvijas skolās biežāk lietotajās mācību grāmatās.

Un tā tapa:

1. grāmata "Изучение основ информатики и вычислительной техники" (līdzautori R.Freivalds, A.Jeršovs, V.Monahovs un citi). Darbs krievu valodā nāca klajā Maskavas izdevniecībā "Просвещение", tas ir tulkots un izdots vēl vismaz sešās valodās un tiek lietots visā bijušās PSRS teritorijā;
2. grāmata "Изучение курса "Основы информатики и вычислительной техники"" (līdzautori R.Freivalds, I.Opmane, S.Pavlovs);
3. grāmata "Решение задачи на повторение" (līdzautors P.Čikusts);
4. grāmata "Algoritmu sastādīšana un programmēšana" (līdzautori A.Kalējs, I.Opmane) /izdošanā/;
5. programmu komplekss "Задачи на повторение".

Īpaši atzīmējams ir pēdējais no uzskaitītajiem darbiem - programmu komplekss "Задачи на повторение". Komplekss satur 35 FOKAL programmas, kas realizē A.Jeršova un V.Monahova redakcijā izdotās "Informātikas un skaitļošanas tehnikas pamatu" mācību grāmatas nodaļas "Uzdevumi atkārtšanai" uzdevumu risinājumus.

Kompleksa programmām ir ērti datu ievada, kontroles un rediģēšanas līdzekļi. Darba rezultāti tiek attēloti uzskatāmā veidā. Programmu teksti ir strukturēti, komentēti.

Ilustrācijai aplūkosim sekojošu uzdevumu:

uzzināt tabulas ves tab A[1:100] nenulles elementu skaitu.

Algoritmiskajā valodā viens no risinājuma pierakstiem var būt šāds:

algoritms 1.VINGRINĀJUMS (arg ves tab A[1:100], rez ves SKAITS)

sākums ves I

SKAITS:=0

priekš I no 1 līdz 100

cikls

ja A[I]=0

tad

SKAITS:=SKAITS+1

viss

stāt

beigas

Programmēšanas valodā FOKAL risinājumu var pierakstīt:

40.01 S KO=0;F I=1,100;D 41

40.03 R

41.01 I (A(I)) 41.03,41.05,41.03

41.03 S KO=KO+1

41.05 R

Jāpiebilst, ka programmā lieluma SKAITS vietā ir mainīgais KO.

Lai programma būtu darbaspējīga, tā jāpapildina ar tabulas A elementu vērtību ievada un nenulles elementu skaita izvada komandām. Viens no variantiem:

10.01 F I=1,100;D 11

10.03 D 40

10.05 T "SKAITS",KO;R

11.01 A "A(I)",A(I);R

Ir iegūta programma, kas lietojama vienīgi 100 elementu tabulu apstrādei, turklāt netiek kontrolēti ievaddati un nav to rediģēšanas iespējas. Vēl kā programmas trūkums atzīmējams ievaddatu un rezultāta nepārskatāmais attēlojums, kā arī lietotājinstrukciju un komentāru trūkums pašā programmā.

Tālāk seko tādas programmas teksts, kurā ir novērsti augstākminētie trūkumi un ir izturēts labs programmēšanas stils:

10.01 С ЛатвГУ им.П.Стучки

10.03 С Вычислительный центр

10.05 С Лаборатория проблем школьной информатики

10.07 С ОСНОВЫ ИНФОРМАТИКИ И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

10.09 С ЧАСТЬ ВТОРАЯ

10.11 С УПРАЖНЕНИЯ ДЛЯ ПОВТОРЕНИЯ

10.13 С УПРАЖНЕНИЕ 1

10.15 С РИГА 1986

10.17 С М.В.Витиньш

10.19 D 11;D 12;D 13;D 20;D 30;D 40;D 50;R

11.01 С СТИРАНИЕ КУРСОРА

11.03 С УСТАНОВКА 32 СИМВОЛА В СТРОКЕ

11.05 С СТИРАНИЕ СЛУЖЕБНОЙ СТРОКИ

11.07 I (FX(0,56,255)) 99.99,11.09,11.11

11.09 X FCHR(154)

11.11 I (FX(0,40,255)) 99.99,11.13,11.15

11.13 X FCHR(155)

11.15 X FCHR(148,158,145)

11.17 R

12.01 С ВВОД ЗАГОЛОВКА

12.03 X FCHR(12)

12.05 X FK(6,1);T "ЛатвГУ им.П.Стучки"

12.07 X FK(5,2);T "Вычислительный центр"

12.09 X FK(1,3);T "Лаборатория проблем школьной"

12.11 X FK(9,4);T "информатики"

12.13 X FK(6,7);T "ОСНОВЫ ИНФОРМАТИКИ"

12.15 X FK(3,8);T "И ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ"

12.17 X FK(9,11);T "ЧАСТЬ ВТОРАЯ"

12.19 X FK(2,14);T "УПРАЖНЕНИЯ ДЛЯ ПОВТОРЕНИЯ"

12.21 X FK(9,17);T "УПРАЖНЕНИЕ 1"

12.23 X FK(10,20);T "РИГА 1986"

12.25 X FK(9,22); T "М.В.Витиньш";S T=30;D 92;R

13.01 С ВЫВОД ИНСТУКЦИЙ

13.03 X FCHR(12)

13.05 X FK(0,3); T "Программа находит количество"

13.07 T "ненулевых элементов в таблице"

13.09 T "целых чисел A[1:N]."; S T=8; D 92

13.11 X FK(0,7); T "Можно задать количество элемен-"

13.13 T "тов N (от 1 до 99) и значения"

13.15 T "элементов (от -99 до 99)."; S T=7; D 92

13.17 X FK(0,11); T "Ошибочно набранное удаляется"

13.19 T "клавишей удаления последнего"

13.21 T "символа в строке."; S T=6; D 92

13.23 X FK(0,15); T "Ввод набранного осуществляется"

13.25 T "клавишей ввода."; S T=4; D 92

13.27 X FK(5,22); T "Нажмите клавишу ввода!"

13.29 S P=FCHR(-1)

13.31 I (P-13) 13.29,13.33,13.29

13.33 R

20.01 С ВВОД КОЛИЧЕСТВА ЭЛЕМЕНТОВ

20.03 X FCHR(12)

20.05 X FK(0,20); T "Количество элементов в таблице",!,",."

20.07 X FCHR(154); D 91; X FCHR(154)

20.09 I (KC) 99.99,20.11,20.17

20.11 T !, "Число!"

20.13 S T=3; D 92; X FK(1,21); X FCHR(153); X FK(0,22); X FCHR(153);

20.15 X FK(1,21); G 20.07

20.17 I (CI-FITR(CI)) 20.19,20.21,20.19

20.19 T !, "Целое число!"; G 20.13

20.21 I (CI) 20.23,20.23,20.25

20.23 T !, "Не меньше !"; G 20.13

20.25 I (CI-99) 20.29,20.29,20.27

20.27 T !, "Не больше 99!"; G 20.13

20.29 X FCHR(12)

20.31 T "Таблица A[1:."; X FK(10,0)

20.33 I (CI-9) 20.35,20.35,20.37

20.35 T %1,CI; X FK(10,0); T "1:."; X FK(13,0); T "T"; R

20.37 T %2,CI; X FK(10,0); T "1:."; X FK(14,0); T "T"; R

30.01 С ВВОД ЗНАЧЕНИЙ ЭЛЕМЕНТОВ

30.03 F J=1,L;D 31

30.05 X FK(0,20);X FCHR(153);X FK(0,21),X FCHR(153)

30.07 X FK(0,2+FITR((J-1)/8));R

31.01 С ВВОД ЗНАЧЕНИЯ J-ОГО ЭЛЕМЕНТА

31.03 X FK(0,20);T "A[";X FK(0,20)

31.05 I (J-9) 31.07,31.07,31.09

31.07 T %1,J;X FK(0,20);T "A[";X FK(3,20);T "]"",!,"";G 31.11

31.09 T %2,J;X FK(0,20);T "A[";X FK(4,20);T "]"",!,"";G 31.11

31.11 X FCHR(154);D 91;X FCHR(154)

31.13 I (KC) 99.99,31.15,31.21

31.15 T !,"Число!"

31.17 S T=3;D 92;X FK(1,21);X FCHR(153);X FK(0,22);X FCHR(153)

31.19 X FK(1,21);G 31.11

31.21 I (CI-FITR(CI)) 31.23,31.25,31.23

31.23 T !,"Целое число!";G 31.17

31.25 I (FABS(CI)-99) 31.33,31.33,31.27

31.27 I (CI) 31.29,99.99,31.31

31.29 T !,"Не меньше -99!";G 31.17

31.31 T !,"Не больше 99!";G 31.17

31.33 S A(J)=CI;X FK(1,21);X FCHR(153)

31.35 X FK((J-FITR((J-1)/8)*8-1)*4,1+FITR((J-1)/8));T %2,A(J);R

40.01 С НАХОЖДЕНИЕ КОЛИЧЕСТВА НЕНУЛЕВЫХ ЭЛЕМЕНТОВ

40.03 S KO=0;F I=1,L;D 41

40.05 R

41.01 I (A(I)) 41.03,41.05,41.03

41.03 S KO=KO+1

41.05 R

50.01 С ВЫВОД КОЛИЧЕСТВА НЕНУЛЕВЫХ ЭЛЕМЕНТОВ

50.03 T !,"Количество ненулевых элементов:";!,%2,KO

50.05 X FK(0,23);X FCHR(154);R

91.01 С ВВОД ЧИСЛА

91.03 S KP=0;S KC=0;S PT=0;S CD=0;S CP=0

91.05 S P=FCHR(-1)

91.07 I (P-13) 91.05,91.09,91.29

91.09 I (KC) 99.99,91.11,91.13

91.11 R

91.13 S CI=0

91.15 I (CD) 99.99,91.17,91.21

91.17 S MN=.1;F I=1,CP;S CI=CI+(C(CD+I)-48)*MN;S MN=MN*.1

91.19 S CI=CI*ZN;R

91.21 S MN=1;S I=CD

91.23 I (I-1) 91.27,91.25,91.25

91.25 S CI=CI+(C(I)-48)*MN;S MN=MN*10;S I=I-1;G 91.23

91.27 I (CP) 99.99,91.19,91.17

91.29 I (P-24) 91.05,91.31,91.49

91.31 I (KP) 99.99,91.05,91.33

91.33 I (CP) 99.99,91.35,91.47

91.35 I (PT) 99.99,91.37,91.45

91.37 I (CD) 99.99,91.39,91.43

91.39 S KP=KP-1

91.41 X FCHR(P);G 91.05

91.43 S CD=CD-1;S KC=KC-1;G 91.39

91.45 S PT=0;G 91.39

91.47 S CP=CP-1;S KC=KC-1;G 91.39

91.49 I (P-43) 91.05,91.51,91.57

91.51 I (KP) 99.99,91.53,91.05

91.53 S ZN=1

91.55 S KP=KP-1;G 91.41

91.57 I (P-45) 91.05,91.59,91.63

91.59 I (KP) 99.99,91.61,91.05

91.61 S ZN=-1;G 91.55

91.63 I (P-46) 91.05,91.65,91.75

91.65 I (KP-7) 91.67,91.05,91.05

91.67 I (KP) 99.99,91.69,91.71

91.69 S ZN=1;S PT=1;G 91.55

91.71 I (PT) 99.99,91.73,91.05

91.73 S PT=KP+1;G 91.55

91.75 I (P-48) 91.05,91.79,91.77

91.77 I (P-57) 91.79, 91.79,91.05

91.79 I (KC-5) 91.81,91.05,91.05

91.81 I (KP) 99.99,91.83,91.87

91.83 S ZN=1;KC=1;S CD=1

91.85 S C(KC)=P;G 91.55

91.87 S KC=KC+1

91.89 I (PT) 99.99,91.91,91.93

91.91 S CD=CD+1;G 91.85

91.93 S CP=CP+1;G 91.85

92.01 С ОРГАНИЗАЦИЯ ПАУЗЫ

92.03 X FX(-1,177660,64);S X=10*T

92.05 S X=X-1

92.07 I (FX(0,177660,128)) 9.99,92.09,92.11

92.09 I (X) 92.13,92.13,92.05

92.11 X FX(1,177662)

92.13 X FX(-1,177660,0);R

3. Gausa metodes vispārināts algoritms

Informātikas pamatkursa programma paredz iepazīstināt skolēnus ar vienkāršām skaitliskajām metodēm. Viena no tādām ir Gausa metode (pakāpeniskās izslēgšanas metode) lineāru vienādojumu sistēmu risināšanai. Tās ideja ir viegli uztverama, un programmrealizācija ir izprotama vidusskolēnam.

Gausa metodes būtība ir pārveidot doto lineāru vienādojumu sistēmu par tai ekvivalentu trīsstūrveida sistēmu, un tad secīgi, sākot no pēdējiem vienādojumiem, iegūt mainīgo vērtības.

3.1. Risinājumu piemēri

Pirmajos trīs piemēros ir risinātas sistēmas, kurām, attiecīgi, ir viens atrisinājums, nav atrisinājuma, ir bezgala daudz atrisinājumu.

Piemērs # 1. Atrisināt vienādojumu sistēmu:

$$(S) = \begin{cases} x - 4y - 7z = 2 \\ 3x - 8y - 9z = -6 \\ 4x - 5y - 3z = -1 \end{cases}$$

Pārveido sistēmu (S) par tai ekvivalentu sistēmu (S1), kas otrajā un trešajā vienādojumā nesatur mainīgo x. Lai to panāktu, atņem pirmo vienādojumu, pareizinātu ar 3, no otrā un, savukārt, pareizinātu ar 4, no trešā vienādojuma.

$$(S1) = \begin{cases} x - 4y - 7z = 2 \\ 4y + 12z = -12 \\ 11y + 25z = -9 \end{cases}$$

Dala pārveidotās sistēmas (S1) otrā vienādojuma abas puses ar 4 un iegūst vienādojumu ar koeficientu 1 pie mainīgā y:

$$y + 3z = -2$$

Pieskaita iegūto vienādojumu, pareizinātu ar -11, pie sistēmas (S1) trešā vienādojuma. Rezultātā iegūst viegli atrisināmu sistēmu (S2).

$$(S2) = \begin{cases} x - 4y - 7z = 2 \\ y + 3z = -2 \\ -8z = 24 \end{cases}$$

No trešā vienādojuma nosaka mainīgā z vērtību: $z = -3$, tad no otrā vienādojuma atrod, ka $y = 6$, un, visbeidzot, ievietojot iegūtās mainīgo y un z vērtības pirmajā vienādojumā, izskaitļo x :

$$x = 2 + 7 \cdot (-3) + 4 \cdot 6 = 5$$

Atbilde: $\{(5; 6; -3)\}$.

Piemērs # 2. Atrisināt vienādojumu sistēmu:

$$(S) = \begin{cases} x - 2y + 5z = 5 \\ 3x - 8y - 9z = -6 \\ 4x - 5y - 3z = -1 \end{cases}$$

Izslēdz mainīgo x no sistēmas (S) otrā un trešā vienādojumā un iegūst:

$$(S1) = \begin{cases} x - 2y + 5z = 5 \\ 8y - 24z = -24 \\ 8y - 24z = -19 \end{cases}$$

Nākošajā jau sistēmas (S1) pārveidojuma solī atņem otro vienādojumu no trešā:

$$(S2) = \begin{cases} x - 2y + 5z = 5 \\ 8y - 24z = -24 \\ 0z = 5 \end{cases}$$

Tā kā neeksistē mainīgā z vērtība, kas apmierinātu sistēmas (S2) trešo vienādojumu, tad, dabiski, arī sistēmai (S) nav atrisinājuma.

Atbilde: \emptyset .

Piemērs # 3. Atrisināt vienādojumu sistēmu:

$$(S) = \begin{cases} 2x + y - 3z = -5 \\ 4x + y - 4z = -9 \\ 2x - 2y + 3z = -2 \end{cases}$$

Pārveido doto sistēmu tai ekvivalentā trīsstūra veida sistēmā.

$$(S1) = \begin{cases} x + 0.5y - 1.5z = -2.5 \\ y - 2z = -1 \\ 0z = 0 \end{cases}$$

Sistēmas (S1) trešo vienādojumu apmierina mainīgā z jebkura vērtība. Apzīmē mainīgā z vērtības ar z , un no otrā un trešā vienādojuma nosaka mainīgo y un x atbilstošās vērtības:

$$y = -1 + 2z;$$

$$x = -2.5 + 1.5z - 0.5y = -2.5 + 1.5z - 0.5(2z - 1) = -2.5 + 1.5z - z + 0.5 = 0.5z - 2$$

Atbilde: $\{(0.5z - 2; 2z - 1; z) \mid z \in \mathbb{R}\}$.

Ceturrtā piemēra risinājuma gaitā ir jāmaina vietām vienādojumi.

Piemērs # 4. Atrisināt vienādojumu sistēmu:

$$(S) = \begin{cases} x - 3y + 3z - 6t = 5 \\ 2x - 6y - 2z - 6t = -2 \\ 3x - 9y - z - 4t = -6 \\ 3x - 5y + 2z + 2t = -1 \end{cases}$$

Izslēdz mainīgo x no sistēmas (S) otrā, trešā un ceturrtā vienādojuma:

$$(S1) = \begin{cases} x - 3y + 2z - 6t = 5 \\ -6z + 6t = -12 \\ -7z + 14t = -21 \\ 4y + 4z + 20t = -16 \end{cases}$$

Maina vietām otro un ceturto vienādojumu, un tad dala otrā un trešā vienādojuma abas puses attiecīgi ar 4 un -6. Iegūst:

$$(S2) = \begin{cases} x - 3y + 2z - 6t = 5 \\ y - z + 5t = -4 \\ z - t = 2 \\ -7z + 14t = -21 \end{cases}$$

Pieskaita ceturrtajam vienādojumam trešo vienādojumu, kas pareizināts ar 7:

$$(S3) = \begin{cases} x - 3y + 2z - 6t = 5 \\ y - z + 5t = -4 \\ z - t = 2 \\ 7t = -7 \end{cases}$$

Nu jau bez grūtībām secīgi var iegūt mainīgo t , z , y un x vērtības.

Atbilde: $\{(3; 2; 1; -1)\}$.

Piektajā piemērā ir jāmaina vietām mainīgie.

Piemērs # 5. Atrisināt vienādojumu sistēmu:

$$(S) = \begin{cases} x + 2y - 3z + 5t = 1 \\ 3x + 6y - 5z + 3t = -9 \\ 2x + 4y + z + 3t = -5 \\ 4x + 8y - 3z - 7t = -5 \end{cases}$$

Atņemot no otrā vienādojuma pirmo vienādojumu, no trešā vienādojuma dubultotu pirmo un no ceturtā - četrkārsētu pirmo vienādojumu, iegūst:

$$(S1) = \begin{cases} x + 2y - 3z + 5t = 1 \\ 4z - 12t = -12 \\ 7z - 7t = -7 \\ 9z - 27t = -9 \end{cases}$$

Maina vietām mainīgos y un z un daļa sistēmas (S1) otro vienādojumu ar 4:

$$(S2) = \begin{cases} x - 3z + 2y + 5t = 1 \\ z + 0y - 3t = -3 \\ 7y + 0y - 7t = -7 \\ 9z - 0y - 27t = -9 \end{cases}$$

Nākošajā pārveidojumu solī izslēdz no trešā un ceturtā vienādojuma mainīgo z.

$$(S3) = \begin{cases} x - 3z + 2y + 5t = 1 \\ z + 0y - 3t = -3 \\ 14t = 14 \\ 0t = 18 \end{cases}$$

Sistēmai (S3) nav atrisinājuma, jo satur pretrunīgu vienādojumu: $0t = 18$. Tā kā (S3) ir ekvivalenta sistēmai (S), tad arī tai nav atrisinājuma.

Atbilde: \emptyset .

Ir iespējama situācija, kad risinājuma gaitā jāmaina vietām gan vienādojumi, gan mainīgie.

3.2. Algoritma pieraksts algoritmiskajā valodā

Gausa metodes vispārināts algoritms ir vienlīdz rezultatīvi lietojams gan sistēmām ar vienu atrisinājumu, gan sistēmām bez atrisinājuma, kā arī sistēmām ar bezgala daudziem atrisinājumiem.

Algoritma GAUSA METODE argumenti: vienādojumu skaits R , mainīgo skaits K , koeficientu tabula A un brīvo locekļu tabula B . Algoritma rezultāti ir pazīme PZ , kas informē par atrisinājuma esamību vai neesamību, un atrisinājums (ja tas eksistē), kas pierakstīts tabulā RE . Tabulas RE katra rinda satur viena mainīgā vērtību, kas izteikta ar neatkarīgo mainīgo, kuriem var piešķirt jebkuru vērtību, un skaitļa lineāru kombināciju.

Tā sistēmas

$$\begin{cases} 2x + y - 3z = -5 \\ 4x + y - 4z = -9 \\ 2x - 2y + 3z = -2 \end{cases}$$

atsisinājums $\{(0.5z - 2; 2z - 1; z) \mid z \in \mathbb{R}\}$ tabulā RE tiek pierakstīts sekojoši:

$$\begin{array}{cccc} 0 & 0 & 0.5 & -2 \\ 0 & 0 & 2 & -1 \\ 0 & 0 & 1 & 0 \end{array}$$

Algoritms GAUSA METODE izsauc palīgalgoritmus, kas, savukārt, vēršas vēl pie citiem palīgalgoritmiem. Palīgalgoritmu funkcionālā nozīme ir viegli izsecināma no to tekstiem. Tabulā M fiksē tabulas A kolonnu maiņu, kas saistīta ar mainīgo maiņu vietām. Mainīgais DD satur trīsstūra veida sistēmas galvenās diagonāles nenulles koeficienta lielāko numuru.

Algoritmu tekstus ērtas lasāmības nolūkos izvieto pa lapām tā, lai tie sāktos un beigtos vienā un tajā pašā lapā.

algoritms GAUSA METODE (arg ves R, K, arg re tab A[1:R,1:K], arg re tab B[1:R], rez bur PZ,

rez re tab RE[1:K,1:K+1])

sākums ves tab M[1:K], ves I, ID, DD

priekš I no 1 līdz K

cikls

M[I]:=I

stāt

ID:=1

kamēr ID<=MIN(R, K)

cikls

ja A[ID, ID]=0

tad ELEMENTA MAIŅA (R, K, A, B, M, ID)

viss

ja A[ID, ID]<>0

tad

RINDAS ATŅEMŠANA (R, K, A, B, ID)

ID:=ID+1

citādi

DD:=ID-1

ID:=MIN(R, K)+2

viss

stāt

ja ID=MIN(R, K)+1

tad DD:=MIN(R, K)

viss

ATRISINĀJUMA ESAMĪBAS PĀRBAUDE (R, B, DD, PZ)

ja PZ="ir"

tad ATRISINĀJUMA VĒRTĪBU NOTEIKŠANA (R, K, A, B, M, DD, RE)

viss

beigas

algoritms ves MIN (arg ves I, J)

sākums

ja $I < J$

tad vērtība:=I

citādi vērtība:=J

viss

beigas

algoritms ELEMENTA MAIŅA (arg ves R, K, arg rez re tab A[1:R, 1:K], arg rez re tab B[1:R],

 arg rez ves tab M[1:K], arg ves ID)

sākums

 ELEMENTA MAIŅA 1 (R, K, A, B, M, ID)

 ja A[ID, ID]=0

tad ELEMENTA MAIŅA 2 (R, K, A, B, M, ID)

viss

beigas

algoritms ELEMENTA MAIŅA 1 (arg **ves** R, K, arg **rez re tab** A[1:R,1:K], arg **rez re tab** B[1:R],
arg **rez ves tab** M[1:K], arg **ves** ID)

sākums **ves** IR, IK

IR:=ID+1

kamēr IR<=R

cikls

ja A[IR, ID]<>0

tad

 RINDU APMAIŅA (R, K, A, B, ID, IR)

 IR:=R+1

citādi IR:=IR+1

viss

stāt

ja A[ID, ID]=0

tad

 IK:=ID+1

kamēr IK<=K

cikls

ja A[ID, IK]<>0

tad

 KOLONNU APMAIŅA (R, K, A, M, ID, IK)

 IK:=K+1

citādi IK:=IK+1

viss

stāt

viss

beigas

algoritms RINDU APMAIŅA (**arg ves** R, K, **arg rez re tab** A[1:R, 1:K], **arg rez re tab** B[1:R],
arg ves ID, IR)

sākums **ves** I, **re** P

priekš I no 1 līdz K

cikls

P:=A[ID, I]

A[ID, I]:=A[IR, I]

A[IR, I]:=P

stāt

P:=B[ID]

B[ID]:=B[IR]

B[IR]:=P

beigas

algoritms KOLONNU APMAIŅA (**arg ves** R, K, **arg rez re tab** A[1:R, 1:K],

arg rez ves tab M[1:K], **arg ves** ID, IK)

sākums **ves** I, **re** P, **ves** PP

priekš I no 1 līdz R

cikls

P:=A[I, ID]

A[I, ID]:=A[I, IK]

A[I, IK]:=P

stāt

PP:=M[ID]

M[ID]:=M[IK]

M[IK]:=PP

beigas

algoritms ELEMENTA MAINĀ 2 (arg **ves** R, K, arg **rez re tab** A[1:R, 1:K], arg **rez re tab** B[1:R],
 arg **rez ves tab** M[1:K], arg **ves** ID)

sākums **ves** JD

JD:=ID+1

kamēr JD<=MIN(R, K)

cikls

ja A[JD, JD]<>0

tad

 RINDU APMAIŅA (R, K, A, B, ID, JD)

 KOLONNU APMAIŅA(R, K, A, M, ID, JD)

viss

ja A[ID, ID]=0

tad

 KOLONNAS CAURSKATE (R, K, A, B, M, ID, JD)

viss

ja A[ID, ID]=0

tad

 RINDAS CAURSKATE (R, K, A, B, M, ID, JD)

viss

ja A[ID, ID]=0

tad JD:=JD+1

citādi JD:=MIN(R, K)+1

viss

stāt

beigas

algoritms KOLONNAS CAURSKATE (arg ves R, K, arg rez re tab A[1:R, 1:K],

arg rez re tab B[1:R], arg rez ves tab M[1:K], arg ves ID, JD)

sākums ves IR

IR:=JD+1

kamēr IR<=R

cikls

ja A[IR, JD]<>0

tad

 RINDU APMAIŅA (R, K, A, B, ID, IR)

 KOLONNU APMAIŅA (R, K, A, M, ID, JD)

 IR:=R+1

citādi IR:=IR+1

viss

stāt

beigas

algoritms RINDAS CAURSKATE (arg ves R, K, arg rez re tab A[1:R, 1:K], arg rez re tab B[1:R],

arg rez ves tab M[1:K], arg ves ID, JD)

sākums ves IK

IK:=JD+1

kamēr IK<=K

cikls

ja A[JD, IK]<>0

tad

 RINDU APMAIŅA (R, K, A, B, ID, JD)

 KOLONNU APMAIŅA (R, K, A, M, ID, IK)

 IK:=K+1

citādi IK:=IK+1

viss

stāt

beigas

algoritms RINDAS ATŅEMŠANA (arg ves R, K, arg rez re tab A[1:R, 1:K],

arg rez re tab B[1:R], arg ves ID)

sākums ves IR, I

IR:=ID+1

kamēr IR<=R

cikls

B[IR]:=B[IR]*A[ID, ID]-B[ID]*A[IR, ID]

I:=K

kamēr I>=ID

cikls

A[IR, I]:=A[IR, I]*A[ID, ID]-A[ID, I]*A[IR, ID]

I:=I-1

stāt

IR:=IR+1

stāt

beigas

algoritms ATRISINĀJUMA ESAMĪBAS PĀRBAUDE (arg ves R, arg re tab B[1:R], arg ves DD,

rez bur PZ)

sākums ves I

PZ:="ir"

I:=DD+1

kamēr I<=R

cikls

ja B[I]<>0

tad

PZ:="nē"

I:=R+1

citādi I:=I+1

viss

stāt

beigas

algoritms ATRISINĀJUMA NOTEIKŠANA (arg ves R, K, arg re tab A[1:R, 1:K],

arg re tab B[1:R], arg ves tab M[1:K], arg ves DD, arg rez re tab RE[1:K, 1:K+1])

sākums ves I, J

I:=DD+1

kamēr I<=K

cikls

priekš J no 1 līdz K+1

cikls

RE[M[I], J]:=0

stāt

RE[M[I], M[I]]:=1

I:=I+1

stāt

I:=DD

kamēr I>=1

cikls

priekš J no 1 līdz K

cikls

RE[M[I], J]:=0

VĒRTĪBAS NOTEIKŠANA (R, K, A, M, I, J, RE)

RE[M[I], J]:=RE[M[I], J]/A[i, I]

stāt

RE[M[I], J]:=B[I]

VĒRTĪBAS NOTEIKŠANA (R, K, A, M, I, J, RE)

RE[M[I], J]:=RE[M[I], J]/A[i, I]

I:=I-1

stāt

beigas

algoritms VĒRTĪBAS NOTEIKŠANA (arg ves R, K, arg re tab A[1:R, 1:K],

arg ves tab M[1:K], arg ves I, J, arg rez re tab RE[1:K, 1:K+1])

sākums ves L

L:=I+1

kamēr I<=K

cikls

RE[M[I], J]:=RE[M[I], J]-A[I, L]*RE[M[L], J]

L:=L+1

stāt

beigas

3.3. Algoritma programmrealizācija MSX BASIC

Protams, ņemot vērā algoritmiskās valodas un BASIC izteiksmes līdzekļu iespējas un dažādību, algoritma programmrealizācijas BASIC pieraksts burtiski nedublē algoritma pierakstu algoritmiskajā valodā, taču funkcionālie bloki ir viegli atpazīstami. Apakšprogrammas masīviem un mainīgajiem lietoti tādi paši nosaukumi kā atbilstošajām tabulām un lielumiem algoritma pierakstā. Apakšprogrammu izsaucošajā programmā ir jābūt aprakstītiem sekojošiem masīviem:

DIM A(R%,K%), DIM B(K%), DIM RE(K%,K%+1)

Apakšprogrammas teksts:

9000 DIM M%(K%) /GAUSA METODE/

9002 FOR I%=1 TO K%:M%(I%)=I%:NEXT I% /MIN/

9004 IF R%>K% THEN MI%=K% ELSE MI%=R%

9006 ID%=1

9008 IF ID%>MI% THEN 9016

9010 IF A(ID%,ID%)=0 THEN GOSUB 9100

9012 IF A(ID%,ID%)<>0 THEN GOSUB 9200:ID%=ID%+1

ELSE DD%=ID%-1:ID%=MI%+2

9014 GOTO 9008

9016 IF ID%=MI%+1 THEN DD%=MI%


```

3018 PZ$="IR":I%=DD%+1 /ATRISINĀJUMA
3020 IF I%>R% THEN 9026 ESAMĪBAS PĀRBAUDE/
3022 IF B(I%)<>0 THEN PZ$="NAV":I%=R%+1 ELSE I%=I%+1
3024 GOTO 9020
3026 IF PZ$="IR" THEN GOSUB 9300
3028 RETURN

9100 IR%=ID%+1 /ELEMENTA MAIŅA 1/
9102 IF IR%>R% THEN 9108
9104 IF A(IR%,ID%)<>0 THEN GOSUB 9150:IR%=R%+1
      ELSE IR%=IR%+1
9106 GOTO 9102
9108 IF A(ID%,ID%)<>0 THEN RETURN
9110 IK%=ID%+1
9112 IF IK%>K% THEN 9118
9114 IF A(ID%,IK%)<>0 THEN GOSUB 9160:IK%=K%+1
      ELSE IK%=IK%+1
9116 GOTO 9112
9118 IF A(ID%,ID%)<>0 THEN RETURN

9120 JD%=ID%+1 /ELEMENTA MAIŅA 2/
9122 IF JD%>MI% THEN RETURN
9124 IF A(JD%,JD%)<>0 THEN IR%=JD%:GOSUB 9150:IK%=JD%:
      GOSUB 9160:RETURN

9126 IR%=JD%+1 /KOLONNAS
9128 IF IR%>R% THEN 9134 CAURSKATE/
9130 IF (IR%,JD%)<>0 THEN GOSUB 9150:IK%=JD%:GOSUB 9160:
      IR%=R%+1 ELSE IR%=IR%+1
9132 GOTO 9128
9134 IF A(ID%,ID%)<>0 THEN RETURN

```

```

9136 IK%=JD%+1 /RINDAS CAURSKATE/
9138 IF IK%>K% THEN 9144
9140 IF A(JD%,IK%)<>0 THEN IR%=JD%:GOSUB 9150:GOSUB 9160:
      IK%=K%+1 ELSE IK%=IK%+1
9142 GOTO 9138
9144 IF A(ID%,ID%)<>0 THEN RETURN
9146 JD%=JD%+1:GOTO 9122

9150 FOR I%=1 TO K%:P=A(ID%,I%):A(ID%,I%)=A(IR%,I%): /RINDU APMAIŅĀ/
      A(IR%,I%)=P:NEXT I%
9152 P=B(ID%):B(ID%)=B(IR%):B(IR%)=P:RETURN

9160 FOR I%=1 TO R%:P=A(I%,ID%):A(I%,ID%)=A(I%,IK%): /KOLONNU APMAIŅĀ/
      A(I%,IK%)=P:NEXT I%
9162 PP%=M%(ID%):M%(ID%)=M%(IK%):M%(IK%)=PP%:RETURN

9200 IR%=ID%+1 /RINDU ATŅEMŠANA/
9202 IF IR%>R% THEN RETURN
9204 B(IR%)=B(IR%)*A(ID%,ID%)-B(ID%)*A(IR%,ID%)
9206 I%=K%
9208 IF I%<ID% THEN IR%=IR%+1:GOTO 9202
9210 A(IR%,I%)=A(IR%,I%)*A(ID%,ID%)-A(ID%,I%)*A(IR%,ID%):
      I%=I%-1:GOTO 9208

9300 I%=DD%+1 /ATRISINĀJUMA
9302 IF I%>K% THEN 9306 NOTEIKŠANA/
9304 FOR J%=1 TO K%+1:RE(M%(I%),J%)=0:NEXT J%:
      RE(M%(I%),M%(I%))=1:I%=I%+1:GOTO 9302
9306 I%=DD%
9308 IF I%<1 THEN RETURN
9310 FOR J%=1 TO K%:RE(M%(I%),J%)=0:GOSUB 9320:
      RE(M%(I%),J%)=RE(M%(I%),J%)/A(I%,I%):NEXT J%
9312 RE(M%(I%),J%)=B(I%):GOSUB 9320:
      RE(M%(I%),J%)=RE(M%(I%),J%)/A(I%,I%)
9314 I%=I%-1:GOTO 9308

```

9320	L%=I%+1	✓VĒRTĪBAS
9322	IF L%>K% THEN RETURN	NOTEIKŠANA/
9324	RE(M%(I%),J%)=RE(M%(I%),J%)-A(I%,L%)*RE(M%(L%),J%):	
	L%=L%+1:GOTO 9322	

Apakšprogrammas darbderību var pārbaudīt ar sekojošas programmas palīdzību:

```

11  INPUT "R";R%;INPUT "K";K%
13  DIM A(R%,K%):DIM B(R%):DIM RE (K%,K%+1)
15  FOR I%=1 TO R%:FOR J%=1 TO K%:INPUT "A(I,J)":A(I%,J%):
    NEXT J%:INPUT "B(I)";B(I%):NEXT I%

21  GOSUB 9000

31  PRINT PZ$
33  IF PZ$="IR" THEN FOR I%=1 TO K%:FOR J%=1 TO K%:
    PRINT RE(I%,J%);:NEXT J%:PRINT RE(I%,K%+1):NEXT I%

91  END

```

4. Mācību datu bāze

4.1. Vispārējās ziņas

4.1.1. Datu bāzes vadības sistēmas darbošanās vide

Datu bāzes vadības sistēma darbojas KYBT-86 (ДБК, БК-0010 (BASIC)) vidē, izmantojot lokālo tīklu (tikla programma NET vai NET3).

4.1.2. Datu bāzes vadības sistēmas palaišana

Uz datora ДБК palaiž tīkla programmu turpinājuma režīmā un datorā БК ievada komandu BLOAD"TT:D",R.

4.1.3. Datu bāzes struktūra un parametri

Datu bāzi raksturo:

- datu organizācija - tabula (relāciju tipa datu bāze);
- ierakstu maksimālais skaits - 999 (atkarīgs no ieraksta garuma);
- lauku maksimālais skaits ierakstā - 255;
- divi lauka tipi - teksta un skaitļa;
- teksta lauka maksimālais garums - 255 simboli, lauks var saturēt jebkuras ar datora klaviatūru ievadāmas zīmes, izņemot simbolus "" un "?". Latviešu alfabēta burtus ar diakritiskajām zīmēm iegūst, nospiežot taustiņu AP2 vienlaicīgi ar attiecīgā burta taustiņu;
- skaitļa lauka maksimālais garums - 14 simboli. Lauks var saturēt zīmju virkni, kas reprezentē skaitliskas vērtības pierakstu desmitnieku skaitīšanas sistēmā (cipari, mīnuss zīme un decimālais punkts). Decimāldaļskaitļa pieraksta laukam atvēlēto simbolu skaitā ietilpst gan zīmes, gan decimālpunkta pozīcijas;
- lauka nosaukuma garums - 12 simboli. To var veidot no jebkurām ar datora klaviatūru ievadāmām zīmēm.

Datu bāze tiek veidota un apstrādāta kā viens un nedalāms objekts datora БК atmiņā (saturiskās informācijas apjoms ierobežots). Informācijas apmaiņa starp datora un ārējo atmiņu notiek, tikai saglabājot un ielādējot datu bāzi, kā arī, sagatavojot pārskata teksta failu izvadei ar drukas iekārtu.

4.1.4. Datu bāzes vadības sistēmas funkcijas

Datu bāzes vadības sistēma realizē sekojošas funkcijas:

- datu bāzes apraksti;
- datu bāzes aizpildi un labošanu;
- datu bāzes ierakstu apskati;
- datu bāzes saglabāšanu;
- datu bāzes ielādi;
- datu bāzes visu ierakstu dzēšanu;
- datu bāzes dzēšanu;
- ierakstu atlasī (iezīmēšana);
- statistisko apstrādi;
- pārskata veidošanu izvadei ar drukas iekārtu.

Datu bāzes aprakste ietver nosaukuma piešķiršanu datu bāzei un ierakstu struktūras definēšanu. Ieraksta struktūru nosaka gan lauku secība, gan to apraksti. Lauka aprakstu veido nosaukums un atribūti - tips (teksts vai skaitlis) un garums (skaitļa tipa laukam papildus jānorāda decimālās daļas garums).

Datu bāzes aizpilde ir vērtību piešķire (ievade) ierakstu laukiem. Labošana ir ierakstu pievienošana, iestarpināšana starp ierakstiem, izmešana un ieraksta lauku vērtību nomaiņa. Datu bāzes aizpilde un labošana notiek datora BK atmiņā. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

Datu bāzes ierakstu apskati var veikt gan pa ierakstiem (tabulas rindas), gan pa laukiem (tabulas ailes). Var tikt apskatīti datu bāzes visi vai tikai iezīmēto ierakstu kopai piederošie ieraksti. Atzīme "" pie ieraksta numura norāda, ka šis ieraksts ir iezīmēts.

Datu bāzes saglabāšana ir datora BK atmiņā esošās datu bāzes ierakstīšana disketes failā. Par faila vārdu izmanto datu bāzes nosaukumu. Datu bāzes fails tiek saglabāts ar paplašinājumu BIN.

Datu bāzes ielāde ir pretēja darbība datu bāzes saglabāšanai. Dati no disketes faila tiek pārsūtīti datora BK atmiņā.

Datu bāzes visu ierakstu dzēšanas rezultātā datora BK atmiņā tiek dzēsta datu bāzes saturiskā informācija (ieraksti). Datu bāzes apraksts saglabājas. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

Datu bāzes dzēšanas rezultātā datora БК atmiņā tiek dzēsta gan datu bāzes saturiskā informācija, gan datu bāzes apraksts. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

Ierakstu atlase (iezīmēšana) ietver nosaukuma piešķiršanu iezīmēto ierakstu kopai un atlases nosacījumu uzdošanu. Atlases nosacījumus var uzdot gan ar vienkāršām loģiskām izteiksmēm, kas satur vienu apgalvojumu, gan saliktām, izmantojot loģiskās operācijas "UN" un "VAI". Operācijas "UN" prioritāte ir augstāka nekā operācijas "VAI" prioritāte, tas nozīmē, ka "UN" piesaista operandus ciešāk nekā "VAI". Iekavu lietošana nav paredzēta. Skaitļu lauku vērtībām tiek pieļautas 6 veidu salīdzināšanas ar skaitlisku konstanti. Tās ir: =; ≠; >; ≥; <; ≤. Teksta lauku vērtības var tikt pārbaudītas sakrišanai vai nesakrišanai ar paraugtekstu. Paraugtekstā simboliem "?" un "*" ir īpaša nozīme. "?" nozīmē, ka teksta lauka vērtības atbilstošā pozīcija var saturēt jebkuru zīmi. Ar "*" apzīmētā vietā salīdzināmā teksta lauka vērtība var saturēt patvaļīgu zīmju virkni, ieskaitot tukšu. Iezīmēto ierakstu kopā var veikt ierakstu apskati, skaitļu lauku statistisko apstrādi. Arī pārskatu var sagatavot, izvēloties tikai iezīmētos ierakstus.

Statistiskā apstrāde sniedz ziņas par visu vai iezīmēto ierakstu skaitļu lauku mazāko, lielāko un vidējo aritmētisko vērtību, kā arī par amplitūdu (starpība starp lielāko un mazāko vērtību) un summu.

Pārskata veidošanas režīmā tiek sagatavots teksta fails disketē, kas satur lietotāja izvēlētu datu bāzes informāciju. Izvēli var attiecināt gan uz ierakstiem, gan laukiem. Izvadāmo informāciju var sakārtot pēc viena vai vairāku lauku vērtībām, norādot lauku kārtības prioritātes secību. Skaitļu laukus var kārtot augošā vai dilstošā secībā, teksta lauki tiek kārtoti leksikogrāfiskā secībā. Sagatavotā teksta faila saturu var izvadīt ar datora ДБК speciālu programmu palīdzību gan ar D100, gan Robotron tipa drukas iekārtām.

Atgādinājums!

Veidojot datu bāzes un lauku nosaukumus, lauku atribūtus un lauku vērtības, darbojas visi datora БК rediģēšanas taustiņi.

4.1.5. Lietotāja un sistēmas dialogs

Lietotāja dialogs ar datu bāzes vadības sistēmu tiek realizēts ar komandkartēm. Darba režīma izvēle un operāciju izpilde notiek pēc atbilstošā cipara taustiņa nospiešanas.

Datu bāzes vadības sistēmas galvenā komandkarte (lietotājam iespējams izvēlēties ar "*" atzīmēto režīmu):

Darba režīms	Nav bāze	Ir tikai bāzes apraksts	Ir aizpildīta bāze
0 Datu bāzes ielāde	*		
1 Datu bāzes aprakste	*	*	*
2 Bāzes aizpilde un labošana		*	*
3 Datu bāzes ierakstu apskate			*
4 Datu bāzes saglabāšana		*	*
5 Visu ierakstu dzēšana			*
6 Datu bāzes dzēšana		*	*
7 Statistiskā apstrāde			*
8 Ierakstu atlase			*
9 Pārskata veidošana			*

Nospiežot taustiņu STOP, datu bāzes vadības sistēma atsāk izvēlētā darba režīma izpildi.

Darbošanos ar datu bāzes vadības sistēmu beidz, nospiežot taustiņu E un nospiežot J, apstiprina izvēli.

Atgādinājums!

Ar aizpildītu datu bāzi saprot datu bāzi, kura satur vismaz vienu aizpildītu ierakstu.

4.1.6. Programmas

Datu bāzes vadības sistēmu veido sekojoši programmu moduļi, kas izpilda attiecīgas funkcijas:

D.BIN

- titullapas izvadi,

DB0.BIN	- galvenās komandkartes apstrādi, - datu bāzes saglabāšanu, - datu bāzes ielādi, - datu bāzes visu ierakstu dzēšanu, - datu bāzes dzēšanu;
DB1.BIN	- datu bāzes apraksti;
DB2.BIN	- datu bāzes aizpildi un labošanu;
DB3.BIN	- datu bāzes ierakstu apskati;
DB4.BIN	- statistisko apstrādi;
DB5.BIN un DB5A.BIN	- ierakstu atlasī (iezīmēšana);
DB6.BIN un DB6A.BIN	- pārskata veidošanu.

Sagatavotās atskaites izvadi realizē drukas programmas:

D100L.SAV	- drukas iekārtai D100 vai
ROBL.SAV	- drukas iekārtai Robotron.

Datu bāzes vadības sistēmas pilnvērtīgai funkcionēšanai nepieciešami šādi sistēmlīdzekļi:

RT11CD.SYS; SWAP.SYS; TT.SYS; LP.SYS; MX.SYS; SL.SYS;
PIP.SAV; DIR.SAV; DUP.SAV; DATE.SAV un
tikla programma NET.SAV vai NET3.SAV.

Atgādinājums!

Strādājot ar datu bāzes vadības sistēmu, ieteicams izmantot datora DBK abas diskiekārtas. Nultajā disku iekārtā ievieto disketi ar sistēmlīdzekļiem, tikla programmu un drukas programmām. Uz disketes pirmajā disku iekārtā glabā datu bāzes vadības sistēmas programmas moduljus, datu bāžu un pārskatu failus. Pie šāda programmu un datu failu izvietojuma tikla programmas palaišanai jāuzdod komandas:

```
ASSIGN MX1: DK:
R NET
```

Tikla programmas darba laikā nedrīkst mainīt disketi, kas satur datu bāzes vadības sistēmas programmas moduljus un datu failus. Ja ir nepieciešamība mainīt disketi, tad jāaptur tikla programma, jānomaina diskete un atkārtoti jāpalaiž tikla programma.

Nevajadzīgos datu failus izdzēs ar komandu DELETE un disketes saturu sablīvē ar komandu SQUEEZE.

4.2. Datu bāzes vadības sistēmas funkciju apraksts

4.2.1. Datu bāzes ielāde

Pēc režīma izvēles instrukcijas logā ziņo, ka jāievada datu bāzes nosaukums. Tiek atgādināts, ka nosaukuma pierakstā var lietot lielos latīņu burtus un ciparus un ka nosaukums nedrīkst sākties ar D. Veidojot nosaukumu, darbojas visi datora BK rediģēšanas taustiņi. Nosaukumu ievada, nospiežot taustiņu ←.

Ja disketē ir fails ar ievadīto nosaukumu, tad instrukcijas logā ziņo, ka notiek ielāde un jāgaida. Pēc datu bāzes faila saņemšanas datora BK atmiņā informācijas logā redzams datu bāzes esošā stāvokļa raksturojums: lauku, aizpildīto un maksimālais iespējamo ierakstu skaits, kā arī iezīmēto ierakstu (ja tādi ir bijuši) kopas nosaukums un skaits.

Ja fails nav datu bāzes fails (kļūmīgas saglabāšanas rezultāts vai nejauša nosaukumu sakritība), tad instrukcijas logā par to ziņo, un darba turpināšanai ir jānospiež taustiņš ←.

Ja disketē nav faila ar ievadīto nosaukumu, tad instrukcijas logā par to ziņo, un darba turpināšanai jānospiež taustiņš ←.

Gan sekmīgas, gan nesekmīgas datu bāzes ielādes gadījumā sistēma atgriežas izvēles režīmā - uz ekrāna galvenā komandkarte.

4.2.2. Datu bāzes aprakste

Jaunas datu bāzes apraksta veidošana sākas ar datu bāzes nosaukuma ievadišanu. Nosaukuma pierakstā var lietot lielos latīņu burtus un ciparus. Nosaukums nedrīkst sākties ar D, tā garums nedrīkst pārsniegt 6 simbolus. Ja datu bāzes nosaukumu sistēma akceptē, tad lietotājam jāsāk veidot pirmā lauka apraksts.

Lauka aprakstu veido nosaukums un atribūti.

Lauka nosaukuma maksimālais garums ir 12 simboli. Datu bāzes vadības sistēma neakceptē lauku nosaukumus, kas satur tikai tukšumus. Lauka nosaukums var saturēt jebkuras ar datora klaviatūru ievadāmas zīmes.

Latviešu alfabēta burtus ar diakritiskajām zīmēm iegūst, nospiežot taustiņu AP2 vienlaicīgi ar attiecīgā burta taustiņu.

Datu bāzes vadības sistēma izšķir divus lauka tipus - skaitļa un teksta laukus, aprakstā attiecīgi apzīmējot ar S vai T. Līdz taustiņa ← nospiešanai lauka tipu var mainīt, nospiežot taustiņu T vai S.

Teksta lauka atribūts ir tā garums robežās no 1 līdz 255 pozīcijām. Ja lauka garums neatrodas šajās robežās vai tā pieraksts nav skaitlis, tad lauka garuma ievads netiek akceptēts.

Skaitļa lauka maksimālais garums - 14 simboli. Norādot skaitļa lauka kopējo garumu, jāievēro, ka atvēlēto simbolu skaitā bez ciparu pozīcijām ietilpst gan zīmes, gan decimālpunkta pozīcijas. Laukam, kuram atvēlētas 3 un vairāk pozīcijas, jānorāda decimālās daļas garums. Decimālās daļas garumam jābūt vismaz par 2 pozīcijām isākam kā visa lauka garumam. Ja kāds no skaitļa lauka atribūtiem neatrodas informācijas logā norādītajās robežās vai nav skaitlis, tad lauka garuma ievads netiek akceptēts.

Atkarībā no tā, vai datu bāze satur vai nesatur aizpildītus ierakstus, datu bāzes aprakstes režīmā lietotājs var no komandkartes izvēlēties sekojošas darbības (atzīmētas ar *):

	Darbība	Ir tikai bāzes apraksts	Ir aizpildīta bāze
↑ ↓	lauka izvēle	*	*
BC	pirmais lauks	*	*
KT	pēdējais lauks	*	*
ШАГ	+9 lauki	*	*
←	lauka veidošana	*	*
I	iestarpināt lauku	*	
D	izmest lauku	*	
СБР	nomainīt lauku	*	
E	beigt	*	*

Ar taustiņu ↑ vai ↓, BC, KT un ШАГ palīdzību, novietojot kursoru uz lauka numura, izvēlas interesējošo lauku.

Lauka veidošanas operācija darbojas dažādi atkarībā no tā, vai datu bāze satur vai nesatur aizpildītus ierakstus:

- aizpildītai datu bāzei iespējams mainīt tikai lauka nosaukumu;

- ja datu bāze nesatur ierakstus un ja kursora atrodas uz pēdējā lauka numura, tad, nospiežot taustiņu ←, ir jāveido jauna lauka apraksts;
- ja datu bāze nesatur ierakstus un ja kursora atrodas uz izveidota lauka numura, tad, nospiežot taustiņu ←, var pakāpeniski rediģēt šī lauka aprakstu. Ja nosaukums vai kāds no atribūtiem nav jāmaina, tad, nospiežot taustiņu ←, attiecīgi lauka nosaukums vai atribūts netiks izmainīts.

Ierakstu struktūru datu bāzē nosaka ne tikai lauku apraksti, bet arī to secība. Neaizpildītas datu bāzes apraksta veidošanas laikā lauku secību var mainīt, iestarpinot vai izmetot lauku. Lauku iestarpināšanai novieto kursoru uz tā lauka numura, kura vietā jāatrodas jaunajam laukam. Pēc taustiņa I nospiešanas lietotājam jāsāk iestarpinātā lauka apraksta veidošana. Kārtējā lauka izmešana notiek, nospiežot taustiņu D. Ja datu bāze satur tikai vienu lauka aprakstu, tad to izmest nevar. Lauka apraksta nomaiņai jānospiež taustiņš CБP un jāsāk veidot lauka apraksts no jauna.

Ja datu bāzes apraksta veidošana ir pabeigta, tad, nospiežot taustiņu E, datu bāzes vadības sistēma pēc programmas pārlādes pāriet izvēles režīmā. Uz ekrāna ir redzama galvenā komandkarte.

Atgādinājums!

Datu bāzes apraksta veidošanas laikā informācijas logā ziņo par izveidoto lauku skaitu un maksimāli iespējamo ierakstu skaitu datu bāzē ar esošo struktūru.

Veidojot datu bāzes aprakstu (ievadot nosaukumus un atribūtus), darbojas visi datora БК rediģēšanas taustiņi.

Nosaukumu un atribūtu ievads notiek, nospiežot taustiņu ←.

Lauku maksimālais skaits - 255.

Ar aizpildītu datu bāzi saprot datu bāzi, kas satur vismaz vienu aizpildītu ierakstu.

4.2.3. Datu bāzes aizpilde un labošana

Datu bāzes aizpildes un labošanas režīmā notiek vērtību piešķīre (ievade) ierakstu laukiem, ieraksta pievienošana, iestarpināšana un izmešana, kā arī ieraksta lauku vērtību nomaiņa.

Sākot aizpildīt datu bāzi, uz ekrāna tiek izvadīti lauku nosaukumi un lauku vērtību logi pirmā ieraksta aizpildei. Vērtības jāpiešķir visiem laukiem pēc kārtas, sākot ar pirmo lauku. Instrukcijas logā ziņo par aizpildāmā lauka tipu. Datu bāzes vadības sistēma nepieļauj neaizpildītas lauku vērtības. Vērtības piešķire laukam notiek pēc taustiņa ← nospiešanas.

Teksta tipa lauka vērtība var saturēt jebkuras no datora klaviatūras ievadāmas zīmes, izņemot simbolus "*" un "?", kurus izmanto ierakstu atlasē. Latviešu alfabēta burtus ar diakritiskajām zīmēm iegūst, nospiežot taustiņu AP2 vienlaicīgi ar attiecīgā burta taustiņu.

Instrukcijas logā par skaitļa lauku ziņo, vai ievadāmais skaitlis ir veselais vai decimāldaļskaitlis. Lauka vērtība var saturēt mīnuss zīmi un ciparus. Decimāldaļskaitļa laukam decimālpunkta pozīcija tiek izvadīta vērtības logā saskaņā ar lauka aprakstu. Ja ievadāmās vērtības veselā daļa neaizņem visas tai atvēlētās pozīcijas, tad, nospiežot punkta taustiņu, ciparu virkne līdz kursoram tiek pārvietota pie decimālpunkta, kursors pārvietojas uz pirmo decimāldaļas pozīciju. Ja skaitļa tipa laukā ievadītā informācija nav skaitlis, tad lauka vērtība netiek akceptēta.

Pēc vērtības piešķires pēdējam laukam, lai veidotu jaunu ierakstu, jānospiež taustiņš ←. Uz ekrāna tiek izvadīts nākamā ieraksta numurs, lauku nosaukumi, sākot ar pirmo, un lauku vērtību logi. Kursors atrodas pirmā lauka vērtības logā.

Ja jaunu ierakstu neveido un nospiež taustiņu E, tad, tāpat kā izvēloties aizpildes un labošanas režīmu aizpildītai datu bāzei, instrukcijas logā redzama sekojoša komandkarte (turpmāk saukta par pirmo komandkarti):

B	darbs ar bāzi
← →	ieraksta izvēle
BC	bāzes sākums
KT	bāzes beigās
ШИАГ	+5 ieraksti
←	fiksēta lauka apskate ierakstos
E	beigt

Darbs ar bāzi ietver ierakstu pievienošanu datu bāzes beigās, iestarpināšanu starp ierakstiem, izmešanu un nomaiņu (ieraksta lauku vērtību nomaiņu). Pēc

taustiņa B nospiešanas instrukcijas logā tiek izvadīta šāda komandkarte:

V	veidot ierakstu
D	izmest ierakstu
I	iestarpināt ierakstu
←	ieraksta apstrāde
← →	cits ieraksts
E	beigt

Nospiežot taustiņu V, datu bāzei tiek pievienots jauns ieraksts - uz ekrāna tiek izvadīti lauku nosaukumi un tukši vērtību logi. Pēc vērtības piešķires pēdējam laukam instrukcijas logā ziņo, ka pēc taustiņa ← nospiešanas tiks veidots nākošais ieraksts. Nospiežot taustiņu E, notiek datu bāzes vadības sistēmas atgriešanās uz pirmo komandkarti.

Datu bāzes saturu var mainīt, ievietojot jaunus vai izmetot aizpildītus ierakstus. Ar taustiņu ← vai → palīdzību var panākt, ka uz ekrāna tiek izvadīts interesējošais ieraksts. Nospiežot taustiņu D, instrukcijas logā redzams jautājums:

Izmest šo ierakstu? J/N

Nospiežot taustiņu J, uz ekrāna izvadītais ieraksts no datu bāzes tiek dzēsts. Datu bāzes ieraksti tiek pārnumurēti. Uz ekrāna tiek izvadīts kārtējā ieraksta saturs. Par kārtējo ierakstu kļūst nākošais aiz izmestā, izņemot gadījumu, kad tika izmests pēdējais ieraksts. Šajā gadījumā uz ekrāna datu bāzes vadības sistēma izvada iepriekšējo ierakstu.

Ja nospiež taustiņu N, tad datu bāzes vadības sistēma atgriežas uz pirmo komandkarti. Datu bāzē izmaiņas nenotiek.

Nospiežot taustiņu I, datu bāzē pirms kārtējā ieraksta tiek iestarpināts tukšs ieraksts, kas lietotājam ir jāaizpilda. Datu bāzes sekojošie ieraksti tiek pārnumurēti. Pēc vērtības piešķires pēdējam laukam datu bāzes vadības sistēma atgriežas uz pirmo komandkarti.

Izvēloties ieraksta apstrādi, nospiežot taustiņu ←, ir iespējama ierakstu apskate un labošana pa laukiem. Kursors pārvietojas no ieraksta numura uz kārtējā

ieraksta pirmā lauka numuru. Instrukcijas logā redzama jauna komandkarte:

↑ ↓	lauka izvēle
BC	pirmais lauks
KT	pēdējais lauks
IIIAΓ	+5 lauki
←	lauka labošana
← →	cits ieraksts
E	beigt

Ar taustiņu ↑ vai ↓, BC, KT un IIIAΓ palīdzību var izvēlēties interesējošo lauku. Izvēlēto lauku katrā nākošā vai iepriekšējā ierakstā var aplūkot pēc taustiņa → vai taustiņa ← nospiešanas.

Ja nepieciešama lauka vērtības nomaiņa, tad pēc kursora novietošanas uz interesējošā lauka numura, jānospiež taustiņš ←. Kursors pārvietojas uz lauka vērtības logu, un lietotājs var izmainīt lauka vērtību. Instrukcijas logā ir informācija par lauka atribūtiem. Jaunās vērtības piešķire notiek, nospiežot taustiņu ←. Par jauno lauka vērtību datu bāzes vadības sistēma uzskata visu vērtības logā redzamo informāciju, neatkarīgi no kursora atrašanās vietas. Ja vērtības nomaiņa izdarīta nekorekti, tad jaunā vērtība netiek akceptēta, līdz kļūda nav izlabota un atkārtoti nospiests taustiņš ←. Ja lauka vērtības nomaiņa izdarīta korekti, tad pēc taustiņa ← nospiešanas uz ekrāna tiek izvadīts nākošā ieraksta saturs, kursoram atrodoties uz izlabotā lauka numura. Ja izlabotais lauks atrodas pēdējā ierakstā, tad pēc lauka vērtības nomaiņas uz ekrāna saglabājas šī ieraksta saturs.

Nospiežot taustiņu E, datu bāzes vadības sistēma, uz ekrāna saglabājot kārtējo ierakstu, atgriežas uz iepriekšējo komandkarti. Iziet no datu bāzes aizpildes un labošanas režīma var tikai no pirmās komandkartes, tādēļ atkārtoti jānospiež taustiņš E.

Ja uz ekrāna atrodas pirmā komandkarte, lietotājs var veikt datu bāzes apskati pa ierakstiem:

- nospiežot taustiņu →, uz ekrāna tiek izvadīts nākošā ieraksta saturs. Ja uz ekrāna jau atrodas pēdējais ieraksts, tad datu bāzes vadības sistēma uz taustiņa → nospiešanu nereaģē;
- nospiežot taustiņu ←, uz ekrāna izvada iepriekšējā ieraksta saturu. Ja uz ekrāna atrodas pirmais ieraksts, tad datu bāzes vadības sistēma uz taustiņa ← nospiešanu nereaģē;

- nospiežot taustiņu BC, uz ekrāna tiek izvadīts pirmā ieraksta saturs;
- nospiežot taustiņu KT, uz ekrāna tiek izvadīts pēdējā ieraksta saturs;
- nospiežot taustiņu IIIAΓ, uz ekrāna tiek izvadīts tā ieraksta saturs, kura numuru iegūst, pieskaitot pie kārtējā ieraksta numura 5, vai uz pēdējo ierakstu, ja ieraksta ar tik lielu numuru nav.

Nospiežot taustiņu ←, notiek pāreja no apskates pa ierakstiem uz datu bāzes apskati un labošanu pa laukiem. Instrukcijas logā tiek izvadīta komandkarte:

←	lauka labošana
← →	ieraksta izvēle
BC	bāzes sākums
KT	bāzes beigas
IIIAΓ	+5 ieraksti
↑ ↓	cits lauks
E	beigt

Ar taustiņu ← vai →, BC, KT un IIIAΓ palīdzību var izvēlēties interesējošo ierakstu. Ar taustiņu ↑ vai ↓ palīdzību var novietot kursoru uz interesējošā lauka numura.

Ja nepieciešama lauka vērtības nomaiņa, jānospiež taustiņš ←. Kursors pārvietojas uz lauka vērtības logu, un lietotājs var izmainīt lauka vērtību. Instrukcijas logā ir informācija par lauka atribūtiem. Jaunās vērtības piešķire notiek, nospiežot taustiņu ←. Par jauno lauka vērtību datu bāzes vadības sistēma uzskata visu vērtības logā redzamo informāciju, neatkarīgi no kursora atrašanās vietas. Ja vērtības nomaiņa izdarīta nekorekti, tad jaunā vērtība netiek akceptēta, līdz kļūda nav izlabota un atkārtoti nospiež taustiņš ←. Ja lauka vērtību datu bāzes vadības sistēma akceptē, tad pēc taustiņa ← nospiešanas uz ekrāna tiek izvadīts nākošā ieraksta saturs, kursoram atrodoties uz izlabotā lauka numura. Ja labotais lauks atrodas pēdējā ierakstā, tad pēc lauka vērtības nomaiņas uz ekrāna saglabājas šī ieraksta saturs.

Nospiežot taustiņu E, instrukcijas logā tiek izvadīta pirmā komandkarte. Iziet no datu bāzes aizpildes un labošanas režīma var tikai no pirmās komandkartes, tādēļ atkārtoti jānospiež taustiņš E. Datu bāzes vadības sistēma atgriežas izvēles režīmā - uz ekrāna pēc programmas pārlādes ir galvenā komandkarte.

Atgādinājums!

Veidojot un mainot lauku vērtības, darbojas visi datora БК rediģēšanas taustiņi.

Datu bāzē aizpildīto ierakstu skaitu norāda informācijas loga rekvizīts "Pilni ieraksti".

Ja izmaiņas veiktas datu bāzē, kas satur iezīmēto ierakstu kopu, tad tā tiek anulēta.

Datu bāzes aizpilde un labošana notiek datora БК atmiņā. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

4.2.4. Datu bāzes ierakstu apskate

Uzsākot datu bāzes ierakstu apskati, uz ekrāna tiek izvadīts pirmā ieraksta saturs. Kursors atrodas uz ieraksta numura. Lietotājs var izvēlēties: apskatīt datu bāzi pa ierakstiem (tabulas rindas) vai pāriet uz datu bāzes apskati pa laukiem (tabulas ailes), izmantojot sekojošus taustiņus:

- nospiežot taustiņu →, uz ekrāna tiek izvadīts nākošā ieraksta saturs. Ja uz ekrāna jau atrodas pēdējais ieraksts, tad datu bāzes vadības sistēma uz taustiņa → nospiešanu nereaģē;
- nospiežot taustiņu ←, uz ekrāna izvada iepriekšējā ieraksta saturu. Ja uz ekrāna atrodas pirmais ieraksts, tad datu bāzes vadības sistēma uz taustiņa ← nospiešanu nereaģē;
- nospiežot taustiņu BC, uz ekrāna tiek izvadīts pirmā ieraksta saturs;
- nospiežot taustiņu KT, uz ekrāna tiek izvadīts pēdējā ieraksta saturs;
- nospiežot taustiņu ШАГ, uz ekrāna tiek izvadīts tā ieraksta saturs, kura numuru iegūst, pieskaitot pie kārtējā ieraksta numura 5, vai pēdējais ieraksts, ja ieraksta ar tik lielu numuru nav.

Nospiežot taustiņu ←, notiek pāreja no apskates pa ierakstiem (tabulas rindām) uz datu bāzes apskati pa laukiem (tabulas ailēm):

- ar taustiņu ↑ vai ↓ palīdzību kursoru novieto uz apskatāmā lauka numura;
- nospiežot taustiņu BC, kursors tiek novieto uz pirmā lauka numura;
- nospiežot taustiņu KT, kursors tiek novieto uz pēdējā lauka numura;
- nospiežot taustiņu ШАГ, kursors pārvietojas uz tā lauka numura, kuru iegūst, pieskaitot pie kārtējā lauka numura 5. Ja lauka ar tik lielu numuru nav, tad kursors pārvietojas uz pēdējo lauku;
- atrodoties uz interesējošā lauka, nospiežot taustiņu → vai ←, var aplūkot šo lauku attiecīgi nākošajā vai iepriekšējā ierakstā (ja tāds ir);

- nospiežot taustiņu E, datu bāzes vadības sistēma pāriet no apskates pa laukiem uz datu bāzes apskati pa ierakstiem.

Ja taustiņu E nospiež laikā, kad notiek datu bāzes apskate pa ierakstiem, tad sistēma pāriet izvēles režīmā - pēc programmas pārlādes uz ekrāna ir galvenā komandkarte.

Lai apskatītu tikai iezīmēto ierakstu kopai (ja tāda ir) piederošos ierakstus, jānospiež taustiņš Z laikā, kad notiek datu bāzes apskate pa ierakstiem. Pēc taustiņa Z nospiešanas uz ekrāna tiek izvadīts iezīmētais ieraksts ar vismazāko numuru. Iezīmēto ierakstu kopā, analogi kā visā datu bāzē, apskati var organizēt gan pa ierakstiem, gan pa laukiem. Atgriešanās uz visu ierakstu apskati notiek, nospiežot taustiņu E.

Atgādinājums!

Apskatot datu bāzi pa ierakstiem, uz ekrāna izvada ieraksta saturu, sākot ar pirmo lauku. Ja ieraksta visu saturu nevar parādīt uz viena ekrāna, tad sekojošo ieraksta lauku apskatei jāizmanto datu bāzes apskate pa laukiem.

4.2.5. Datu bāzes saglabāšana

Saglabāšanas operācija sākas ar pārbaudi, vai disketē ir fails ar tādu pašu nosaukumu kā datu bāzei datora BK atmiņā.

Ja disketē nav faila ar tādu pašu nosaukumu kā datu bāzei datora atmiņā, tad instrukcijas logā redzams jautājums:

Mainīt bāzes nosaukumu? J/N

Ja atbild apstiprinoši (J), tad instrukcijas logā ziņo, ka jāievada datu bāzes nosaukums. Tiek atgādināts, ka nosaukuma pierakstā var lietot lielos latīņu burtus un ciparus, un tas nedrīkst sākties ar D. Veidojot nosaukumu, darbojas visi datora BK rediģēšanas taustiņi. Nosaukumu ievada, nospiežot taustiņu ←. Pēc nosaukuma nomaiņas saglabāšanas operācija atsākas no jauna.

Pēc N nospiešanas instrukcijas logā redzams jautājums:

Saglabāt datu bāzi? J/N

Pēc J nospiešanas datu bāze tiek rakstīta failā ar uzdoto nosaukumu un paplašinājumu BIN. Instrukcijas logā ziņo, ka notiek saglabāšana un jāgaida. Pēc veiksmīga ieraksta sistēma pāriet izvēles režīmā - uz ekrāna ir galvenā

komandkarte. Ja fails kaut kādu iemeslu dēļ (programmas NET režīms - ieraksts aizliegts, disketē nav vietas, tehnika nav darbspējīga) nav ierakstīts disketē, tad instrukcijas logā ziņo par kļūmi. Pēc taustiņa ← nospiešanas sistēma pāriet izvēles režīmā - uz ekrāna ir galvenā komandkarte.

Pēc N nospiešanas sistēma atgriežas izvēles režīmā - uz ekrāna galvenā komandkarte.

Ja disketē jau ir fails ar tādu pašu nosaukumu kā datu bāzei datora BK atmiņā, tad instrukcijas logā redzams ziņojums:

Disketē jau ir šāds fails! Turpinājumam nospieš ←.

Instrukcijas logā jautājums:

Mainīt bāzes nosaukumu? J/N

Turpmākās rīcības iespējas jau ir aprakstītas.

Atgādinājums!

Ja saglabāšanas kļūmes cēlonis ir vietas trūkums disketē, tad sistēma ieraksta disketē vienu bloku garu fiktīvu failu (ja ir vieta). Pēc vietas izbrīvēšanas disketē vai disketes nomaiņas saglabāšanas operācija jāatkārto. Ja ir nepieciešamība mainīt disketi, tad jāaptur tikla programma, jānomaina diskete un atkārtoti jāpalaiž tikla programma.

Minimālā apjoma datu bāzes faila garums ir 17 bloki.

4.2.6. Visu ierakstu dzēšana

Pēc operācijas izvēles instrukcijas logā jautājums:

Dzēst visus ierakstus? J/N

Pēc J nospiešanas datora atmiņā tiek dzēsta datu bāzes saturiskā informācija (ieraksti). Informācijas logā aizpildīto (pilno) ierakstu skaits kļūst 0. Tiek anulētas ziņas par iezīmētiem ierakstiem (dzēs to kopas nosaukumu un to skaits kļūst 0). Datu bāzes apraksts datora atmiņā saglabājas. Turpinājumā sistēma pāriet izvēles režīmā - uz ekrāna galvenā komandkarte.

Pēc N nospiešanas sistēma pāriet izvēles režīmā - uz ekrāna galvenā komandkarte.

Operāciju "visu ierakstu dzēšana" var izmantot, ja, aizpildot datu bāzi, konstatēts, ka definētā struktūra ir nepilnīga un nepieciešams to mainīt.

Atgādinājums!

Ierakstu izmešana notiek tikai datu bāzē datora БК atmiņā. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

4.2.7. Datu bāzes dzēšana

Pēc operācijas izvēles instrukciju logā jautājums:

Dzēst datu bāzi? J/N

Pēc J nospiešanas visa informācija datora BK atmiņā par datu bāzi tiek dzēsta (gan struktūras apraksts, gan ieraksti). Informācijas logā tiek anulētas visas ziņas par datu bāzi (dzēš bāzes un iezīmēto ierakstu kopas nosaukumus, lauku, pilno, kopā un iezīmēto ierakstu skaits kļūst 0). Turpinājumā sistēma pāriet izvēles režīmā - uz ekrāna galvenā komandskarte.

Pēc N nospiešanas sistēma pāriet izvēles režīmā - uz ekrāna galvenā komandskarte.

Atgādinājums!

Datu bāzes dzēšana notiek tikai datora БК atmiņā. Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

4.2.8. Statistiskā apstrāde

Statistiskā apstrāde sniedz ziņas par visu vai iezīmēto ierakstu skaitļu lauku mazāko un lielāko vērtību, vidējo aritmētisko, kā arī par amplitūdu (starpība starp lielāko un mazāko vērtību) un summu.

Logā ekrāna kreisajā pusē izvadīti datu bāzes skaitļu lauku numuri un nosaukumi. Ar taustiņu ↑ vai ↓ palīdzību kursoru novieto uz interesējošā lauka numura. Nospiežot taustiņu BC, kursors tiek novietots uz pirmā lauka numura. Nospiežot taustiņu KT, kursors tiek novietots uz pēdējā lauka numura. Nospiežot taustiņu ШАГ, kursors pārvietojas uz tā lauka numuru, kuru iegūst, pieskaitot pie kārtējā lauka numura 5. Ja lauka ar tik lielu numuru nav, tad kursors pārvietojas uz pēdējā lauka numuru.

Logā ekrāna vidusdaļā tiek norādīts izvēlētā lauka nosaukums un atribūti, tā mazākā un lielākā vērtība, kā arī vidējais aritmētiskais, amplitūda un summa. Zem statistiskajām vērtībām ir informācija, vai statistiskie lielumi iegūti, izmantojot visu vai tikai iezīmēto ierakstu lauku vērtības. Pāreja no visu ierakstu uz iezīmēto ierakstu statistisko apstrādi vai otrādi notiek, nospiežot taustiņu Z vai V.

Nospiežot taustiņu E, notiek datu bāzes vadības sistēmas atgriešanās izvēles režīmā - pēc programmas pārlādes uz ekrāna redzama galvenā komandkarte.

Atgādinājums!

Uzsākot statistisko apstrādi, uz ekrāna tiek izvadītas ziņas par ieraksta pirmā skaitļa tipa lauku, izmantojot visu ierakstu vērtības.

Ja datu bāzes ieraksti nesatur skaitļa tipa laukus, tad par to tiek ziņots informācijas logā. Turpinājumam jānospiež taustiņš ←. Datu bāzes vadības sistēma pāriet izvēles režīmā - pēc programmas pārlādes uz ekrāna galvenā komandkarte.

4.2.9. Ierakstu atlase (iezīmēšana)

Uzsākot ierakstu atlasī, lietotājam jāievada iezīmēto ierakstu kopas nosaukums. Nosaukuma pierakstā var tikt izmantota jebkura simbolu virkne, tai skaitā arī tukša. Datu bāze var saturēt tikai vienu iezīmēto ierakstu kopu, tāpēc, uzsākot ierakstu atlasī, esošā iezīmēto ierakstu kopa datora atmiņā tiek anulēta.

Attiecīgajā disketes datu bāzes failā (ja tāds ir) izmaiņu nav.

Pēc iezīmēto ierakstu kopas nosaukuma ievadīšanas lietotājam jāuzdod atlasē nosacījumi. Atlasē nosacījumi var būt gan vienkārša loģiska izteiksme, kas satur vienu apgalvojumu, gan salikta. Saliktu loģisku izteiksmi veido vienkāršas loģiskas izteiksmes, kas saistītas ar loģiskajām operācijām "UN" un "VAI". Operācijas "UN" prioritāte ir augstāka kā operācijas "VAI" prioritāte, tas nozīmē, ka "UN" piesaista operandus ciešāk nekā "VAI". Iekavu lietošana nav paredzēta.

Ar taustiņu ↑ vai ↓, BC, KT un ШАГ palīdzību novieto kursoru uz tā lauka numura, kura vērtību izmantos loģiskajā izteiksmē. Izvēles apstiprinājums notiek, nospiežot taustiņu ←. Uz ekrāna tiek atstāts tikai salīdzināmā lauka

nosaukums. Atkarībā no lauka tipa datu bāzes vadības sistēma piedāvā lietotājam salīdzināšanas operāciju.

Skaitļa tipa lauku vērtībām ir pieļautas 6 veidu salīdzināšanas ar skaitlisku konstanti. Tās ir: =; ≠; >; ≥; <; ≤. Operācijas izvēle notiek, lietotājam nospiežot cipara taustiņu, kurš komandkartē atrodas pie izvēlētās operācijas. Aiz lauka nosaukuma datu bāzes vadības sistēma izvada izvēlēto salīdzināšanas operāciju un logu paraugvērtības pierakstam. Lietotājam jāievada skaitliskā konstante, ar kuru tiks salīdzinātas lauku vērtības visos ierakstos.

Paraugvērtības pieraksta loga izmēri tiek izvadīti saskaņā ar salīdzināmā lauka aprakstu.

Teksta lauku vērtības var pārbaudīt, vai tās sakrīt vai nesakrīt ar paraugtekstu. Lietotājs var izvēlēties: atšķirt vai neatšķirt lielos un mazos burtus paraugtekstā un salīdzināmajā tekstā. Paraugtekstā simboli "?" un "" ir ar īpašu nozīmi. "?" nozīmē, ka salīdzināmā teksta lauka vērtības atbilstošā pozīcija var saturēt jebkuru zīmi, bet ar "" apzīmētā vietā salīdzināmā lauka vērtība var saturēt patvaļīgu zīmju virkni, ieskaitot tukšu.

Pēc paraugvērtības ievades datu bāzes vadības sistēma loga "Rezultāti" rekvizīts "Ir atrasti" rāda ierakstu skaitu, kas apmierina pēdējo ievadīto vienkāršo loģisko izteiksmi. Lietotājam ir iespēja turpināt atlases nosacījumu, izmantojot loģiskās operācijas "UN" vai "VAI", kā arī beigt nosacījuma veidošanu.

Ja tiek turpināta atlases nosacījuma veidošana, tad izvēlētās loģiskās operācijas nosaukums tiek izvadīts logā "Loģiskā operācija". Logā "Rezultāti" rekvizīts "Apgalvojuma Nr" palielinās par vienu, un tiek dzēsts iepriekšējās vienkāršās loģiskās izteiksmes apmierinošo ierakstu skaits. Sākas jaunas vienkāršas loģiskas izteiksmes veidošana ar salīdzināmā lauka izvēli.

Ja jauni apgalvojumi netiks pievienoti, tad pēc taustiņa E nospiešanas blakus iezīmēto ierakstu kopas nosaukumam redzams atlases rezultātā iegūto iezīmēto ierakstu skaits. Pievienot jaunus apgalvojumus vairs nevar. Informācijas logā redzama komandkarte, kurā lietotājam tiek piedāvātas sekojošas iespējas:

- | | |
|---|------------------|
| A | ierakstu apskate |
| M | sākt no jauna |
| E | beigt |

Izvēloties ierakstu apskati, iezīmēto ierakstu kopā var veikt apskati pa ierakstiem vai laukiem:

- nospiežot taustiņu →, uz ekrāna tiek izvadīts nākošā iezīmētā ieraksta saturs. Ja uz ekrāna jau atrodas pēdējais iezīmētais ieraksts, tad datu bāzes vadības sistēma uz taustiņa → nospiešanu nereaģē;
- nospiežot taustiņu ←, uz ekrāna izvada iepriekšējā iezīmētā ieraksta saturu. Ja uz ekrāna atrodas iezīmētais ieraksts ar vismazāko numuru, tad datu bāzes vadības sistēma uz taustiņa ← nospiešanu nereaģē;
- ar taustiņu ↑ vai ↓ palīdzību var apskatīt izvēlēto ierakstu pa laukiem;
- lai beigtu iezīmēto ierakstu apskati, jānospiež taustiņš E. Datu bāzes vadības sistēma atgriežas uz iepriekšējo komandkarti.

Nospiežot taustiņu M, datu bāzes vadības sistēma sāk jaunu augstāk aprakstīto atlasī ar iezīmēto ierakstu kopas nosaukuma ievadi. Iepriekšējā iezīmēto ierakstu kopa tiek anulēta.

Ierakstu atlases beigšanai lietotājam jānospiež taustiņš E. Pēc programmas pārlādes uz ekrāna redzama galvenā komandkarte.

Atgādinājums!

Ja datu bāze satur iezīmētu ierakstu kopu, tad, izvēloties ierakstu atlases režīmu, tā tiek anulēta.

Ierakstu iezīmēšana notiek tikai datora BK atmiņā. Lai saglabātu iezīmēto ierakstu kopu failā uz disketes, pēc ierakstu iezīmēšanas jāizpilda datu bāzes saglabāšanas operācija.

Veidojot iezīmēto ierakstu kopas nosaukumu un salīdzināmo lauku vērtības, darbojas visi datora BK rediģēšanas taustiņi. Ievads notiek, nospiežot taustiņu ←.

4.2.10. Pārskata veidošana

4.2.10.1. Pārskata teksta faila sagatavošana izvadei uz drukas iekārtas

Pārskata teksta faila sagatavošanas laikā datu bāzes vadības sistēma ļauj izvēlēties ierakstus un laukus pārskatam, kas ar īpašām programmām izvadāms no datora DBK gan ar D100, gan Robotron tipa drukas iekārtām. Lauku vērtības pārskatā tiek izvietotas pa ailēm, kas var tikt atdalītas ar

vertikālām svītrām. Kopējais pārskata platums nevar pārsniegt 128 pozīcijas. Par aizņemto pozīciju skaitu ziņo rekvizīts "Pārskata platums". Kā pirmā var būt izvadāmo ierakstu numuru aile. Otrā aile var saturēt norādi par ierakstu piederību iezīmēto ierakstu kopai. Pārskatā nevar izvadīt vairāk kā 12 ailes, neskaitot divas īpašās ailes.

Pārskata formēšana notiek dialogā:

Veidot vertikālās svītras? J/N

Pēc taustiņa J nospiešanas pārskata apraksta logā redzams ziņojums "Ir vertikālas svītras".

Ja nospiests taustiņš N, tad pārskata apraksta logā redzams ziņojums "Nav vertikālas svītras".

Izvadīt ierakstu numurus? J/N

Pēc taustiņa J nospiešanas pārskata apraksta logā redzams ziņojums "Ir numerācija", pārskata platums kļūst 6.

Ja nospiests taustiņš N, tad pārskata apraksta logā redzams ziņojums "Nav numerācija", atskaites platums kļūst 2.

Izvadīt visus ierakstus? J/N

Pēc taustiņa J nospiešanas pārskata apraksta logā izvadāmo ierakstu skaits kļūst vienāds ar aizpildīto ierakstu skaitu.

Ja nospiests taustiņš N un ja datu bāze satur iezīmētu ierakstu kopu, tad instrukcijas logā tiek izvadīts jautājums:

Izvadīt iezīmētos ierakstus? J/N

Pēc taustiņa J nospiešanas pārskata apraksta logā izvadāmo ierakstu skaits kļūst vienāds ar iezīmēto ierakstu skaitu.

Ja iezīmēto ierakstu nav, vai tika nospiests taustiņš N, tad dialoga turpinājums ir sekojošs:

Pirmais ieraksts?

Jāievada skaitlis robežās no 1 līdz pilno ierakstu skaitam (to apzīmē ar k). Ja k mazāks par pilno ierakstu skaitu, tad instrukcijas logā jautājums:

Pēdējais ieraksts?

Jāievada skaitlis robežās no k līdz pilno ierakstu skaitam.

Ja datu bāze satur iezīmēto ierakstu kopu, un netiek izvadīti tikai iezīmētie ieraksti, tad tiek jautāts:

Atzīmēt iezīmētos ierakstus? J/N

Ja nospiež taustiņu J, tad izvadāmajā pārskatā būs aile, kas saturēs norādi - simbolu "*" - par ieraksta piederību iezīmēto ierakstu kopai. Pārskata platums palielinās par 2.

Turpinājumā uz ekrāna tiek izvadīti lauku numuri un nosaukumi. Instrukcijas logā datu bāzes vadības sistēma izvada komandkarti:

↑ ↓	lauka izvēle
←	ailēs veidošana
BC	pirmais lauks
KT	pēdējais lauks
ШAГ	+5 lauki
E	beigt

Ar taustiņu ↑ vai ↓, BC, KT un ШAГ palīdzību lietotājs var novietot kursoru uz lauka numura, kura vērtības viņš vēlas izvadīt pārskatā. Izvēloties laukus, lietotājs norāda secību, kādā lauku vērtības tiks izvadītas pārskata ailēs. Izvēli apstiprina, nospiežot taustiņu ←. Turpinājumā datu bāzes vadības sistēma izvada izvēlētā lauka nosaukumu un ailes platumu vērtības izdrukai. Ja izvadāmā lauka garums ir mazāks par 12 simboliem, tad informācijas logā ziņo, ka pārskata ailes platumu lietotājs var palielināt līdz 12 simboliem (lauka nosaukuma garums). Ja izvadāmā lauka garums ir lielāks par 20 simboliem, tad informācijas logā ziņo par lauka garumu un robežām, kādās var palielināt sistēmas ieteikto ailes platumu - 20 simboli (lauka vērtības slejas platums uz ekrāna aizpildes un apskates režīmos). Ja atlikušajā atskaites platumā neietilpst izvēlētā lauka vērtības izvadaailes minimālais platums, tad informācijas logā datu bāzes vadības sistēma ziņo, ka lauks ir par garu un darba turpinājumam jānospiež taustiņš ←. Lai pabeigtu veidot pārskata ailes, lietotājam jānospiež taustiņš E. Pārskatu nevar beigt veidot, ja nav izvēlēts vismaz viens lauks.

Dialoga turpinājumā lietotājs var norādīt izvadāmo ierakstu kārtošanas principus. Ierakstu kārtošanai datu bāzes vadības sistēma izmanto tikai atskaitē izvadāmo lauku vērtības.

Sakārtot izvadāmos ierakstus? J/N

Ja nospiests taustiņš N, tad ieraksti atskaitē tiks izvadīti secībā, kādā tie glabājas datu bāzē.

Pēc taustiņa J nospiešanas informācijas logā sekojoša komandkarte:

↑ ↓ lauka izvēle
← prioritātes piešķiršana

Ar taustiņu ↑ vai ↓ palīdzību var izvēlēties lauku, pēc kura vērtībām tiks kārtoti ieraksti. Ja izvēlēts teksta tipa lauks, tad kārtošana notiek leksikogrāfiskā secībā. Ja izvēlēts skaitļa tipa lauks, tad lietotājam papildus jānorāda, vai kārtošana jāveic vērtību augošā vai dilstošā secībā. Ja kārtošana jāveic pēc vairāku lauku vērtībām, tad, atbildot uz jautājumu:

Vai vajag 2.prioritāti? J/N,

jānospiež taustiņš J.

Nākošo lauku, pēc kura vērtībām tiks kārtoti ieraksti, lietotājs var izvēlēties no kārtošana vēl neiesaistītiem laukiem. Šādā veidā kārtošana var iesaistīt visu izvadāmo lauku vērtības.

Ja nospiests taustiņš N, vai prioritātes jau ir piešķirtas, tad informācijas logā redzams jautājums:

Vai pārskatam vajag virsrakstu? J/N

Pēc taustiņa J vai N nospiešanas notiek programmas pārlāde, par ko ziņo informācijas logā. Ja tika nospiests taustiņš J, tad lietotājam jāievada pārskata virsraksts. Tas var aizņemt vienu rindu un tā garums nedrīkst pārsniegt 128 pozīcijas. Virsraksta veidošanas laikā darbojas visi datora БК rediģēšanas taustiņi. Virsraksta ievads notiek, nospiežot taustiņu ←. Pirms faila saglabāšanas disketē lietotājam ir iespēja rediģēt pārskata galviņu.

Lai saglabātu pārskata teksta failu disketē, lietotājam jāievada tā nosaukums. Saglabāšanas operācijas izpilde sākas ar analīzi, vai disketē ir fails ar ievadīto nosaukumu. Ja disketē ir fails ar šādu nosaukumu, tad sistēma par to ziņo. Turpinājumam jānospiež taustiņš ← un jānomaina faila nosaukums. Ja izvadāmās informācijas apjoms ir tik liels, ka viss pārskata teksts nevar ievietoties datora БК atmiņā (ekrāna atmiņas apgabalā), tad datu bāzes vadības sistēma veic teksta sagatavošanu pa daļām. Katra pārskata daļa disketē tiek saglabāta atsevišķi, norādot citu nosaukumu turpinājuma failam.

Pēc pārskata saglabāšanas (disketē faili tiek saglabāti ar paplašinājumu BIN) datu bāzes vadības sistēma turpina dialogu:

Sagatavot citu teksta failu? J/N

Pēc taustiņa J nospiešanas notiek programmas pārlāde, un datu bāzes vadības sistēma atsāk jauna pārskata teksta faila sagatavošanu iepriekš aprakstītajā veidā. Ja tika nospiests taustiņš N, tad pēc programmas pārlādes sistēma pāriet izvēles režīmā - uz ekrāna tiek izvadīta galvenā komandkarte.

4.2.10.2. Pārskata izdruka

Pārskata izdruku realizē īpaši programmlīdzekļi, kas darbojas autonomi (ārpus datu bāzes vadības sistēmas) un izmanto tikai datora ДБК vidi. Atkarībā no drukas iekārtas tipa D100 vai Robotron, jālieto attiecīgi programma D100L.SAV vai ROBL.SAV.

Programmas darbības laikā dialoga režīmā lietotājam jāievada diska iekārtas numurs, kurā atrodas diskete ar sagatavotajiem teksta failiem, un izvadāmā faila nosaukums, kā arī jānorāda izdrukas blīvums: normālais - drukājot 80 simbolus vienā rindiņā, vai saspieštais - drukājot 128 simbolus vienā rindiņā. Pārskatu, kas disketē glabājas vairākos failos, jāizdrukā saglabāšanas secībā. Ja disketē nav faila ar ievadīto nosaukumu, tad programma par to ziņo un beidz darbu. Drukājot pārskatu, kas novietots vairākos failos, izdrukas programma jādarbina ik reizes no jauna. Ja pārskata aiļu kopējais platums pārsniedz 80 pozīcijas un izvēlēts normāls izdrukas blīvums, tad pārskata rindas tiek lauztas.

Atgādinājums!

Tā kā drukas iekārtas Robotron tehniskās iespējas nepieļauj vienlaicīgi izvadīt gan latīņu, gan krievu alfabētā veidotus tekstus, tad programma ROBL.SAV realizē pārskata teksta izdruku tikai latviešu alfabētā.

4.3. Datu bāzes vadības sistēmas lietošana

4.3.1. Uzdevuma "Ārstniecības augi" apraksts

Uzdevums ir izveidot datu bāzi, kas satur ziņas par ārstniecības augu drogām un to lietošanu.

4.3.1.1. Datu bāzes struktūras plānošana

Jaunas datu bāzes veidošana jāsāk ar tās plānošanu. Jānoskaidro, kādas ziņas datu bāzei jāsaturs, kādi ir iespējamie pieprasījumi ziņu iegūšanai no datu bāzes. Tas noteiks ieraksta struktūru un lauku atribūtus. Jāpievērš uzmanība tam, lai informācija nebūtu pārāk sadrumstalota, kā arī, lai viens lauks nesaturētu savstarpēji mazsaistītu informāciju. No lauku skaita un garumiem atkarīgs iespējamo ierakstu skaits datu bāzē.

Viens no ārstniecības augu datu bāzes plānojumiem varētu būt šāds:

- lauks "Nosaukums" (teksta tipa lauks, lauka garums 20 simboli) saturēs ārstniecības auga nosaukumu;
- lauks "Drogas veids" (teksta tipa lauks, lauka garums 18 simboli) saturēs no ārstniecības auga izgatavojamo drogas veidu;
- lauks "Lieto" (teksta tipa lauks, lauka garums 100 simboli) saturēs drogas farmakoterapeitiskās iedarbības īsu aprakstu;
- lauks "Ievākšana" (teksta tipa lauks, lauka garums 12 simboli) saturēs auga ievākšanas kalendāru (ievākšanas kalendāru var norādīt dažādos veidos - norādot ievākšanas mēnešus vai dažādi tos kodējot). Šajā gadījumā teksta lauka 12 simboli norādīs pēc kārtas sekojošus 12 mēnešus. Simbols "+" nozīmēs, ka šajā mēnesī augi ievācam, "-" nozīmēs, ka šajā mēnesī augi nav derīgi ievākšanai;
- lauks "Nožuvums(%)" (skaitļa tipa lauks, lauka garums 2, decimālās daļas garums 0 simboli) saturēs ārstniecības auga nožuvumu procentos;
- lauks "Iepērk" (skaitļa tipa lauks, lauka garums 5, decimālās daļas garums 2 simboli) saturēs drogas aptiekas iepirkuma cenu par kilogramu;
- lauks "Pārdod" (skaitļa tipa lauks, lauka garums 5, decimālās daļas garums 2 simboli) saturēs augu drogu kilograma pārdošanas cenu.

Ja datu bāzes struktūra ir izplānota, tad var ielādēt datorā BK datu bāzes vadības sistēmu un no galvenās komandkartes izvēlēties režīmu "Datu bāzes aprakste", nospiežot taustiņu 1.

4.3.1.2. Datu bāzes "AUGI" aprakste

Jaunas datu bāzes apraksta veidošana sākas ar datu bāzes nosaukuma ievadišanu. Kursors atrodas bāzes nosaukuma logā. Instrukcijas logā redzama norāde, ka nosaukuma pierakstā drīkst lietot lielos latīņu alfabēta burtus un tas

nedrīkst sākties ar D. Ievada datu bāzes nosaukumu AUGI un nospiež taustiņu ←. Kursors pārvietojas uz pirmā lauka nosaukuma logu. Lietotājam pakāpeniski jāveido lauka apraksts, ievadot lauka nosaukumu, tipu un garumu.

Lai iegūtu latviešu alfabēta burtus ar diakritiskajām zīmēm, vienlaicīgi jānospiež taustiņš AP2 un attiecīgā burta taustiņš. Teksta un vērtību ievades laikā darbojas visi datora rediģēšanas taustiņi.

Pirmā lauka apraksta ievadam jānospiež šādi taustiņi:

Nosaukums ← T ← 20 ←

Ja lauka aprakstu datu bāzes vadības sistēma akceptē, tad uz ekrāna tiek izvadīti tukši nākošā lauka apraksta logi, kursoram atrodoties uz lauka numura.

Instrukcijas logā redzama šāda komandkarte:

↑ ↓	lauka izvēle
BC	pirmais lauks
KT	pēdējais lauks
ИИAG	+9 lauki
←	lauka veidošana
I	iestarpināt lauku
D	izmest lauku
CBP	nomainīt lauku
E	beigt

Lai ievadītu otrā lauka aprakstu, jānospiež taustiņš ←. Kursors pārvietojas uz lauka nosaukuma logu, un lietotājam izplānotās datu bāzes struktūras veidošanai jānospiež šādi taustiņi (kombināciju AP2 + burta taustiņš turpmāk arī sauksim par taustiņu):

Drogas veids ← T ← 18 ←

←

Lieto ← T ← 100 ←

←

Ievākšana ← T ← 12 ←

←

Nožuvums (%) ← S ← 2 ←

←

Iepērk ← S ← 5 ← 2 ←

←

Pārdod ← S ← 5 ← 2 ←

Datu bāzes apraksta veidošanas laikā informācijas logā sistēma ziņo par izveidoto lauku skaitu un iespējamo ierakstu skaitu datu bāzē ar esošo struktūru.

Ja nepieciešami papildinājumi vai labojumi ievadītajā struktūrā, tad tos var veikt ar komandkartē piedāvāto iespēju palīdzību.

Ja bāzes apraksts izveidots, tad jānospiež taustiņš E, un datu bāzes vadības sistēma pēc programmas pārlādes atgriežas uz galveno komandkarti.

Ja darbs ar datu bāzi netiks turpināts (to tagad neaizpildīs), tad datu bāzes aprakstu var saglabāt disketē. No galvenās komandkartes jāizvēlas datu bāzes saglabāšanas režīms, nospiežot taustiņu 4.

4.3.1.3. Datu bāzes "AUGI" aizpilde

Lai aizpildītu datu bāzi, lietotājam no galvenās komandkartes jāizvēlas režīms "Bāzes aizpilde un labošana", nospiežot taustiņu 2.

Datu bāzes aizpildei izmantotas ziņas no H.Rubenes, S.Ozolas, V.Eniņa grāmatas "Ārstniecības augu sagatavošana un lietošana" un instrukcijas par augu iepirkšanu un pārdošanu Latvijas aptiekās.

Uzsākot aizpildīt datu bāzi, kura nesatur nevienu aizpildītu ierakstu, uz ekrāna tiek izvadīti lauku nosaukumi un lauku vērtību logi. Kursors atrodas pirmā lauka vērtības logā. Vērtības jāpiešķir visiem ieraksta laukiem pēc kārtas. Datu bāzes vadības sistēma nepieļauj tukšas lauku vērtības. Instrukcijas logā ziņo par aizpildāmā lauka tipu. Pirmā ieraksta aizpildei jānospiež šādi taustiņi:

kumelīte←

zieds←

nervu nomierināšanai, pret sāpēm, kā sviedrējošu līdzekli, pret iekaisumiem, kompresēm←

(Aizpildot teksta laukus, kas garāki par 20 simboliem, der atcerēties, ka pārskatā atbilstošo ailu platumu var mainīt no 20 līdz maksimāli iespējamajam ailes platumam.)

----++++-----←

75←

1←

2.6←

Pēc vērtības piešķires pēdējam laukam pilno ierakstu skaits informācijas logā par vienu palielinās. Lai aizpildītu nākošo ierakstu, lietotājam jānospiež taustiņš ←. Ieraksta numurs par vienu palielinās, uz ekrāna tiek izvadīti tukši lauku vērtību logi. Lietotājam jāaizpilda nākošais ieraksts, nospiežot šādus taustiņus:

mātere←

laksts←

sirds un asinsvadu ārstēšanai, asinsspiediena pazemināšanai, nervu nomierināšanai←

-----+++-----←

75←

.6←

1.5←

Nākošais ieraksts tiks aizpildīts ar ziņām par asinszāli. Ja grib ievietot to kā pirmo ierakstu, tad jānospiež taustiņš E. Informācijas logā redzama pirmā komandkarte, no kuras jāizvēlas režīms "darbs ar bāzi", nospiežot taustiņu B. Ar taustiņa ← palīdzību panāk, ka uz ekrāna tiek izvadīts pirmais ieraksts. Nospiežot taustiņu I, kārtējais un tam sekojošie ieraksti tiek pārnumurēti, atbrīvojot vietu iestarpināmajam ierakstam. Uz ekrāna tiek izvadīti lauku vērtību logi, kursoram atrodoties pirmā lauka vērtības logā. Lietotājam jāaizpilda šis ieraksts, nospiežot sekojošus taustiņus:

asinszāle←

laksts←

pret caureju, kā pretmikrobionālu un antiparazitāru līdzekli, kompresēm←

-----+++-----←

78←

1.2←

3.2←

Lai turpinātu ierakstu pievienošanu, no komandkartes jāizvēlas režīms "Veidot ierakstus", nospiežot taustiņu V. Uz ekrāna tiek izvadīti lauku vērtību logi. Pēc vērtību piešķires pēdējam laukam lietotājam jāizvēlas - turpināt ierakstu pievienošanu, nospiežot taustiņu ←, vai to beigt, nospiežot taustiņu E.

Ja jauni ieraksti netiks pievienoti, lietotājs var veikt aizpildītās datu bāzes apskati un labošanu.

Atgriešanās uz galveno komandkarti notiek no pirmās komandkartes, nospiežot taustiņu E.

4.3.1.4. Datu bāzes apraksta mainīšana

Ja, aizpildot datu bāzi, ir secināts, ka tās struktūra ir nepilnīga, piemēram, lauki par īsiem, tad to var mainīt šādi:

- atgriežas uz galveno komandkarti;
- izvēlas režīmu "Visu ierakstu dzēšana", nospiežot taustiņu 5;

uz jautājumu

Dzēst visus ierakstus? J/N

atbild apstiprinoši, nospiežot taustiņu J;

- izvēlas režīmu "Datu bāzes aprakste", nospiežot taustiņu 2. Uz ekrāna tiek izvadīta datu bāzes struktūra, un lietotājs ar komandkartē piedāvātajiem līdzekļiem var to mainīt.

4.3.1.5. Ziņu iegūšana no datu bāzes "AUGI"

Izmantojot datu bāzi "AUGI", atrast atbildes uz šādiem jautājumiem:

1. Kuriem ārstniecības augiem ievācam ziedi?
2. Kurus augus var ievākt vasarā?
3. Kuru augu drogas pārdod visdārgāk?
4. Kuru augu ziedus iepērk vislētāk?
5. Kā sauc nervus nomierinošos augus, kuru drogās neizmanto ne ziedus, ne augļus?
6. Kuras pretsāpju drogas ievāc tikai vasarā?
7. Vai ir drogas, kuras ievāc vismaz trīs mēnešus gadā?
8. Kuras drogas ievāc vismaz četrus mēnešus pēc kārtas?
9. Kuru drogu iepirkuma cena atrodas robežās no Ls .60 līdz Ls 1.40?
10. Kuriem augiem, kuru drogās neizmanto augļus, nožuvuma procents nav lielāks par 3/4 no augu sākotnējā svara?

1. Kuriem ārstniecības augiem ievācami ziedi?

Lai atlasītu nosacījumam atbilstošos ierakstus, jāatrod ieraksti, kuriem lauka "Drogas veids" vērtība satur vārdu "zieds" vai "ziedi". Tā kā bez vārda "zieds" laukā var būt arī citi vārdi (citi drogu veidi), tad paraugvērtība veidojama šādi: *zied*.

Darbības shēma:

1. Ievada iezīmēto ierakstu kopas nosaukumu.
2. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Drogas veids" numura un nospiež taustiņu ←.
3. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
4. Tā kā šajā gadījumā, salīdzinot vērtības, nav būtiski ievērot lielos un mazos burtus, tad nospiež taustiņu N.
5. Paraugvērtības logā ieraksta *zied* un nospiež taustiņu ←.
6. Beidz nosacījuma veidošanu, nospiežot taustiņu E.
7. Izvēlas iezīmēto ierakstu apskati, nospiežot taustiņu A.
8. Nolasa lauka "Nosaukums" vērtības ierakstos. Vispirms ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Nosaukums" numura, tad ar taustiņu ← → palīdzību pārvietojas pa ierakstiem.

2. Kurus augus var ievākt vasarā?

Pārveido nosacījumu "var ievākt vasarā" datu bāzes vadības sistēmai saprotamā loģiskas izteiksmes formā. "Var ievākt vasarā" nozīmē to, ka "var ievākt vismaz vienā no vasaras mēnešiem", kas nozīmē, ka "var ievākt jūnijā" VAI "var ievākt jūlijā" VAI "var ievākt augustā". Pārveidotais nosacījums neizslēdz iespēju, ka augi derīgi ievākšanai arī citos mēnešos, kā arī visos trīs vasaras mēnešos vai jebkuros divos no tiem. Apgalvojums "var ievākt jūnijā" jāpiemēro lauka "Ievākšana" vērtībai (teksta tipa lauks ar garumu 12, kas reprezentē pēc kārtas sekojošus 12 mēnešus; simbols "+" kādā pozīcijā nozīmē, ka attiecīgajā mēnesī augus var ievākt, "-" nozīmē, ka šajā mēnesī augi nav derīgi ievākšanai). Apgalvojuma patiesuma vērtības (paties, aplams) noteikšanai var izmantot paraugvērtību ??????+*. Pirmie pieci simboli "?" un simbols "***", kas aizstāj sešu simbolu "?" virkni, saturīgi izsaka, ka apgalvojuma patiesuma vērtību neiespaido augu ievākšana no janvāra līdz maijam un no

jūlija līdz decembrim. Simbols "+" paraugvērtības sestajā pozīcijā ir būtisks apgalvojuma "var ievākt jūnijā" patiesuma vērtības noteikšanā. Analogi uzraksta pārējo apgalvojumu pārbaudei atbilstošās paraugvērtības:

"var ievākt jūlijā" paraugvērtība ??????+*,

"var ievākt augustā" paraugvērtība ??????+*.

Darbības shēma:

1. Ievada iezīmēto ierakstu kopas nosaukumu.
2. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Ievākšana" numura un nospiež taustiņu ←.
3. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
4. Tā kā šajā gadījumā burti netiks salīdzināti, tad nospiež taustiņu N.
5. Paraugvērtības logā ieraksta ??????+* un nospiež taustiņu ←.
6. Turpina veidot nosacījumu ar loģisko operāciju VAI, nospiežot taustiņu 2.
7. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Ievākšana" numura un nospiež taustiņu ←.
8. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
9. Tā kā šajā gadījumā burti netiks salīdzināti, tad nospiež taustiņu N.
10. Paraugvērtības logā ieraksta ??????+* un nospiež taustiņu ←.
11. Turpina veidot nosacījumu ar loģisko operāciju VAI, nospiežot taustiņu 2.
12. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Ievākšana" numura un nospiež taustiņu ←.
13. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
14. Tā kā šajā gadījumā burti netiks salīdzināti, tad nospiež taustiņu N.
15. Paraugvērtības logā ieraksta ??????+* un nospiež taustiņu ←.
16. Beidz veidot nosacījumu, nospiežot taustiņu E.
17. Izvēlas iezīmēto ierakstu apskati, nospiežot taustiņu A.
18. Nolasa lauka "Nosaukums" vērtības ierakstus. Vispirms ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Nosaukums" numura, tad ar taustiņu ← → palīdzību pārvietojas pa ierakstiem. To, cik ierakstu apmierina katru atsevišķo ievadīto apgalvojumu, rāda loga "Rezultāti" rekvizīts "Ir atrasti". Ierakstu skaitu, kas apmierina visu salikto

nosacījumu, rāda loga "Iezīmēti ieraksti" rekvizīts "Skaitis" tikai pēc visa nosacījuma ievadišanas (taustiņa E nospiešanas).

3. Kuru augu drogas pārdod visdārgāk?

Lai atbildētu uz jautājumu, jānoskaidro, pirmkārt, kāda ir visaugstākā pārdošanas cena, otrkārt, kurām drogām tā atbilst. Visaugstāko pārdošanas cenu rāda rakvizīts "Lielākā vērtība", veicot visu ierakstu statistisko apstrādi laukam "Pārdod". Atbilstošo drogu nosaukumus iegūst, atlasot ierakstus, kuriem lauka "Pārdod" vērtība ir vienāda ar iepriekš atrasto.

Darbības shēma:

1. No galvenās komandkartes izvēlas statistisko apstrādi, nospiežot taustiņu 7.
2. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Pārdod" numura.
3. Nolasa un pieraksta vislielāko pārdošanas cenu, kura redzama logā "Lielākā vērtība".
4. Atgriežas uz galveno komandkarti, nospiežot taustiņu E.
5. No galvenās komandkartes izvēlas ierakstu atlasī, nospiežot taustiņu 8.
6. Ievada iezīmēto ierakstu kopas nosaukumu.
7. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Pārdod" numura un nospiež taustiņu ←.
8. No komandkartes izvēlas salīdzināšanas operāciju = (vienāds), nospiežot taustiņu 1.
9. Paraugvērtības logā ieraksta iepriekš nolasīto un pierakstīto skaitli un nospiež taustiņu ←.
10. Beidz veidot nosacījumu, nospiežot taustiņu E.
11. Izvēlas iezīmēto ierakstu apskati, nospiežot taustiņu A.
12. Nolasa lauka "Nosaukums" vērtības ierakstos. Vispirms ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Nosaukums" numura, tad ar taustiņu ← → palīdzību pārvietojas pa ierakstiem.

4. Kuru augu ziedus iepērk vislētāk?

Lai iegūtu atbildi uz jautājumu, pirmkārt, no visiem augiem jāatlasa tie, kuriem ievācami ziedi, otrkārt, iezīmēto ierakstu kopā jānolasa lauka "Iepērk" statistiskās apstrādes rezultātā iegūtā mazākā vērtība, treškārt, jāizdara to

ierakstu atlase, kuriem ievācamā ziedi un iepirkšanas cena ir viszemākā.

Atlases rezultātā iegūtais iezīmēto ierakstu skaits norāda, cik prasīto augu ir, bet, nolasot lauka "Nosaukums" vērtības, var uzzināt, kuri tie ir.

Darbības shēma:

1. Ievada iezīmēto ierakstu kopas nosaukumu.
2. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Drogas veids" numura un nospiež taustiņu ←.
3. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
4. Tā kā šajā gadījumā, salīdzinot vērtības, nav būtiski ievērot lielos un mazos burtus, tad nospiež taustiņu N.
5. Paraugvērtības logā ieraksta *zied* un nospiež taustiņu ←.
6. Beidz nosacījuma veidošanu, nospiežot taustiņu E.
7. No galvenās komandkartes izvēlas statistisko apstrādi, nospiežot taustiņu 7.
8. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Iepērk" numura.
9. Nolasa un pieraksta vismazāko pārdošanas cenu, kura redzama logā "Mazākā vērtība".
10. Atgriežas uz galveno komandkarti, nospiežot taustiņu E.
11. No galvenās komandkartes izvēlas ierakstu atlasī, nospiežot taustiņu 8.
12. Ievada iezīmēto ierakstu kopas nosaukumu (izvēloties ierakstu atlases režīmu, iepriekšējā iezīmēto ierakstu kopa tiek dzēsta).
13. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Drogas veids" numura un nospiež taustiņu ←.
14. No komandkartes izvēlas salīdzināšanas operāciju = (sakrīt ar paraugvērtību), nospiežot taustiņu 1.
15. Tā kā šajā gadījumā, salīdzinot vērtības, nav būtiski ievērot lielos un mazos burtus, tad nospiež taustiņu N.
16. Paraugvērtības logā ieraksta *zied* un nospiež taustiņu ←.
17. Turpina veidot nosacījumu ar loģisko operāciju UN, nospiežot taustiņu 2.
18. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Iepērk" numura un nospiež taustiņu ←.
19. No komandkartes izvēlas salīdzināšanas operāciju = (vienāds), nospiežot taustiņu 1.

20. Paraugvērtības logā ieraksta iepriekš nolasīto un pierakstīto skaitli un nospiež taustiņu ←.
21. Beidz veidot nosacījumu, nospiežot taustiņu E.
22. Izvēlas iezīmēto ierakstu apskati, nospiežot taustiņu A.
23. Nolasa lauka "Nosaukums" vērtības ierakstos. Vispirms ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Nosaukums" numura, tad ar taustiņu ← → palīdzību pārvietojas pa ierakstiem.

5. Kā sauc nervus nomierinošus augus, kuru drogās neizmanto ne ziedus, ne augļus?

Lai atbildētu uz jautājumu, jāatlasa ieraksti, kuros minētās drogas lieto nervu nomierināšanai un kuru drogas veids nav zieds vai auglis. Iezīmēto ierakstu kopā jānolasa lauka "Nosaukums" vērtības. Tā kā loģiskā operācija UN piesaista operandus (apgalvojumus) ciešāk kā loģiskā operācija VAI, tad ierakstu atlasei jāizmanto šāds nosacījums:

"Lieto" = *nervu nomierināšanai* UN "Drogas veids" ≠ *zied*
VAI
"Lieto" = *nervu nomierināšanai* UN "Drogas veids" ≠ *aug*

6. Kuras pretsāpju drogas ievācamas tikai vasarā?

Lai atbildētu uz jautājumu, jāatlasa tie ieraksti, kuros minētās drogas lieto pret sāpēm un ir ievācamas tikai vasaras mēnešos. Nosacījuma "ievācamas tikai vasarā" pārbaudes atbilstošā paraugvērtība: ----???----, jo "ievācamas tikai vasarā" nozīmē, ka, izņemot jūniju, jūliju un augustu, augus neievāc. Lai gan teorētiski pastāv iespēja, ka nosacījumam atbilstošajam ierakstam arī visu trīs simbolu "?" vietā varētu būt simbols "-", tomēr jāsaprot, ka nav tāda ārstniecības augs, kuru neievāc nekad.

7. Vai ir drogas, kuras ievāc vismaz trīs mēnešus gadā?

Nosacījums "vismaz trīs mēnešus gadā" nozīmē, ka lauka "Ievākšana" vērtībai jāsaturs vismaz trīs simboli "+". Tā kā jautājumā nav norādīti konkrēti mēneši, tad nosacījuma pārbaudei izmantojamā paraugvērtība: *+*+*+*.

8. Kuras drogas ievāc vismaz četrus mēnešus pēc kārtas?

Atšķirībā no iepriekšējā jautājuma, šajā gadījumā lauka "Ievākšana" vērtībai jā satur četri pēc kārtas sekojoši simboli "+". Atlasei izmantojamais nosacījums ir šāds: *++++* VAI +*+++ VAI ++*++ VAI +++*+.

9. Kuru drogu iepirkuma cenas atrodas robežās no Ls .60 līdz Ls 1.40?

Drogu iepirkuma cena atrodas robežās no .60 līdz 1.40 - nozīmē:

$.60 \leq \text{"Iepērk"} \leq 1.40$, tas ir,
 $\text{"Iepērk"} \geq .60$ UN $\text{"Iepērk"} \leq 1.40$.

10. Kuriem augiem, kuru drogās neizmanto augļus, nožuvuma procents nav lielāks par 3/4 no augu sākotnējā svara?

Lai atbildētu uz jautājumu jāatlasa tie ieraksti, kuriem:

"Drogas veids" \neq *aug* UN "Nožuvums(%)" ≤ 75 .

Atlases rezultātā iegūtās iezīmēto ierakstu kopas skaits dos atbildi uz jautājumu "cik", bet lauka "Nosaukums" vērtības - kuri ir prasītie augi.

5.3.1.6. Pārskatu sagatavošana, izmantojot datu bāzi "AUGI"

Uzdevums: Izveidot pārskatu, kas satur ziņas par aptiekā pārdošanā esošajiem augiem, tā, lai tie būtu sakārtoti pārdošanas cenu dilšanas secībā.

Pieņem, ka pārskatā izvadāmas šādu lauku vērtības: "Augu nosaukums", "Lieto" un "Pārdod".

Pārskata veidošanas darbības shēma:

1. Izvēlas no galvenās komandkartes pārskata veidošanas režīmu, nospiežot taustiņu 9.
2. Uz jautājumu "Izvadīt ierakstu numurus? J/N" atbild apstiprinoši, nospiežot taustiņu J.
3. Uz jautājumu "Veidot vertikālas svītras? J/N" atbild apstiprinoši, nospiežot taustiņu J.

4. Uz jautājumu "Izvadīt visus ierakstus? J/N" atbild apstiprinoši, nospiežot taustiņu J.
5. Uz jautājumu "Atzīmēt iezīmētos ierakstus? J/N" atbild noliedzoši, nospiežot taustiņu N.
6. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Nosaukums" numura un nospiež taustiņu ←.
7. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Lieta" numura un nospiež taustiņu ←.
8. Tā kā lauka "Lieta" garums pārsniedz 20 simbolus, tad lietotājs var norādīt pārskatā izvadāmās ailes platumu instrukcijas logā redzamajās robežās. Šajā gadījumā ailes platums jāizvēlas tā, lai paliktu vieta lauka "Pārdod" vērtībām (lauka "Pārdod" platums saskaņā ar datu bāzes aprakstu 5; pārskata maksimālais kopējais platums 128). Ievada izvēlēto ailes platumu 40 un nospiež taustiņu ←.
9. Ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Pārdod" numura un nospiež taustiņu ←.
10. Tā kā lauka "Pārdod" garums 5 simboli, un līdz atskaites maksimālajam platumam šajā gadījumā ir vairāk nekā 12 simboli, tad izvadāmās ailes platumu var palielināt. Ievada ailes platumu 10 un nospiež taustiņu ←.
11. Beidz veidot atskaites ailes, nospiežot taustiņu E.
12. Uz jautājumu "Sakārtot izvadāmos ierakstus? J/N" atbild apstiprinoši, nospiežot taustiņu J.
13. Prioritātes piešķiršanai ar taustiņu ↑ un ↓ palīdzību novieto kursoru uz lauka "Pārdod" numura un nospiež taustiņu ←.
14. Uz jautājumu "Augošā kārtībā? J/N" atbild noliedzoši, nospiežot taustiņu N.
15. Uz jautājumu "Vai vajag 2.prioritāti? J/N" atbild noliedzoši, nospiežot taustiņu N.
16. Uz jautājumu "Vai pārskatam vajag virsrakstu? J/N" atbild apstiprinoši, nospiežot taustiņu J.
17. Ievada atskaites virsrakstu:

Šodien aptiekā pārdošanā šādas drogas

un nospiež taustiņu ←.
18. Ieraksta nepieciešamās izmaiņas pārskata ailu nosaukumos un nospiež taustiņu ←.
19. Ievada teksta faila nosaukumu.

20. Uz jautājumu "Sagatavot citu teksta failu? J/N" atbild noliedzoši, nospiežot taustiņu N.

Atskaiti izdrukā no skaitļotāja ДБК ar programmu D100L.SAV ar drukas iekārtu D100 vai ROBL.SAV ar drukas iekārtu Robotron.

Nobeigums

Darbā ir dots pārskats par vispārizglītojošās vidusskolas informātikas pamatkursa saturu un dažu tēmu programmnodrošinājumu, kuru autors ir izstrādājis viens, vai ir izstrādē piedalījies kā līdzautors. Protams, kursa programmnodrošinājumu bez autora un līdzautoru kolektīva ir izstrādājuši arī citi gan Latvijā, gan aiz tās robežām. Šo visu izstrāžu rezultātā ir iespējams Latvijas vispārizglītojošajās vidusskolās pilnībā realizēt vidējās izglītības standarta prasības informātikā.

Darba tiešu turpinājumu ir vēlams izvērst vismaz divos virzienos:

- programmnodrošinājuma atlase un izstrāde, kā arī mācību grāmatu un metodisku līdzekļu sagatavošana informātikas pamatkursa realizācijai IBM un IBM saderīgu datoru vidē;
- informātikas profilkursa programmu (satura), programmnodrošinājuma atlase un izstrāde, kā arī mācību grāmatu un metodisku līdzekļu sagatavošana.

Publicēto darbu saraksts

Grāmatas

1. Витиньш М. Системы линейных уравнений. Рига: ЛГУ им.П.Стучки. 1978. 20 с.
2. Витиньш М., Кикуст П. Решение задач на повторение. Рига: ЛГУ им.П.Стучки. 1986. 74 с.
3. Ершов А., Монахов, В.Витиньш М. и др. Изучение основ информатики и вычислительной техники. Часть II. М.: Просвещение. 1986. 207 с.
4. Ершов А., Монахов В.Витиньш М. и др. Студия базелор информатичий ши але техничий компутационале. Партя а 2-а. Кишинэу: Лумина. 1987. 208 п.
5. Витиньш М., Опмане И., Павлов С., Фрейвалд Р. Изучение курса "Основы информатики и вычислительной техники". Рига: ЛГУ им.П.Стучки. 1988. 98 с.
6. Jeršovas A., Monachovas V., Vitinis M. ir kt. Informatikos ir skaičiavimo technikos pagrindų mokymas. II dalis. Kaunas: Šviesa. 1988. 246 p.
7. Ozoliņa L., Raudis A., Vitiņš M. Mācību datu bāze. Lietošanas apraksts. Rīga: Latvijas Universitāte. 1990. 55 lpp.

Raksti

1. Витиньш М. Некоторые вопросы разработки обучающей системы. В сб.: Разработка автоматизированных систем управления в Латвийской ССР. Рига: ЛГУ им.П.Стучки. 1977. С.158-161.
2. Витиньш М. Распознавание обучающей системой равносильности систем линейных уравнений с одинаковыми переменными. В сб.: Углубленное изучение математики и ее приложений. М.: НИИ СиМО АПН СССР. 1977. С.33-36.
3. Витиньш М. Автоматизация обучения решению систем линейных уравнений. В сб.: Методика создания математического обеспечения ЭВМ. Рига: ЛГУ им.П.Стучки. 1980. С. 96-103.

4. Витиņш М. Электронно-вычислительная машина в учебном процессе. В сб.: Аннотированная тематика лекций по естественно-научным знаниям. Рига: Общество "Знание" ЛатвССР. 1980. С.80-85.
5. Vitins M. Ueber die Arbeit des Laboratoriums fuer Schulinformatik an der Lettischen Peter-Stutschka-Universitaet. Zeitschr."Rechentechnik Datenverarbeitung". 1986. B.9. S.13-14.
6. Vitiņš M. Vārds programmētājam. Laikr."Skolotāju avīze". 1987. Nr.5. 4 lpp.
7. Витиņш М. Обобщенный алгоритм метода Гаусса для применения в обучающей программе. В сб.: ЭВМ в образовании. Программное обеспечение. Рига: ЛГУ им.П.Стучки. 1988. С.138-147.
8. Витиņш М. Пакет программ "Задачи на повторение". В кн.: Пакет обучающих программ для микроЭВМ "Электроника БК-0010". Рига: ЛГУ им.П.Стучки. 1988. С.5.
9. Andžāns A., Vitiņš M. Kā pierādīt algoritma pareizību? Laikr."Skolotāju avīze". 1988. Nr.41. 4 lpp.
10. Витиņш М., Озолия Л., Раудис А. Учебная система управления базой данных для КУВТ-86. В сб.: ЭВМ в образовании. Педагогические программные средства. Рига: ЛГУ им.П.Стучки. 1989. С.124-128.
11. Andersone L., Vitiņš M. Programma vispārizglītojošām vidusskolām. Gr.: Informātikas programmas. Rīga: TIM. 1990. Lpp.7-13.
12. Andersone L., Vitiņš M., Traidāss A. Programma arodvidusskolām. Informātika un ražošanas automatizācija. Rīga: TIM. 1990. Lpp.27-33.
13. Vitiņš M. Vidējo mācību iestāžu 6.informātikas olimpiādes norise un rezultāti. Laikr."IAI Palīgs Skolotājam". 1993. Nr.13. 2 lpp.

Tēzes

1. Витиņш М. Обучение с помощью ЭВМ. Тезисы докладов младших научных сотрудников и аспирантов "Совершенствование методов обучения в средней школе". Часть II. М.: НИИ СиМО АПН СССР. 1977. С.10-11.
2. Витиņш М. Система обучения решению систем линейных уравнений. Тезисы докладов и сообщений всесоюзного совещания

"Использование ЭВМ в обеспечении учебного процесса и управлении образованием". Свердловск: Свердловский ГПИ. 1979. С.139.

3. Витиньш М. Обеспечение качественного изучения курса "Основы информатики и вычислительной техники". Тезисы докладов всесоюзной научно-методической конференции "Интенсификация учебного процесса в высшей школе на базе микропроцессорных вычислительных систем". Воронеж: ВПИ. 1987. С.150.
4. Buligins L., Pavlov S., Vitins M. Ueber die Erarbeitung von Software fuer das Bildungswesen. Kurzfassung VI Wissenschaftliches Kolloquium "Leistungsorganisation und elektronische Datenverarbeitung". Rostock: Wilhelm-Pieck-Universitaet. 1988. S.123-124.
5. Витиньш М. Экспериментальные программы курса ОИВТ для общеобразовательных школ ЛатвССР. Материалы всесоюзной научно-практической конференции "Психолого-педагогические вопросы компьютеризации обучения в средней школе". М.: НИИ СиМО АПН СССР. 1989. С.117-118.
6. Витиньш М., Озолия Л., Раудис А. Учебная база данных для КУВТ-86. Тезисы докладов региональной научно-методической конференции преподавателей вузов Прибалтики, Беларусской ССР и Калининградской области РСФСР "Опыт применения новых методов и технических средств обучения". Калининград: Калининградский ГУ. 1990. С.114.
7. Витиньш М. Курс информатики в средних общеобразовательных школах Латвии. Тезисы докладов конференции "Дименсия европейского образования в реформе систем образования стран Балтики". Вильнюс: Министерство культуры и образования. 1992. С.45-46.