

**H.Orazberdiýew, I.Kaibow**

**Hasaplaýyş fizikasy we  
mikroprosessor ulgamlary**

**AŞGABAT – 2010**

**Hojamhammet Orazberdiyew,  
Ilmyrat Kaibow**

**Hasaplaýyş fizikasy we mikroprosessor ulgamlary**

Ýokary okuw mekdepleriniň radiofizika we elektronika  
hünäriniň talyplary üçin.

Ý.B.Seisowyň redaksiýasy bilen

11. „Täze Galkynyş eýýamy. Wakalaryň senenamasy-2007 ýyl.” Aşgabat, 2008.
12. Gurbanguly Berdimuhamedow. Ösüşiň täze belentliklerine tarap.Saýlanan eserler. I tom.Aşgabat, 2008.
13. Akbibi Ýusubowa „Beýik Galkynyşyň waspy,” Aşgabat, 2008.
14. Э.В. Бурсиан – Задачи по физике для компьютера. М. “Просвещение”. 1991.
15. А.В.Стрижков и др. – Практикум по программированию. М. “Высшая школа.” 1988
16. Бьерн Страуструп. Язык программирования C++.М. “Mir”. 1988.
17. Фигурнов Э.В. IBM PC для пользова́теля. - М.: ИНФРА-М, 1999.
18. Каймин В.А. Информа́тика. Учебник для вузов. - М.: «Высшее образование», 1998.
19. Каймин В.А. Информа́тика и дистанционное обучение. - М.: Научная книга, 1996.
20. Каймин В.А. Информа́тика. Учебник для дистанционного обучения. М., МЭСИ, 1999.
21. Информа́тика. Энциклопедический словарь для начинающих/ Сост. Д.А. Поспелов. - М.: «Просвещение», 1994.
22. Глушков В.М. Основы безбумажной информа́тики. - М.: Наука, 1982.
23. Байков В. ИНТЕРНЕ́Т: от E-MAIL к WWW в примерах. -СПб.: BHV, 1996.
24. Артамонов Б.Н. и др. Основы современных компьютерных технологий. - Спб.: КОРОНА, 1998.

**H.Orazberdiyew, I.Kaibow**

## **Hasaplaýyş fizikasy we mikroprocessor ulgamlary**

Ýokary okuw mekdepleriniň radiofizika we elektronika  
hünäriň talyplary üçin okuw gollanmasy

*Türkmenistanyň Bilim ministrligi tarapyndan hödürlendi*

AŞGABAT – 2010

**H.Orazberdiýew, I.Kaibow**

Hasaplaýyş fizikasy we mikroprosessor ulgamlary.  
Ýokary okuw mekdepleriniň radiofizika we elektronika  
hünäriniň talyplary üçin okuw gollanmasy. 2010.

**EDEBIÝAT**

1. Gurbanguly Berdimuhamedow, „Türkmenistanda saglygy goraýyşy ösdürmegiň ylmy esaslary,” Aşgabat,2007.
2. Türkmenistanyň Prezidenti Gurbanguly Mälikgulyýewiç Berdimuhamedow. Gysgaça terjimehal. Aşgabat,2007.
3. „Halkyň ynam bildireni” .Aşgabat,2007.
4. Gurbanguly Berdimuhamedow, „Garaşsyzlyga guwanmak, Watany, halky söýmek bagtdyr” . Aşgabat,2007.
5. „Türkmenistanyň Prezidenti Gurbanguly Berdimuhamedowyň daşary syýasaty. Wakalaryň hronikasy.” Aşgabat,2007.
6. Gurbanguly Berdimuhamedow, „Türkmenistan – sagdynlygyň we ruhybelentligiň ýurdy,” Aşgabat,2007.
7. Gurbanguly Berdimuhamedow.Eserler ýygyndysy.Aşgabat,2007.
8. Türkmenistanyň Prezidenti Gurbanguly Berdimuhamedowyň ýurdy täzeden galdyrmak baradaky syýasaty.Aşgabat,2007.
9. „Parahatçylyk, döredijilik,progress syýasatynyň dabaralanmagy.” Aşgabat,2007.
10. Türkmenistanyň Prezidenti Gurbanguly Berdimuhamedowyň Umumymilli „Galkynyş” Hereketiniň we Türkmenistanyň Demokratik partiýasynyň nobatdan daşary V gurultaýlarynyň bilelikdäki mejlislerinde sözlän sözi.

386 prosesoryň döremegi bilen personal kompýuterlerde ýadyň has öndürijilikli ulanmaga mümkinçilik açyldy hem-de hakyky režim giňden ulanylyp başlady. Birinji çykarylan prosesoryň käbiri 32-razryadly režimde öz iş mümkinçiligini görkezip bilmeýärdi. Bu ýagdaýy bildirmek üçin onda “16 bit operations only” belgi goýulýardy. Şeýle prosesory kompýuterlerde 32 – razryadly operasion ulgamy gurnajak bolsaň onda ol üstünliksiz tamamlanýardy(Windows 95 gurnamakçy bolanyňda ýalňyşlyk barada habar berilýär “B1”). Emma 386 prosesoryň aglaba köplügi ýokarda görkezilen, 32-razryadly prosesoryň aýratynlyklaryna niýetlenen.

## MAZMUNY

1. Giriş. Kompýuterde çözüýän radiofiziki meseleler....	6
2. Algoritmiki we obýekt-oriýentirlenen diller. C++ dili...	12
3. C++ dilinde dolandyryýan gurluşlar. Şahalanýan gurluşly algoritmleri ulanýan fiziki meseleleri programmirlemek.....	20
4. C++ dilinde funksiýalar. Funksiýalary ýazmak. Funksiýalary kesgitlemek.....	26
5. C++ dilinde massiwler bilen işlemek. Bir ölçegli massiwleri ulanmak bilen radiofiziki meseleleri programmirlemek.....	32
6. C++ diliniň grafiki mümkinçilikleri. C++ diliniň grafiki operatorlary.....	37
7. Radiofiziki meselelerde sanly differensirmek üçin ulanylýan algoritmler.....	40
8. Radiofiziki meselelerde sanly integrirmek üçin ulanylýan algoritmler.....	44
9. Monte-Carlo hasaplaýyş algoritmleri. Monte-Carlo hasaplaýyş usulynyň aýratynlyklary.....	47
10. Kompýuteriň esasy gurluşlary. Kompýuterleriň funksional bölekleri.....	51
11. Mikroprosessor sistemalary. Mikroprosessor sistemalarynda analog we sanly signallar.....	56
12. Mikroprosessorlarda signallaryň strukturasy. Signallaryň klassiki gurluşy.....	62
13. Mikroprosessor sistemalarynyň iş rezimleri. Programmalaýyn informasiýa çalşygy.....	68
14. Mikroprosessor sistemalarynyň arhitekturasy.....	71
15. Mikroprosessor sistemalarynda informasiýa alyş-çalşyny guramak.....	75
16. Mikroprosessor sistemalarynyň şinalary. Maglumat we adres şinalary.....	79
17. Mikroprosessor sistemasynda informasiýa alyş-çalşyş sikli. Programmalaýyn alyş-çalşyş sikli.....	83
Edebiýat.....	85

## **1.Giriş.Kompýuterde çözülyän radiofiziki meseleler.**

Biziň ýaşaýan dünýämiziň ýaşı ýigirmi milliard ýyl hasap edilýär. Dünýädäki hemme jisimler makro we mikro obýektlerden durýar. Makro obýektlere degişli jisimleriň ölçegleri adamyň ölçegleri bilen deňeşdirilendir. Adatça, makro jisimleri janly we jansyz diýen böleklere bölýärler. Janly jisimler bölegine adamlar, haýwanlar we ösümlikler girýärler. Jansyz bölegine jaýlar, transport serişdeleri, stanoklar, mebelleri we ş.m. degişlidirler.

Makro jisimler mikro jisimlerden ýagny molekulalardan, atomlardan ybaratdyr. Bular hem, öz gezeginde, elementar bölejiklerden durýar. Bizi gurşap alan dünýäni obýektleriň iýerarhiki hatary hökmünde göz önüne getirmek mümkin: elementar bölejikler, atomlar, molekulalar, makrojisimler, ýyldyzlar, galaktikalar(1.1-nji surat).

Adamzadyň tebigaty özleşdirmekde toplan tejribesi we bilimleri informasiýany özleşdirmek prosesi bilen baglydyr. Adamlar ir döwürden bäri informasiýa bilen iş salyşýarlar. Ilki informasiýa nesilden nesile dil üsti bilen geçirilipdir. Bular awtomatlaşyş tärleri, ekin meýdanlaryny işläp bejermek baradaky we ş.m. informasiýalardyr. Soňra informasiýany grafiki şekilde aňladýarlar. Ilkinji daş ýüzündäki jandarlaryň, ösümlikleriň we adamlaryň şekilleri 20-30 müň ýyl mundan öň döredilendir.

Informasiýany has döwrebap fiksirlenmegiň gözlegi hat ýazuwynyň döremegine getiripdir.

Häzirki wagtdaky kitaplaryň akymy, tehniki dokumentler, gazet-žurnallar, radio, telewideniýe we beýlekiler informasiýa ummanyny emele getirýär.

Wagtyň geçmegi bilen, kitap hem amatsyz, çylşyrymly, gymmat, esasan hem haýal informasiýa görerijisine öwürlip barýar.

## **17. Mikroprosessor sistemasynda informasiýa alyş-çalyş sikli. Programmalaýyn alyş-çalyş sikli.**

1985 ýylda çykarylan Intel386 birinji 32-raýadly prosessordy. Onda 32-bitly aýratyn salgy we berlenler şinalary we konweýýer salgylanma we berlenler şinanyň (16/32) raýadlylygyny dolandymaga mümkinçiligi bardy. Soňra 1988 ýylda Intel firmasy 16-raýadly berlenler şinaly we 24-raýadly salgylar şinaly Intel386<sup>tm</sup> SX prosessory çykardy, dolyraýadly prosessor bolsa Intel386<sup>tm</sup> DX adyny aldy. Kompýuteriň bahasyny arzanlatmak üçin SX görnüşini çykaryldy. Prosessor Intel386<sup>tm</sup> SX tak-mynan birýarym esse pes çaltlykly işleýär Intel386<sup>tm</sup> DX prosessor bilen deňeşdirilende şol bir takt ýygylkda. Intel386<sup>tm</sup> SX we Intel386<sup>tm</sup> DX prosessorlar diňe daşky berlenler we salgylar şinalary bilen tapawutlanýarlar, olaryň maksatnama şekilleri birmeňzeş. 1990 ýylda Intel386<sup>tm</sup> SL prosessor çykdy, onuň elektrik ýomitlendiriji serişdeleri ýörite portatiw kompýuterler üçin ýasalan.

386 prosessorlaryň buýruklarynyň içki tertib nobaty 16 baýt ölçegi eýeleýär. Instruksiýalary ýerine ýetirmek üçin gerek bolan taktlaryň mukdary edil 80286 prosessorlaryndaky ýaly. 8087 we 80287 soproprocessorlaryň maksatnamalaýyn orny tutýan Intel387 soproprocessor peýdalanmak göz önüne tutulan. 80287 soproprocessor hem ulanmak mümkin, emma onuň öndürüljigi pes.

segmentler görnüşinde düzülip bilýar (hakyky režimda – fiksirlenen uzynlykly). Segmentler diske ýüklenip hökmançylyk ýüze çykmagy bilen ondan fiziki ýat tarapyndan alynyp biliner. Segmentlemiden başgada goralan režimde logiki yadyň her birini fiziki yadyň islendik ýaýlasyna şekilendirip bolýan 4 Kbaýtlyk sahypalara (Paging) bölüşdirmek bolýar. Segmentleme we sahypalara bölmeklik islendik utgaşdyrmada ulanylyp bilinerl. Segmentleme munuň özi logiki yady edara etmek üçin niýetlenen serişde bolup hyzmat edýär.

Huşda ulanylşy boýunçy üç sany salgylar giňişligini tapawutlandyrýarlar: logiki, çyzykly we fiziki. 32 – razrýadly prosessorlaryžimi diýip salgylar giňişligini özgertmegiň ähli mehanizimi işliýän, goralan režim hasaplanylýar.

Wirtual hem diýip atlandyrylýan logiki salgy selektordan (hakyky režimde – ýöne segmentiň salgysy) we süýşmeden ybarat. Süýşme effektiv salga düzüjileriň (base, index, disp – komponentleriň ) jemlemek arkaly düzülýär. Her bir mesele 16K çenli selektorlara eýýe bolup bilýänligi we segmentiň ölçegi bilen çäklendirilen süýşme 4 Gbaýta ýetip bilýänligi üçin, her mesele üçin salgy giňişli 64 terabaýta çenli ýetip biýär.

Wirtual yadyň bu ähli giňişligini maksatnmaçy ulanyp biler (oňa operasion ulgam mümkinçilik bermeli).

Elektron hasaplaýyş maşynlary (EHM)-kompýuterleri XX asryň iň uly oýlanyp tapylan tehniki serişdeleriniň biridir. Olar, aragatnaşyk serişdereleri bilen bilelikde, bu gigant informasiýa ummanyny hemmeler üçin elýeterli edýärler.

Häzirki wagtda personal kompýuterleriň önümçilikde giňden ornaşdyrylmagy bilen bilelikde olarda çözülyän meseleleriň dürliligi-de, çylşyrmlylygy-da artýar. şu sebäpli programmistleriň önünde ylmy we ykdysady meseleleri çözmek üçin bar bolan dilleriň haýsysyny saýlap almaly diýen sorag durýar. DBase-III++, dBase-IV, FoxPro Pascal ýa-da olaryň rus dilindäki analoglary Rebus, Karat ýaly ulgamlaryň dilinde maksatnama düzmeği hünärmen derejesinde öwrenen programmistler mümkin boldugyça derrew Pascal Delphi, Jawa, C++ ulgamlarda işlemäge geçýändigini tejribe görkezýär.

Bu görkezilen dillerde düzülen maksatnamalar diňe bu dilleriň interpretatorlarynda işleýän bolsa, Delphi, Jawa, C++ düzülen maksatnama .EXE görnüşe özgerdilyär we islendik operasion ulgamda işleýär.

## OPERASIÝALAR

Object Pascalda şu aşakdaky operasiýalar ulanylýar:

**unar not, @ ;**

**multiplikatiw \*, /, div, mod, and, shi, shr;**

**additiw +, -, or, xor;**

**gatnaşyk =, <>, <, >, <=, >=, in.**

Operasiýalaryň artykmaçlyklary tertip boýunça peselýär. Dürli görnüşdäki operandlar bilen işleýän operasiýalar tablisada görkezilen.

Operasiýa	Täsir	Operandyň typy	Netije
not	Inkär etmek	Logiki	Logiki

@	Salgy	Islendik	Görkeziji
*	Köpeltmek	Islendik bitin	Iň kiçi bitin san
*	Köpeltmek	Islendik hakyky	Extended
/	Bölmek	Islendik san	Extended
div	Bitin bölmek	Islendik bitin	Iň kiçi bitin san
mod	Bölmeden galyndy	Islendik bitin	Iň kiçi bitin san
and	Logiki I	Logiki	Logiki
+	Goşmak	Islendik bitin san	Iňkiçi bitin san
+	Goşmak	Islendik hakyky	Extended
+	Setirleri birikdirmek	Setir	Setir
-	Aýyrmak	Islendik bitin	Iň kiçi bitin
-	Aýyrmak	Islendik hakyky	Extenden
or	Logiki ýa-da	Logiki	Logiki
=	Deňdir	Islendik yönekeý ýa- da setir	Logiki
<>	Deň däldir	Islendik yönekeý ýa- da setir	Logiki

Goralan režim eýýäm 80286 prosessorlardam bardy, emma ol 32-razrýadly prosessorlara mahsus bolan ähli mümkinçiliklere eýe däl. Elýeterli däl buýruklary bermeklik, ýadyň jogap beriş meýdanyndan we giriş çykyş/çykyş mümkinçilik berlen ýaýlasýndan çykmaklyk gorag ulgamy arkaly gözegçilik edýär.

Processor 8, 16 we 32 bitly operondlar bilen baýlar setiri, sözler we ikeldilen sözler, şeýle hem bitlar, bitlar meýdany we bitlar setiri bilen işläp bilýär. Otur bilen ýazylan sanlar hem-de transsendent funksiýalar bilen iş salşanda 32 razrýadly prosessor kuwwatly matematiki soproessory ulanýar. 486 we ondan soňky prosessorlarda adatça soproessor ösünde saklanýar 386 prosessor üçin 80387 soproessor niýetlenen (ol käbir üýgeşikleriň bolmagy bilen 8087 we 80287 bilen ylalaşykly) 80387 soproessordan başgada has öndürjilikli FPU C<sub>x</sub>83987 we C<sub>x</sub>83D87 soproessorlar bar (olar Cyrix firmasynyňky we 3865X we DX üçin degişliilkde niýetlenen).

80x86 prosessorlar üçin huş baýtlara (8 bit), sözlere (16 bit), ililik (32 bit) sözlere bölünýärler. Sözler kiçi baýtdan başlap iki sany goňşy baýtlara ýazylýarlar. Ililik sözler şeýle hem kiçi baýtdan başlanýar we dört sany goňşy baýtlarda ýazylýar, kiçi baýtyň salgysy ililik sözüň hem salgysy bolýar.

Sahypalar we segmentler huşyň ulyrak birlikler bolup durýarlar. Logiki taýdan huş bir ýada köp üýtgeýan uzynlykly



ondan soňkylary 386+ belgi bilen, 486 we ondan soňkylary 486+ we meňzeş belgilemeleri ulnmakçy.

### Arhitektura:

Ähli 32-razrýadly prosessorlar iki režimde işläp bilýärler. Bu režimleriniň arasynda iki tarapa hem aňsat we çalt geçiş üpjün edilen.

Real Address Mode – salgylamanyň hakyky režimi (ýa-da ýöne hakyky režim – Real Mode), 8086 bilen doly ylalaşykly. Bu režimde 1Mbaýtlyk fiziki ýada salgylanyp bolýar (hakykatdan edil 80286-laryňky ýaly 64 Kbaýt köp).

Protected Virtual Address Mode – wirtual salgylanyşly goralan režim (ýa-da ýöne goralan režim – Protected Mode). Bu režimde prosessor 4 Gbaýtlyk fiziki ýada salgylanmaga mümkinçilik berýär. Onuň üsti bilen sahypaly salgylanma mehanizmini ulanyp her bir mesele üçin 16 terebaýtlyk wirtual ýad şekillendirip bolýar. Belläp geçmeli täzelikleriň biri hem virtual 8086 Mode - 8086(V86) prosessoryň wirtual režimi. Bu ýagdaýda prosessor meseleler bilen iş salşanda özüni edil 8086 prosessor ýaly alyp barýar. Bu režimde şol bir prosessor bilen şol bir wagtda dürli hakyky resurslary ulanýan meseleler bilen işleşip bolýar. Bu ýagdaýda ýadyň fiziki salgylar giňişligine ýüzlenmeler segmentleme we geçirme mehanizmleri bilen dolandyrylýar.

<	Kiçidir	Logiki	Logiki
<=	Kiçi ýa-da deň	Logiki	Logiki
>	Ulydyr	Logiki	Logiki
>=	Uly ýa-da deň	Logiki	Logiki

### Bitin tipli berlenleriň üstünde logiki operasiýalar

Operand 1	Operand 2	not	and	or	xor
1	-	0	-	-	-
0	-	1	-	-	-
0	0	-	0	0	0
0	1	-	0	1	1
1	0	-	0	1	1
1	1	-	1	1	0

### Logiki tipli berlenleriň üstünde logiki operasiýalar

Operand 1	Operand 2	not	and	or	xor
True	-	False	-	-	-
False	-	True	-	-	-
False	False	-	False	False	False
False	True	-	False	True	True
True	False	-	False	True	True
True	True	-	True	True	False

Ady	Bitin tipler	
	Uzynlygy,baýt	Bahalarynyň diapazony
Cardinal	4	0...2 147 483 647
Byte	1	0...255
Shortint	1	-128...+127
Smallint	2	-32 768...+32 767
Word	2	0...65 535
Integer	4	-2 147 483 648...+2 147 483 647
Longint	4	-2 147 483 648...+2 147 483 647
Int64	8	$-9 \cdot 10^{18} \dots +9 \cdot 10^{18}$
LongWord	4	0...4 294 967 295

#### Standart proseduralar

Ýüzlenme	Netijäniň tipy	Täsir
abs (x)	x	X-iň modulyny gaýtarýar
chr(b)	Char	Kody boýunça simwoly gaýtarýar
dec (vx [, i] )	-	vx-iň bahasyny i sana azaldýar
inc(vx[,i])	-	vx-iň bahasyny i sana atdyrýar

başgasy).Sinronizosiýany üpjün etmek mümkin, soprocessoryň iş bilen baş ululygyny habar berýän BUSY çykyş, TESTCPU giriş bilen birikdirýärler.

## 16. Mikroprocessor sistemalarynyň şinalary.

### Maglumat we adres şinalary.

Intel 32-razrýadly processorlaryň taryhy Intel386-dan başlaýar. Bu processorlar özünden öňki 16-razrýady bolan 8086/88 we 80286 processorlaryň ähli häsiýetini özünde saklaýar. Beýle ýagdaý öň ýazylyp goýlan ägirt uly maksatnamalar toplumynyň bu processorlarda ylalaşykly işlemäge mümkinçilik berýär. Ýöne olarda häzirki zaman nukdaý nazaryndan seredilende örän berk çäklendirme (ýadyň üznüksiz segmenti – 64 Kbaýt) aýryldy. Goralýan režimde ol 4Gbaýta çenli süýşirildi (fiziki salgylandyrylýan ýadyň çägi). Olaryň giňeldien buýruklar toplumy bolup, bu toplumda 8086, 80286 processorlaryň ähli buýruklary saklanýar. Processor dört derejeli ýadyň giriş/çykyşyň we dürli meselelere geçişni gorag ulgamy bilen üpjün edilen. Bu processorlaryň ählisi 64 terobaýtlyk wirtual ýady bar.

386 prosessordan soň täze nesil döredi – 486, Pentium, Pentium Pro. Ähli maşgalanyň wekillerindäki ýaly olarda öňki agzalar bilen maksatnama ylalaşygy üpjün edilen. Biz 386 we

					80287-a ýazmak
00fc	1	0	1	0	Inkär etmäniň görkeziji siniň ýazgysy
00fc	1	0	0	1	Ätiýaçla ndyrylan

Tablisa 6.1 80287 so prosessoryň şinaaýlawlaryň niýetlenenişi.

Tablisa 4.2 80287 soprosessoryn signallaryň niýetlenen işi

Signal	1/0	Niýetlenen işi
CLK	1	Clock-paossesoryň sinhronizirleýji daşky signaly
CKM	1	Clock Mode-sinhronirleme režimi: CKM=1 ýagdaýda soprosessoryň içki ýygylgy daşky bilen gabat gelýär,CKM=0 bolanda içki ýygylgy CLK 3 bölmek

Enjamyň interfeýsi we çykyşyň ýerleşýi edil maksimal rejimde işleýän 8086/8088 pressesoryňky ýaly. Soprosessoryň we prosessoryň biratly çykyşlary özara birikdirilen (RQ/GT1-den

Hi(w)	Byte	Argumentiň uly baýtyny gaýtarýar
Hi(I)	Byte	Tertip boýunça üçünji baýty gaýtarýar
Lo(i)	“	Argumentiň kiçi baýtyny gaýtarýar
Lo(w)	“	Argumentiň uly baýtyny gaýtarýar
odd(l)	Boolean	Argument jübt däl san bolsa True gaýtarýar
Random(w)	Islendik	Tötän san gaýtarýar
sqr(x)	X	Argumentiň kwadratyny gaýtarýar
swap(i)	Integer	Sözde baýtlaryň ýerini çalyşýar
swap (w)	Word	Sözde baýtlaryň ýerini çalyşýar

Meselem, indiki maksatnama işledilse ekranda 0 peýda bolýar:

**procedure** TfmExample.bbRunClick(Sender: TObject) ;

**var**

k: Word;

**begin**

k := 65535; // Word tipiň maksimal bahasy

k := k+1; // matematikanyň düzgünleri boýunça

```
k=65536 IbOutput.Caption := IntToStr(k); // Hakykatdan bolsa
k=0!
end;
```

## 2.Algoritmiki we obyekt-orientirlenen diller. C++ dili.

Informasiýa adalgasy „informatio“ diýen latyn sözünden delip çykandyr. Ol düşündirmek, beýan etmek, habarlylyk ýaly manylary berýär. Adamlar informasiýany biri-birine söz üsti , hat üsti , dürli hereketler bilen we beýleki belgiler arkaly berýärler.

Gerekli informasiýany biz okuw kitaplaryndan, teleýaýlymlardan, Internetden, kinofilmlerden alýarys. Informasiýany öz depderlerimize, konspektlerimize ýazýarys. Adamalar önümçilikde informasiýany biri-birine tekst we çyzgy, hasabat we maglumat, tablisalar we beýleki dokumentler görnüşinde geçirýärler. Informasiýa kompýuterleriň kömegi bilen hem berlip bilner. Kabul edilýän ýa-da iberilýän informasiýanyň käbir formasy bolmalydyr: gepleşik, surat, makala we ş.m. Kitaplar, suratlar, saz eserleri, spektakller, kinofilmler – bularyň hemmesi informasiýany aňlatmagyň formalarydyr.

Informasiýa haýsy formada bolsa-da ol real ýa-da hyýaly dünýäni häsiýetlendirýär. Şunlukda informasiýa jisimler dünýäsini bellikleriň, signallaryň kömegi bilen aňlatmakdyr.

Informasiýanyň takyk kesgitlemesini bermek mümkin däl. Biziň bilşimiz ýaly, şunuň ýaly ýagdaýlar beýleki ylmlarda hem bardyr. Mysal üçin, fizikada materiýa, energiýa, wagt düşüňjeleriniň, geometriýada nokat, tekizlik düşüňjeleriniň takyk kesgitlemeleri ýokdyr. Bular takyk kesgitlemesiz, düýp düşüňjelerdir.

Informasion prosesleriň umumy kanunalaýyklaryna seretmezden, çylşyrymly (biologiki, sosial, tehniki) sistemalaryň

Prosessory bu ýagdaýdan enjamdan NMI signalyň üsti bilen (ony goralan režimde galdyryp) ýa-da REST buýrugynyň üsti bilen hakyky režime getirip çykaryp bileris.

Salgy	CMD1	CMD0	NPRD#	NPWR	Aýlawyň niýetlenen işi
00f8	0	0	1	0	Amalyň belgiyny 80287-ýazmak
00f8	0	0	0	1	80287-den Swýa-da CW okamak
0wfa	0	1	0	1	80287-den maglumatlar okamak
00fa	0	1	1	0	Maglumatlary

(wektor) bilen). Maksatnama arakesmeleri üçin görkezme buýruk bilen berilýär; gizlenilýän enjam arakesmeleri üçin ikinji INTA# aýlawda daşky kontrolýolyň üsti bilen girizilýär; gizlenilmedik arakesmeler belli wektora eýe iňkär etmeler bolsa prosessiriň içinde wektorlaryň önderýärler we geçirilýärler.

Arakesmeleriň her bir tertip sany üçin arakesmeleriň diskriptler tablisasyndaky IDT (Interrupt Descriptor Table) bir element degişli edilýär. Hakyky režimde arakesmeleriň tablisasynyň gurluşy edil 8086/88 prosesordaky ýaly. Ýöne LIDT buýrugyň üsti bilen onuň ýerini birinji megabaýtyň çäklerinde üýtgedip bolýär, ölçegi bolsa (03FFh) 007Fh çenli kiçeldip bolýär. Haçanda tertibi jedweliň berlen ölçeglerinden çykýan arakesmelere hyzmat edilmäge synanşylsa 8 görnüşli iňkär etme öndürilýär. Garalýan režimde IDT jedweli 8 boýtly arakesmeler diskriptyny özünde saklaýar. Ol 32 - den 256 - a diskriptora çenli ölçege eýe bolup bilýär we fiziki ýadyň islendik ýerinde ýerleşýär.

Ikileýin iňkär etmek (Double Fault) - ol prosessoryň bir görkezme bilen işleýärkä iki sany biri-biri bilen baglanşyksyz iňkär etmeler gabat gelmegi. Bu ýagdaýda 8 iňkär etme ulanylýar. Haçanda oňa hyzmat edilýärkä haýsydyr bir iňkär etme bolaysa, prosessoryň öçmesi (Shutdown) bolýar. Öçen wagtynda hiç bir täze görkezmeler ýerine ýetirilmeýär.

gurluşyny we funksionirlenişini düşündirmek mümkin däldir. Informasiýa düşünjesi entropiýa düşünjesi bilen berk baglanşyklydyr.

Termodinamikanyň ikinji kanuny entropiýa barada şeýle diýýär. Eger ýapyk sistema (daşky täsir ýok) özbaşyna goýberilse onda ol termodinamiki deňagramlylyga ymtylýar. Sistemany düzýän elementler garyşýarlar, olaryň strukturasy bozulýarlar we doly tertipsizlik (haos) emele gelýär. Tertipsizligiň ölçegi bolan entropiýa ösýär, tertipliligiň ölçegi bolan informasiýa bolsa azalýar.

Belli alymlar informasiýany dürli hili keagitläpdirlir.

Kibernetikada informasiýa diýip, bilimleriň çylşyrymly sistemalary dolandyrmak, aktiw hereket etdirmek, oriýentirlenmek üçin ulanylýan bölegine düşünilýär. Informasiýa-bu sistemanyň saklanmagy, köpelmegi, ösmegi üçin, bilimleriň gerek bolan bölegidir(N.Winner).

Informasiýa teoriýasynda maglumatlaryň kesgitsizliginiň azalýan bölegine informasiýa diýilip düşünilýär. Maglumatyň, kabul edilmänkä bar bolan kesgitsizligi aýyrýan bölegine, informasiýa diýilýär(K. Şennon).

Semantika (habaryň manysy) teoriýasynda özünde täzelik saklaýan maglumatlar informasiýa diýilýär.

Informasiýanyň başga hili kesgitlemeleri hem bar. Belli alym B.M. Gluşkow informasiýa şeýle kesgitleme berýär. Informasiýa-bu giňişlikde we wagtda energiýanyň we materiýanyň bir hili dälliginiň ölçegidir.

Ýokarda aýdylanlardan netije çykaryp informasiýa şeýle kesgitleme bermegi has maksadalaýyk hasap etmek bolar.

Informasiýa - bu bizi gurşap alan dünýäniň obýektleri baradaky kesgitsizligi aýyrýan, saklamaga, işläp taýýarlamaga we geçirmäge niýetlenen maglumatlardyr.

Signal, habar, maglumatlar düşünjeleri informasiýa düşünjesi bilen ýakyn baglydyrlar.

Maglumatlar – bu formallaşdyrylan görnüşde we tehniki serişdeleriň kömegi bilen täzedan işlemäge niýetlenen ýa-da işlenen informasiýalardyr.

Habar – bu kesgitli formada aňladylan we aralyga berilmäge niýetlenen informasiýadyr.

Signal- bu informasiýany göteriji islendik prosesdir.

Meselem, telegraf boýunça berlen telegrammanyň teksti- bu habardyr. Mazmuny- informasiýadyr. Telefon sim boýunça gelen elektrik togy- signaldyr.

Her bir programmirleýji dildäki ýaly C++ dilinde hem açar sözleri, identifikatorlar, konstantalar we bölüjiler ulanylýar. Boşluklar, tabulýasynyň belgileri, täze setire geçiriş belgileri leksiki birlikler hasaplanylmaýar. şunuň ýaly belgileriň islendik yzygiderligi we programma düşündirişler birlik boşluk ýaly düşünilýär we interwal diýilýär.

şeýlelikde, Turbo Pascal dilinde ýazylan maksatnama leksiki birliklerden we interwallardan durýar. Açar sözleri bolup harplaryň üzülmeyän yzygiderligi bolýar. Iň gysga maksatnama şu aşakdakydyr:

```
begin  
end.
```

Bu maksatnama iki açar sözünden we bölüjiden durýar. Bu maksatnamanyň ýerine ýetirilmegi hiç hili netije bermeýär.

Identifikator bolup harpdan başlanýan harplaryň we sanlaryň yzygiderligidir. Ol açar sözi bolmaly däldir. Käbir identifikatorlaryň maşyna düşnükli manysy bolýar. Identifikatoryň göterýän manysyny programmalaýyn üýtgedip bolýar. Meselem:

```
program jb;  
const  
    false = true;  
begin  
    Writeln ( false )  
end.
```

Bu programmanyň ýerine ýetirilmegi ekranda true sözünüň ýazylmagyna getirer. Eger ikinji we üçünji setirleri aýyrsak, false sözi ýazylar.

## 15. Mikroprocessor sistemalarynda informasiýa alyş-çalşyny guramak.

Edil 8086/88 ýaly 80286 processor hem arakesmeleriň 256 görnüşini kabul edip bilýär. Arakesmeler processorň girişindäki elektrik signallar bilen döredilýän enjam we INT<sub>xx</sub> görkezme hem-de görkezmänik inkär edelmesi bilen döredilýän maksatnama arakesmelerine bölünýär. Enjam we maksatnama arakesmeleri edil ( 8086 processor daky ýaly işleýär. Görkezmeleriň inkär edil- megi ( Instrudion Exeeptions ), ýa-da ýöne inkär etme, iş ýerine ýetirilýärkä aýratyn şertleriň döremegi bilen ýüze çykýär (8086 processorň içki arakesmeleri bolup hyzmat edýär). Arakesmelerden tapawutlykda inkär etme bilen işleşenden soň (soprocessora degişli bolan 9 inkär etmeden başgalary) dolandyryş inkär edilmä getirlen görkezmä geçirilýär. Hakyky we goralan režimleriň Inkär etmeleriniň toplумы we işlenişi dürli. Inkär etmeler üçin Intel 0-31 wektorlary saklaýar. Ýöne personal kompýuterde olaryň bir bölegi Bios we DOS-yň ulgam arakesmeleri tarapyndan eýelenýär.

Arakesmelere ýada inkär etmelre hyzmat edýän prosedura tertip sanlaryň kö- megi bilen jedwellerden kesgitlenilýär (ýagny sekizbitly Arekesme görkezijisi

Selektorlar boýunça operatiw huşda saklanýan jedwelden segmentleriň deskriptorlary çykarylýar;

- ❖ Deskriptor baza salgyny, segmentiň ölçegini (1-64 Kbaýt) we onuň häsiýetlerini beýan edýar;
- ❖ Segmentiň baza salgysy 24 bitli razrýadlylygy eýeleýar. Bu 16 Mbaýt fiziki huşy salgylamagy üpjün edýar.

Prossessor 8086 ýaly 80286 Prossessor hem huşdan aýratyn giňişlikde 64K çenli birbaýtly ýada 32K çenli ikibaýtly registrleri salgylap bilýar. Giriş/çykyş salgylar giňişlikde 00F8-00FF soprossessor üçin saklanýar. Giriş/çykyş operasiýalarda salgynyň uly bity (A[16:20] çyzyklar) ulanylmaýarlar. REP INSB/INSW, REP OUTSB/OUTSW setr buýruklary bloklaýyn giriş/çykyşy DMA standart kontrolleriň analog amallary ýerine ýetiriş tizliginden hem uly tizlikde üpjün edýar.

Prossessor 8086 ýaly 80286 Prossessor hem 256 çenli görnüşli arakesmeleri işledip bilýar. Arakesmeler apparat görnüşinde (olar prossessoriň girişinde elektrik signallar arkaly çagyrylýarlar) we maksatnama görnüşinde (olar INT xx buýruk boýunça ýerine ýetirilýärler) bolup bilýärler.

Konstantalar arifmetiki, simwol, setir we logiki görnüşlere bölünýärler. Arifmetiki konstantalar bitin we hakyky görnüşlere bölünýärler. Bu konstantalara sanlar diýilýär.

Simwollaryň yzygiderligine setir konstantalary diýilýär. Eger setir konstantasy bir simwol bilen aňladylan bolsa ol simwol konstantasy bolýar. Setir konstantasy önünden we yzyndan apostrof bilen çäklenýär.

Logiki konstantalar true we false konstantalar bilen aňladylýar we hakyky - ýalan sözleri aňladýar.

**Sanalýan tip.** Sanalýan tipde alyp bilýän bahalary sanalyp berilýär. Her bir baha identifikator bilen bellenýär we ýaý skobkalary bilen çäklenýär, meselem:

**type**

colors = (red, white, blue);

Sanalýan tipler ulanylanda maksatnama has düşnükli we owardan bolýar. Meselem , maksatnamada ýylyň aýlary peýdalanylýan bolsa maksatnama şu görnüşde bolýar:

**type**

ЊипАý=(Türkmenbaşy,Baýdak,Nowruz,Gurbansoltan,Magtym guly,Oguz,Gorkut,AlpArslan,Ruhnama,Garaşsyzlyk,Sanjar,Bita raplyk);

**var**

Aý : typ Аý;

**begin**

**if aý = Gorkut then**

IbOutput.Caption := 'Deñize gitmek wagty!';

**end.**

Emma maksatnamada diňe latyn harplaryny ulanmaly bolansoň ol şeýle ýazylmaly:

**type**

TypeMonth=(jan,feb,mar,may,jun,jul,aug,sep,oct,nov,dec);

**var**

month: TypeMonth;

```

begin
if month = jun then
    IbOutput.Caption := 'Deňize gitmek wagty!';
end.

    Goý, meselem, şeýle sanalýan tipler berlen bolsun:

type
    colors = (black, red, white);
    ordenal= (one, two, three) ;
    days = (Monday, Tuesday, Wednesday);
    Bu ýerde üç tipyň hemmesi deň:
    Ord(black)=0, ... , Ord(white)=2,
    Ord(one)=0, ... , Ord(three)=2,
    Ord(Monday)=0, ... , Ord(Wednesday)=2.
    Emma eger üýtgeýän ululyklar şeýle kesgitlenen
bolsalar:
var
    col : colors;
    num : ordenal;
    day : days ;
    onda mümkin operatorlar şular bolýar
    col := black;
    num := Succ(two);
    day := Pred(Tuesday);
    emma
    col := one;
    day := black;
    operatorlary ulanmak bolmaýar.

    Sanalýan tipe degişli üýtgeýän ululyklary deslapky
    beýan etmesiz ýazyp bolýar, meselem:

var
    col: (black, white, green);
    Tip-diapazon görmüşinde şeýle ýazylýar:

type
    digit = '0'..'9';
    dig2 = 48 .. 57;

```

Bu prosessoryň artykmaçlygy diňe bölekleyin ulanylanlygyna garamazdan PC bazaryndaky haky öňe gidişlik şol prosessor bilen gazanyldy.

Razraýady 24 bit bolan salgy şinasy 16 Mbaýt fiziki ýa-da salgylanmaga mümkinçilik berýär. Ýöne hakyky režimde diňe kiçi salgylardan başlanýan, 1 Mbaýtyna rugsat berilär. Maksatnamanyň nukdaý nazaryndan belgi, edil 8086 – daky ýaly segmentler görmüşde edara edilýär, ýöne segmentleri dolandyrmaklyk hakyky we goralýan režimler toparyndan düýbünden başgaça amala aşyrylýar.

Hakyky režimde ýa-da sagylanmakdan 8086 prosessor bilen doly ylalaşyk bar.

8086-lardan tapawutlykda 80286 prosessorlar, hakyky režimde işleýän segmetiň çäginde geçmeklige gözegçilik etmäge mümkinçilik berýän enjamlary bar. Eger süýşmesi FFFFh bolan söze salgylanmakçy bolsak (onuň uly baýty segmentiniň çäginde çykýar), prosessor bölünme (prerywaniýa) işläp çykýar. Şeýle ýagdaý ähli baýty bir segmentde ýerleşmeýän amaly ýerne ýetireniňde hem bolýar.

Goralan režimde 8086 üçin we 80286 –nyň hakyky režimi üçin mümkin bolan salgylanma režimler işleýär. Tapawutlar diňe segmentiň kesgitleýişine degişli.

- ❖ CS, DS, SS we ES segment registrleri segmentleriň baza salgylarynyň özleriniň deregine selektorlary saklaýarlar.



huşy salgylanyp bilýär. Onuň üsti bilen sahypa salgylanma mehanizmini ulanyp her meseläni 1 Gigabaýt wirtual huş bilen üpjün edip bolýar.

Goralan režime geçmek ýekeje buýrugyň üsti bilen ýeterlik çalt ýerine ýetirilýär. Tersine geçiş köp wagt talap edýär.

80286 özünde 8086/88 prosessoryň ähli buýruklaryny saklamak bilen bir hatarda bimäçe goşmaça buýruklaryň arasynda iň gyzyklysy baýtyň ýadyň setirine giriş\çykyşyny üpjün edýän instruksiýalar. Hakyky režimde 80286-nyň registorlary düzümi we niýetlenen işi boýunça umuman 8086\88-iň registerleri bilen gabat gelýärler. Üýtgeşmeler flaglaryň registrleriniň bitlerine we segment registrleriň goralan režimde ulanşyna degişli. Edil 8086 ýaly, 80286 prosessorlar hem 16-bit maglumatlar şinasyna we 6-baýtlyk buýruklar komandasyna eýýe. Amallaryň ýerine ýetişini gysgaltmaga mümkinçilik beren arhitekturanyň hasabyna: 80286 prosessor (12,5 MGs takt ýygyllykly) 8086-dan (5MGs takt ýygyllykly) 6 esseden hem çalt işleýar. Ýokary öndürijiligi bolan 80287 matematiki soprasessory ulanmak göz önüne tutulan. Ol maksatnama taýdan 8087 prosessor bilen ylalaşykly.

MS-DOS-yň dolandyryjylarynda 80286 prosessor adatça hakyky režimi ulanylýr. Garalan režimi XENIX, UNIX, OS/2, NetWare286 we MSWindows-yň gabygy ýaly OS-lar ulanylýar.

ýa-da  
var

date : 1. .31;  
month: 1..12;  
Ichr : 'A'..'Z';

Hakyky tipler

Uzynlygy, baýt	Ady	Sanlaryň mukdary	Bahalarynyň diapazony
8	Real	15...16	5.0*10e-324...1.7*10e308
4	Single	7...8	1.5*10e-45...3.4*10e38
8	Double	15...16	5.0*10e324...1.7*10e308
10	Extended	19...20	3.4*10-4951...1.1*10e4932
8	Comp	19...20	-2e63...+2e63-1
8	Currency	19...20	+/-922 337 203 685477,5807

**Meselem:**

**type**

RealType = Real;

**var**

Epsilon : RealType;

**begin**

Epsilon := 1;

**while** 1+Eps4.1on/2 > 1 **do**

Epsilon := Epsilon/2;

IbOutput.Caption := FloatToStr(Epsilon)

end;

### Standart matematiki funksiýalar

Ýýzlenme	Parametrini Netijäniň tipy		Bellik
abs (x)	Real, Integer	Real	Argumentiň moduly
Pi	-	<<	$\pi = 3.141592653...$
ArcTan(x)			Arktangens (radianlarda)
cos (x)	Ol hem <<	Ol hem <<	Kosinus, radianlarda
exp(x)	<<	<<	Ekspontenta
frac(x)	<<	<<	Sanyň drob bölegi
int(x)	<<	<<	Sanyň bitin bölegi
ln(x)	<<	<<	Natural logarifm
Random	-	<<	Tötän san
Randomize	-	-	Generatory täzeden işe girizmek
sin (x)	Real	Real	Sinus, radianlarda
sqr(x)	Real	Real	Argumentiň kwadraty
sqrt(x)	<<	<<	Kwadrat kök

		salgylyar baýt
1	1	Rugsat edilmedik utgaşdyrma

## 14. Mikroprocessor sistemalarynyň arhitekturasy.

1982-nji ýylda döredilen 80286 proserssorlar 16 zaryadly processorlaryň 2-nji nesili bolup durýar. Ol köp ulanyjylar we köp meseleler bilen bir wagtda işlemäge mümkinçilik berýär. 8086/88-den esasy tapawudy bolsa huşy salgyylanmak mehanizminden ybarat. Ýörite enjamlar meseleleri üýtgetmäge mümkinçilik berýärler. Prosessor giňeldilen buýruklar toplumyna eýe bolup, ol goragy üpjün edýän enjamlardan başgada, 8086 prisessorlaryň ähli buýruklaryny, şeýle hem ýenede umumy işlere niýetlenen birnäçe buýruklary eýeleýär. Prosessor iki režimde işlemäge ukyply.

8080 Real Address Mode - hakuky salgylyanyş režimi (ýa-da ýöne hakyky režim – Real Mode), 8086 – bilen doly ylalaşykly. Bu režimde 1 Mbaýt fiziki huşy salgylylananmaga mümkinçilik bar.

Protected Virtual Address Mode – wirtual salgylylanmanyň goralan režimi (ýa-da ýöne goralan režim – Protected Mode). Bu režimde Prosessor 16 Mbaýt çenli fiziki

eksponensial we logarifmik instruksiýalar saklaýan 68 mnemonikleri girizmäge getirdi. Soprocessor 7-görnüşli maglumatlar bilen işleýär : 16-,32-,64-bitli bitin sanlar; 32-,64-,80-bitli otur arkaly ýazylyan sanlaryň formaty IEEE 754 standartta deňişli. Soprocessor matematiki amallary ýerine ýetirjilik öndürjiligi, maksatnama arkaly esasy prosessoryň ýerne ýetirijiliginden 100 esse uly. Käwagt ony FPU (Floating Point Unit) bilen belgileýärler, bu bolsa onuň otursyz sanlar bilen işleýändigini görkezenok. Başgaça belleniş NPU (Numeris Prosessiny Unit)-munda onuň merkezi prosessordan baglylygy görkezilmeýär. Maksatnamanyň nuktaý nazaryndan CPU+NPX-edil bir bütewi ýaly görülyär. Onda CPU-umumy prosesini gidişini dolandyryr we esasy buýruklar topluny işleýär, NPX bolsa diňe özüne magsus buýruklary ýerine ýetirýär.

*Tablisa 4.1 8086 we 80286 prosessorlaryň berlenler şinasy boýunça geçirmeler.*

BHE#	A0	Geçirme
0	0	Bütün söz
0	1	AD[8:15] çyzyklar boýunça täg salgylar baýt
1	0	AD[0:7] çyzyklar boýunça jüwüt

### Sene-wagt bilen işlemek üçin podprogramma

<b>Function</b> Date: TDateTime;	Häzirki senäni gaýtaryr
<b>Function</b> DateToStr(D: TDateTime): <b>String</b> ;	Senäni setire özgerdýär
<b>Function</b> DateTimeToStr(D: TDateTime): <b>String</b> ;	Senäni we wagty setire özgerdýär
<b>Function</b> FormatDateTime (Format: <b>String</b> ; Value: TDateTime): <b>String</b> ;	Format spesifikatory esasynda senäni we wagty setire özgerdýär
<b>Function</b> Now: TDateTime;	Häzirki senäni we wagty gaýtaryr
<b>Function</b> Time: TDateTime;	Häzirki wagty gaýtaryr
<b>Function</b> TimeToStr(T: TDateTime): <b>String</b> ;	Wagty setire özgerdýär

### **3. C++ dilinde dolandyryan gurluslar. Şahalanyan gurlusly algoritmleri ulanyan fiziki meseleleri programmirmek.**

Bizi gurşap alan dünýä dürli obýektlerden durýar. Olaryň her haýsynyň öz häsiýetleri bardyr we olar biri-biri bilen täsir edişýärler. Meselem, gün sistemasynyň planetalarynyň dürli häsiýetleri, massasy, geometriki ölçegleri bardyr. Olar bütindünýä dartýşma kanuny esasynda biri-birleri bilen we beýleki asman jisimleri bilen täsir edişýärler. Gün sistemasy hem öz gezeginde has uly obýekt bolan „Ak maüanyň ýoly“ atly galaktikanyň düzümine girýär.

Planetalar molekulalardan, atomlardan, olar bolsa elementar bölejiklerden durýalar. Yagny her bir obýekt beýleki obýektlerden durýar.

Özara baglanşykly elementleriň bitewi toplumyna sistema diýilýär. Sistemanyň özünde onuň hiç bir elementinde bolmadyk funksiýanyň we häsiýetiň bolmagy, sistemanyň bitewilik häsiýetidir. Ol sistemanyň elementleriniň özara täsiri netijesinde döreýär.

Sistemanyň elementleriniň özara baglanşygynyň bolmagy hökmandyr. Sistema özünden ýokary derejeli sistemanyň elementi bolup hyzmat edip biler.

Element, tutuş sistema görä özbaşdak bolup sistemanyň bir bölegidir. Sistemanyň elementleri bölünmeýär diýilip hasap edilýär. Elementleriň arasyndaky baglanşyk dürli-dürli bolup biler: energetiki, informasion, genetiki, dolandyryş we ş.m.

Sistemalaýyn çemeleşmek ylmy akyl ýetirmegiň we sosial praktikanyň metodologiýasydyr. Ol obýekte sistema hökmünde seretmekligiň esasynda amala aşyrylýar.

Häzirki wagtda informasion sistemalar jemgiýete giňden ornaşýalar. Bu sistemalaryň elementleriniň käbirleri informasiýa onýektleridir. Informasion sistemalar edara, pudak, döwlet derejesinde döredilýärler. Mekdep derejesindäki informasion sistema mekdebiň administrasiýasyny,

çyzyk bolsa diňe DX registeriň üsti bilen salgylananynda salgynyň uly baýtyny saklaýar. Eger göni salgy bilen ýüzlensek A[8:15] çyzyk nullary saklar. Bölünmäniň tassyklama aýlawy portdan okalyş aýlawyna meňzeş, ýöne IORD# signalyň deregine işjeň signal INTA# bolýar, salgylar şinasy bolsa prosessor tarapyndan dolandyrylmaýar.

Maximal režimde şina tarapyndan dolandyrylyşyň berilmegi RQ/GT signalyň üsti bilen amala aşyrylýar. Şinadan dolandyrylyşygy soraýan gurluş, soraýyş impulsyny öndürýär. Ýerne ýetirýän aýlawyny tamamlap prosessor şol çyzykdan (ýoldan) tassyklama impulsyny goýberip şinany boşadýar. Indi şinany soproprocessor dolandyryp başlaýar, ol öz işini gutaranda bolsa ol indiki impuls bilen prosessor şinanyň boşadylandygy baradak habary ýetirýär.

8086 prosessor käbir özboluşly tapawutlara eýe. Bu tapawut maglumatlar şinasynyň razýadynyň iki baýtly bolmagy bilen baglanşykly. Prosessorda AO salgynyň kiçi biti bilen bilelikde maglumatlar şinasynyň baýtyny ulanmagy kesgitleýän goşmaça BHE# signal bar. BHE# signal hakyyk salgy bilen bir wagtda bolýar.

NPX diýip atlandyrmak kabul edilen 8087 soproprocessor 8086/8088, 80186/80188 merkezi prosessoryň matematiki mümkinçiliklerini artdyrmak üçin niýetlenen. Onuň ulanmagy 8086 buyruklar ulgamyna özüde arifmetiki, trigonmetriki,

### 13. Mikroprosessor sistemalarynyň iş režimleri.

#### Programmalaýyn informasiýa çalşygy.

8086 we 8088 prosessorlaryň çykyşlarynyň ýerleşşi köp zatlarda gabat gelýär. MN/MX# girişdäki signalyň derejesine baglylykda prosessorlar minimal (iň kiçi) we maximal (iň uly) režimlerde işläp bilýärler. Minimal režimde daşky şina üçin niýetlenen signallary prosessorlar özi işläp çykarýarlar. Bu režim soproprocessor ulanmaýar, uly bolmadyk çyzgylary gurmakda ulanylýar we 8088 prosessor 8085 maşgalanyň (pezeferiýa) mikroshemalaryny birleşdirmäge mümkinçilik berýär. PC ulanylýan maximal režimde ulgamyň IOR#, IOW#, MEMR#, MEMWR#, INTA# we ALE şinalaryny dolandyryýan signallary i8288 kontroller prosessoryň ýagdaýy baradaky signallaryň üstünden işlenilýär.

#### Prosessoryň şina aýlawlary:

Prosessoryň işiniň PC-da ulanylýan maximal režiminde ulgamyň şinasyny dolandyryýan signallar 8288 kontroller prosessoryň ýagdaýy baradaky signallaryň üsti bilen işlenilýär. Ähli dolandyryjy signallaryň işjeňligi pes logiki derejede. Bu bolsa birnäçe enjamlara şinany dolandyрма işini bölüşdirmäge mümkinçilik berýär.

Portlara ýüzlenme aýlawlary ýadyň salgylar şinasyny ulanylýan aýlawlaryndan tapawutlanýar. Portlara ýüzlenende A[16:19] salgý çyzygy hemişe özünde nollary saklaýar. A[8:15]

mugallymlary, okuwçylary we beýleki işgärleri okuw prosesi üçin gerekli informasiýalar bilen üpjün edip biler we personalyň iş öndürijiligini ýokarlandyrar.

Islendik informasion sisitemany 4 esasy elemente bölmek mümkin:

1. Tehnologiki proses;
2. Prosesi amala aşyrmak üçin apparat;
3. Barlag we dolandyryş serişdeleri;
4. Elementleriň arasyndaky informasion baglanyşyk.

Kompýutere hem dürli gurulşlardan(elementlerden) durýan sistema hökmünde seretmek mümkin. Kompýuteriň elementleri biri-birine apparat we funksional taýdan baglydyr.

Sistemanyň elementleriniň arasynda informasiýa alyşmasy signal formasynda bolýar.

Ätiýaç sözler bolan **begin ... end** bilen çäklenen operatorlaryň yzygiderligine düzümlü operatorlar diýilýär. Olar häzirki zaman strukturaly programmirlenmegiň esasy guraly bolup durýar. Düzümlü operatorlaryň içinde islendikçe ýene düzümlü operatorlaryň bolmagy mümkin. Meselem:

**begin**

**begin**

**begin**

**end;**

**end;**

**end;**

#### **Şertli operator**

Şertli operator käbir şerti barlap, barlag netijesine görä ol ýa-da başga hereketleri etmäge mümkinçilik berýär. Diýmek bu operator hasaplaýyş prosesini şahalandyrmak üçin ulanylýar. Onuň strukturasý şeýledir:

if <şert> **then** <operator1> **else** <operator2>;

bu ýerde if/ then/ else – ätiýaçlyk sözleridir  
 (eger,onda,basgaça);  
 <şert> - erkin logiki aňlatma;  
 <operator1>, <operator2> - Object Pascalyň islendik  
 operatorlary.

**Meselem:**

```
var
X, Y, Max: Integer;
begin .
if X > Max then
Y := Max else
  Y := X;
....
end;
```

**Aýlaw operatorlary**

C++ dilinde aýlaw operatorlaryň üç görnüşi bar.  
 Olaryň kömegi bilen maksatnamanyň gaýtalanyp ýerine  
 ýetirilýän böleklerini programmirläp bolýar.

*Sikli hasaplaýjysy bolan* **FOR** operatory şu görnüşe eýe:

```
for <sikliň parametri> := <baş._bahasy> to <soňky bahasy> do
<operator>;
```

Bu ýerde for, to, do – ätiýaçlyk sözleri (*üçin, çenli, ýerine  
 ýetirmeli*);

<sikliň parametri> - Integer tipli üýtgeýän ululyk;

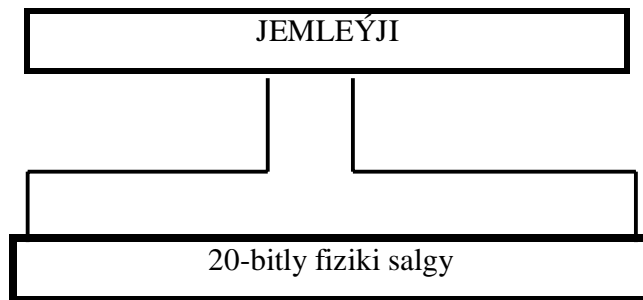
Bu operator ýerine ýetirilende ilki başlangyç baha  
 hasaplanýar we onuň bahasy sikli hasaplaýja berilýär. Mundan  
 soň sikl gaýtalanýar. Baha soňky bahadan uly bolan ýagdaýda  
 For operatory öz işini tamamlýar.

Mysal üçin, islendik bitin sanlary girizip, olaryň jemini  
 hasaplaýan maksatnama seredeliň:

```
procedure TfmExample.bbRunClick(Sender: TObject);
var
```

Segmentleriň registrleri

CS	CODE
DS	DATA
SS	STACK
ES	EXTRA



Çyz. 3.1 I8086/8088 prosessoryň huşyň fiziki salgysyny guralşy

Prosesoryň registrlaryň gurluşy aşakdaky çyzgyda getirilen. Umumy amally

Berlenleriň registrleri

AX	AH	AL
BX	BH	BL
CX	CH	CL
DX	DH	DL

Görkezijileriň we indeksleriň registrleri

SP	STACK POINTER
BP	BASE POINTER
SI	SOURCE INDEX
DI	DESTINATION INDEX

```

i,N,Sum : Integer;

begin
    try // Girizmäniň dogylygyny barlamaly
        N := StrToInt(edInput.Text);
    except // Indiki operatorlar ýalňyşlyk bar bolsa ýerine ýetirilýär
        ShowMessage('Bitin san girizmede ýalňyşlyk bar');
    dinput.SelectAll; // Ýalňyş girizmäni anyklaýarys
    Exit // Işi tamamlaparys
end;

edInput.Text := ' ';
edinput.SetFocus;
Sum := 0; // Sum ululygynyň başlangyç bahasy
for i := 1 to N
do // Jemi hasaplaýan sikl
    Sum := Sum+i;
mmOutput.Lines.Add('Bitin sanlaryň jemi
'+1..'+IntToStr(N)+IntToStr(Sum))+ 'deň');
end ;

Bu ýerde try (barlap görmek), except (aradan aýyrmak)
bölekler gorajjy blogy düzýär.
  
```

Ýokarky maksatnamany başgaça ýazyp bolýar:

```

Sum := 0;
if N >= 0 then
    for i := 1 to N do
        Sum := Sum + i
    else
        for i := -1 downto N do
  
```

Sum := Sum + i ;

### WHILE sikl operatory:

**while** <şert> **do** <operator>;

Bu ýerde while, do – ätiýaçlyk sözleri (entek, ýerine ýetirmeli)

### Meselem:

**procedure** TfmExample.bbRunClick(Sender: TObject) ;

**var**

Epsilon: Real;

**begin**

Epsilon := 1;

**while** 1+Epsilon/2>1 **do**

Epsilon := Epsilon/2;

IbOutput.Caption := 'Maşyn epsilony = ' +FloatToStr(Epsilon)

**end;**

### REPEAT... UNTIL operatory

**repeat** <sikliň bedeni> **Until** <şert>;

Bu ýerde repeat, until – ätiýaçlyk sözleri (gaýtalamaly, entek)

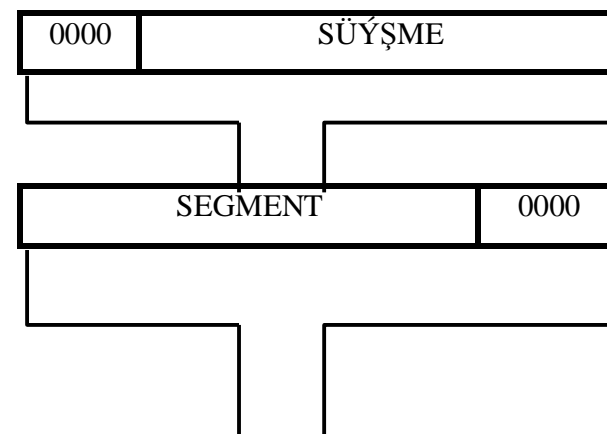
Sikl operatorlarynyň işleşini gowulandyrmak üçin break, continue paramametsiz operatorlar girizilen

**break** – siklden şo bada çykamak;

**continue** – wagtyndan ön siklden çykamak

Huşyň segment gurluşy bilen ýakyn we uzak salgylar düşünjeleri bagly bolup durýarlar. Ýakyn ýada içkisegment ýüzlenmede talap edilýan öýjüge barma diňe süýşmaniň görkezmeği bilen ýerine ýetirilýar, segmentiň salgysy bolsa degişli segment registriň häzirki mazmuny bilen kesgitlenýär. Uzak ýada segmentara ýüzlenmede doly salgı görkezilýar, ol degişli segment registra ýüklenýan segmentiň 16-baýtly bahasyny we 16-baýtly süýşmesini eýeleýar.

Processor huşyň bir baýtyna we iki sany baýtdan düzülýan söze ýada 4 sany baýtdan düzülýan ikilik söze ýüzlenip bilýar. Sözüň sag baýtyna kiçi baýt (Low, L) we çep baýtyna uly baýt (High, H) diýilýar. Huşda birinji kiçi baýt soňra uly baýt ýerleşýärler. Huşyň segment gurluşy we baýtlaryň (L,H) tertibi Intel prosessorlaryň başga prosessorlardan tapawutly häsiýetleri bolup durýar.





I8086/8088 prosessorlar üçin huş *baýtlaryň* çyzyk ýzygiderligi görmüşinde göz önüne getirilýar. Huşa ýüzlenmek üçin prosessor (daşky çyzgy bilen bilelikde) MEMWR# (Memory Write) we MEMRD# (Memory Read) şina signallary düzýar ýazmak we okamak amallar üçin degişlilikde. 1 Mbaýt salgylar giňişlige barmak üçin 20 razrýadly salgy şinasy ulanylýar. Logiki taýdan huş 64 Kbaýt ölçegli segmentlere (böleklere) bölünýar. 20 bit razrýadly salgylar şina gelýan huşyň fiziki salgy iki sany 16-bitli bölümden ybarat. Bu segmentiň salgysy (Seg) we ýerine ýetirilýan salgy EA (executive address). Olar 4 bita süýşme bilen jemlenýärler. Segmentiň salgysynyň 4 bita çepe süýşmesi onuň 16 köpeltme hasylyna deň. Şonuň üçin fiziki salgy (PA):

$$PA = 16 \times Seg + EA$$

Segmentiň salgysy CS, DS, SS we ES registrlaryň birinde ýerleşýar.

Ýerine ýetirilýan salga başgaça effektiw salgy diýilýar. Ol konstanta, registrleriň mazmuny, huşyň öýjüginin mazmuny ýada ençeme ululyklaryň jemi bolup bilýar. Emma bu jemi 16-razrýadly bolýar (sýüşme hasap edilenok). Şeýlelikde, fiziki salgy hiç haçan 64 Kbaýt segmentiň çäginde çykyp bilmez.

### Saýlama operatory

**case** <saýlama açary> of <saýlamak üçin spiok> [ **else** <operatorlar>] **end;**

**Meselem:**

**var**

ch : Char;

**begin**

**case** ch of

'n','N','H','H': IbOutput.Caption := 'Ýok';

'y','Y','д','Д': IbOutput.Caption := 'Hawa';

**end;**

**end.**

#### **4. C++ dilinde funksiýalar. Funksiýalary ýazmak. Funksiýalary kesgitlemek.**

Biri-biri bilen informasiýa alyş-çalyşyny etmek üçin adam tebigy dilleri (Türkmen, Inlis, Hytaý we ş.m.) ulanýar. Diliň esasynda elipbiý durýar. Elipbiýiň simwollarynyň yzygiderliligi grammatiki düzgünlere laýyklykda sözleri emele getirýär. Sözlerden düzgüne laýyklykda sözlemleri emele getirmegiň kadalaryna sintaksis diýilýär. Tebigy dilleriň grammatikasy we sintaksisi köpsanly düzgünler bilen formulirlenýär. Bu düzgünlerden gyşarmalar hem bolýar. Gyşarmalaryň döreýşi taryhydyr.

Tebigy diller bilen birlikde formal diller döredilendir. Olara hasaplaýyş sistemasy, algebra dili, programmirleme dilleri we başgalar degişlidir. Formal dilleriň tebigy dillerden tapawudy- formal dillerde grammatikanyň we sintaksisiň berk düzgüniniň barlygydyr.

Meselem, hasaplaýyş sistemalary, diňe bir elipbiýi (şifrleri) bar bolan sanlary (objektleri) atlandyrmaga we ýazmaga mümkinçilik berýän formal dil dälde, olaryň üstünde berk kesgitli düzgünler esasynda dürli operasiýalary geçirmäge mümkinçilik berýän formal dildir.

Käbir formal diller harplary ýa-da sifrleri ulanman, başga belgileri ulanýar. Meselem ýol belgileri, saz notalary, himiki formulalar, morze elipbiýi we ş.m. Belgileriň dürli tebigaty bolup bilerler. Meselem informasiýa hat formasynda bolanda kagyzyň ýüzünde şekillendiren belgiler ulanylýar. Gepleşik dilinde diliň belgisi hökmünde, dürli sesler (fonemler) ulanylýarlar. Kompýuterde tekstler täzeden işlenende belgiler hökmünde, elektrik impulslarynyň yzygiderliligi peýdalanylýar.

baglanşykly. Maksatnamanyň nukdaý nazarynda bu prosessorlar birmeňzeş, olaryň registerleriniň toplумы we buýruklar toplумы ähli PC bilen ylalaşykly işleýän ähli prosessorlara girizilen. Maşgalanyň başlangyjyndan – 8086 prosessordan – bütin maşgalanyň belgilemesi kesgitlendi. x86 prosessorlar şinany dürli prosessorlar we kontrollerler bilen bilelikde ulanmaga, hem-de apparat we maksatnama tarapyn ýerine ýetirmäge mümkinçilik berýär. Ondan başgada hasaplama öndürijiligini ulaldýan 8087 matematiki soproessory ulanmak göz önünde tutulýar.

Prossessorlarda konweýer arhitekturasy ulanylan. Ol huşdan buýruklaryň belgilerini saýlap almagy we ony belgisizdirlemegi içki amallar edilende ýerine ýetirýär. Konweýer olaryň operasion düwünlerinde saklanma wagty azaldyp, prosessoryň öndürijiligini ulaldýar. 8086 prosessorlaryň konweýerleriniň içki buýruklar hatary 6 – baýtly. Prosessorda 16 bit razrýadly 14 sany registrler bar, operandlar 8 ýada 16 bit razrýadly bolup bilýärler, belgili we belgisiz ikilik we ikilik-onluk sanlary göz önüne getirip bilýärler. Buýruklar ulgamy operandlary salgylamaga 24 sany režimlary eýeleýär. Buýrugy ýerine ýetirmäge ortaça wagt sinhronizirlemäni eýeleýär. Birinji I8088 prosessoryň ýygylgy 4,77 MGs deňdi, soňra 8 we MGs takt ýygylkly prosessorlar peýda hem boldylar (Turbo-XT kompýuterlerde ulanylýadylar).

## 12. Mikroprosessorlarda signallaryň strukturasy.

### Signallaryň klassiki gurluşy.

Häzirki döwürde 16-razryadly 8086/88 we 80286 prosessorlar könelşen görünýär. Emma muňa garamazdan PC-laryň maksatnama üpçünçiliginiň aglaba bölegi häzir hem 16-razryadly prosessorlaryň režiminde işleýärler we olar bilen ylalaşykly. Şonuň üçin olaryň işleýşiniň aýratynlygyny öwrenmek PC-nyň işleýşine düşünmek üçin gerek. Intel firmasy 1977-nji ýylda 8088 prosessory çykardy. Ol 16 bitli prosessorlaryň 1-jni nesline degişli we 1978-nji ýylda çykarylan 8086 prosessoryň özgerdilen görnüşi bolup durýar. Bu prosessorlaryň ikisem 8/16 bitly logiki we arifmetiki amallary, şol bir wagtyň özünde köpeltmegi we bölmegi, setirler bilen amallar we giriş/çykyş amallary ýerine ýetirýärler.

Prossessorlar 1 Mbaýt ýady salgylamaga mümkinçilik berýän 20-razryadly bolýarlar, 8086 prosessorlaryň maglumatlar şinasy 16 razryadly, 8088-de bolsa maglumatlaryň daşky şinasy 8-bita çenli gysgaldylan. Bu kemeldilme ulgamy arzanlaşdyрма maksady bilen edildi we ol öndürijiligiň peselmegine eltdi. 8086 – şinasynyň uly razryady bolany üçin 8088-den şol bir takt ýygylkda 20-60% çalt işleýär. Bu prosessorlaryň şinalarynyň razryadlarynyň dürliligi bilen şertlenen funksional tapawut diňe 8 we 16 razryadly daşgy gurluşlaryň birleşdirilmegi bilen

Uly göwrümlü maksatnamalar döredilende proseduralary we funksiýalary ulanmak zerurlygy çykýar.

Object Pascalda prosedura bolup aýratyn görnüşde şekillendirilen maksatnamanyň bölegi hyzmat edýär. Ol hususy ada eýe. Onuň maksatnamada ady tutulmagy prosedurany aktiwizirleýär we bu ädime prosedurany çagyrmak diýilýär. Çagyrylandan soň prosedura girýän ähli operatorlar ýerine ýetirilýär we dolandyryş çagyran maksatnama bölegine gaýtarylýar.

Proseduralar bilen biz eýýäm duşupdyk - olar Exit, ShowMessage. Biz öz proseduramyzy düzeliň: goý käbir setiriň setir harplary baş harplara özgerdilsin:

**procedure** TfmExample.bbRunClick(Sender: TObject);

**procedure** UpString(stinp: **String**; **var** stOut: **String**);

*{Bu prosedura stinp setiriň harplaryny özgerdýär we stOut setire ýerleşdirýär }*

begin

stOut := stinp;

**end**; // UpString **var**

SI, S2: **String**;

**begin**

SI := edinput.Text; // Başlangyç setiri alýarys

UpString(SI,S2); // Özgerdýäris

IbOutput.Caption := S2; // Netijäni çykarýarys

edinput.Text := " ;

edinput.SetFocus ;

**end** ;

Proseduranyň ýene bir görnüşi:

```
Procedure UpString(stinp: String; var stOut: String);  
var  
k: Integer; // Sikliň parametri  
begin  
    stOut := stinp;  
    for k := 1 to Length(stOut) {Length –setiriň uzynlygy}  
    do  
        begin  
            stOut[k] := UpCase(stOut[k]); // Latinisany özgerdýäris  
            if stOut[k] >= 'a' then // Setir harpy kirillisamy?  
            stOut[k] := // Да: ony özgerdýäris  
                Chr(ord('A') + ord(stOut[k]) - ord('a'));  
            if stOut[k]='e' then  
                stOut[k] := 'Ё'; // ё-ny Ё өзgerdýäris  
            end;  
        end; // UpString  
    Bu maksatnamany başgaça funksiýany ulanyp ýazalyň:  
    procedure TfmExample.bbRunClick(Sender: TObject) ;  
    Function UpString(stinp: String): String;  
    var  
    k: Integer; // Sikliň parametri  
    begin  
        Result := stinp;  
        // Harplar boýunça özgertmek  
        for k := 1 to Length(Result) do  
        begin
```

$$iCOMP^{\text{®}} \text{ Index2.0} = 100 \times \left( \frac{BM_1}{Base\_BM_1} \right)^{P_2} \times \left( \frac{BM_1}{Base\_BM_1} \right)^{P_2} \times \dots$$
$$\dots \times \left( \frac{BM_2}{Base\_BM_2} \right)^{P_n}$$

Pentium 120 MGs	1000
Pentium 133 MGs	1110

1996 - nji ýylda ýenede täze ölçeg birligi kabul edildi - iCOMPTM Index 2.0. Bu ölçeg birligi öňkiden koeffisientleriň we görkezijileriň toplumy bilen, seýs prosessoryň saýlawy bilen tapawutlanýarlar.

Jedwel 2.2 öndürijiligiň kesgitlemegiň agram koeffisientleri.

iCOMP® Index	Agram %
PC Bench 7.0.1	68
SPECint92	25
SPECfp92	5
Whetstone	2
Norton SI32	15
CPUmark32	40
SPECint_base95	20
SPECfp_base95	5
Intel Media Benchmark	20

Öndürijilik indeksleri üçin formula:

```

Result[k] := UpCase(Result[k]); // Latinisany özgerdýäris
if Result[k] >= 'a' then // Setir harpy kirillisamy?
    Result[k] := // Hawa: ony özgerdýäris
    Chr(ord('A') + ord(Result[k]) - ord('a'));
if Result[k]='e' then
    Result[k] := 'Ё'; // ё-ny Ё özgerdýäris
end;
end; // UpString
begin
    lbOutput.Caption := UpString(edinput.Text);
    edinput.Text := "";
    edinput.SetFocus ;
end;

```

### Toplumlar

Object Pascalda ulanylýan toplumlar beýleki dillerdäki ýalydyr.

Meselem:

```

type
    digit = array [0..9] of Char;
    matrix = array [byte] of Single;
var
    m : matrix;
    d : digit;
    i : integer;
begin
    m[17] := ord(d[i-1])/10;
end.

```

Toplumyň ýazgysy şu görmüşde bolýar :

<Tipiň ady>= **array** [ <ind.tipl.spisogy> ] **of** <tip>;

Meselem:

**var**

a,b : **array** [1..10] of Real;

**type**

mat = **array** [0..5] **of array** [-2..2] **of array** [Char] **of** Byte;

Ýa-da:

**type**

mat = array [0..5,-2..2,char] **of** Byte;

Eger, mysal üçin,

**var**

a: **array** [1..2,1..2] **of** Byte;

**begin**

a[1,1] := 1;

a[2,1] := 2 ;

a[1,2] := 3;

a[2,2] := 4;

**end.**

haýsydyr bir prosessoryň göräli kuwatyny ýönekeý usu bilen kesgitlemäge mümkinçilik berýär.

Jedwel 2.1 ICOMP Index - öndürijilik indeksleri

Prossessorlar	ICOMP Index
I1386SX-20	32
I1386SX-25	39
I1386DX-25	49
I1386SX-33	56
I1386DX-33	68
I1386SX-20	78
I1386SX-25	100
I1386DX-25	122
I1386SX-33	136
I1386DX-33	166
I1386DX2-50	231
I1386DX-50	249
I1386DX2-66	297
Pentium 60 MGs	510
Pentium 66 MGs	567
Pentium 75 MGs	610
Pentium 90 MGs	735
Pentium 100 MGs	815

ýerine ýetirilmeli tertibi möhüm bolsa onda amallaryň ýerine ýetirilşini bölümleşdirip bolar.

Prossessoraryň öndürüjilikleriniň deňeşdirmek üçin dürli ölçeg usullary ulanylýar. Şol bir arhitekturasy bolan (ýa-da buýruklar ulgamu meňzeş we operantlarynyň razrýadlary gabat gelýän) prossessorlary deňeşdirmek üçin onuň wagt birliginde ýerne ýetirýän amallarynyň ortaça sanyna deň bolan görkeziji ulanylýar. Mümkin bolduguça ýadyň çaltlygyna baglanmazlyk üçin, operasiýa diýilende registerleriň operandy göz önünde tutulýr. Bu prinsipde MIPS(Mega Instruction Per Second) ölçeg birligi guralan. 8086/88 we 80286 prossessorlar ulanylýan wagtlyary olaryň öndürüjiligi (görmüşini görkezmek bilen bilelikde) ýeterlik aýdyňlykda prossessorýň takt ýygylygy bilen häsiýetlendirip bolar. Soňky döwürde çykan kompýuterleriň arhetekturasynda uly özgerşikler bolup geçdi. Onuň netijesinde eýýäm takt ýygylyk prossessorýň öndürjiligini häsiýetlendirýän ululyk bolmagyndan galdy.

Otur bilen ýazylan sanlaryň üstinde ýerine ýetirilýän hasaplamalar bilen işleşenilende MIPS birlige derek FLOPS(Floating point Operations Pes Second) birligi ulanylýar. Arhitekturasy 80x86 bolan 32 - bitli prossessorlaryň öndürjiligini deňeşdirmek üçin Intel firmasy 1992 ýylda özüniň ölçeg birligini hödürledi - iCOMPTM (Intel Comparative Microprocessor Performance). Bu ölçeg belli bir sany ulanyja

Iki toplumy elementleri boýunça deňeşdirip bolýar, meselem:

```
var  
a,b : array [1..5] of Single;  
eq : Boolean;  
      i : Byte;  
begin  
      eq := True; for i := 1 to 5 do  
      if a[i] <> b[i] then  
      eq := False/if eq then  
      end.
```

## 5. C++ dilinde massiwler bilen işlemek. Bir ölçeqli massiwleri ulanmak bilen radiofiziki meseleleri programmirmek.

Dürli obýektleri özgertmek prosesleri algoritmeleriň üsti bilen ýazylyp bilner. Durmuşda hasaplaýyş algoritmeleri giň ýaýrandyrlar. Algoritm sözi IX asyrdan ýaşan türkmen alymy Al-Horezminiň adynyň latynça aýdylşydyr.

Mysal hökmünde tekst görnüşindäki obýekti redaktirlenmegiň algoritmine seredeliň.

Obýektiň halyny üýtgetmek üçin onuň üstünde käbir operasiýalary ýerine ýetirmeli. Şol operasiýalary ýerine ýetirýän obýekte **ýerine ýetirijiler** diýilýär. Ýerine ýetirijiler hökmünde adam, kompýuter we başgalar bolup bilerler.

Algoritm aýratyn komandalardan durýar. Ýerine ýetiriji olary biri-biriniň yzyndan belli bir yzygiderlikde ýerine ýetirýär. Algoritmleşdirilende informasion prosessi aýratyn komandalara bölüşdirmek algoritmiň wajyp häsiýetidir we oňa algoritmiň **diskretlilik** diýilýär.

Teksti özgertmek prosesini aýratyn operasiýalara bölüşdirmeli we olar ýerine ýetiriji üçin aýratyn komandalar görnüşinde ýazylmaly. Her bir ýerine ýetirijiniň öz ýerine ýetirip bilýän komandalar sistemasy bar. Algoritm diňe ýerine ýetirijiniň komandalar sistemasyna girýän komandalary özünde saklamalydyr.

Tekst redaktirlenende, dürli operasiýalaryň bolmagy mümkin. Öçürme, kopýalama, süýşürme, fragment çalyşma we ş.m.

Algoritmiň ýazylyşy şeýle bolmaly, ýagny ýerine ýetiriji nobatdaky komandany ýerine ýetirip, soňra, indi haýsy komandany ýerine ýetirmelidigini bilmeli. Algoritmiň bu häsiýetine **determinirlilik** diýilýär.

Algoritm gutarnykly ädimlerde obýektiň başky haldan ahyrky halyna öwürilmegini üpjün etmeli. Algoritmiň bu häsiýetine **netijelilik** diýilýär. Şunlukda, redaktirleme

Giriş/çykyş portlaryna ýüzlenende kompýuteriň BIOS-y maksatnama arakesmelerni döretmek üçin 8088, 286 we 386 prosessorlarynda indiki salga şertsiz geçiş buýruklar ulanylýar. Bu komanda konweýeri (debelgiirlenen görkezmeleriň yzygiderligi) taşlaýar we kompýuter amalyň belgisini ýene ýaddan alýar, bu wagt bolsa indiki ýüzlenmä çenli port "dynç alýar". Beýle arakesmeler usuly diňe BIOS-da ulanylman, eýsem ýüklenilen maksatnamalarda hem ulanylýar. Ýöne beýle maksatnama arakesmeleri içki keşli (486 prosessorlar we 386-yň käbir kysymleri) bolan prosessorlar üçin deňeşli däl. Bu ýerde öň ýada ýüzlenmegiň daşky aýlawy döredýän JMP boýruk, içki keş toparyndan alnar we garaşylýan arakesme bolmaz. Çykyş aýlawlaryň arasynda şinanyň daşky aýlawyny girizmegiň bir usuly ýadyň keşirlenmedik ýalňyşyny okamakdan durýar. Görkezmeleri tertip boýunça ýerine ýetirýän prosessorlarda bu amal diňe öňdäki çykyş aýlawynyň gutarmagy bilen bolýar, indiki çykyş amaly bolsa ýatdan okamagyň gutarmagy bilen başlanýar.

Soňky nesilleriň prosessoryndaky kompýuter arhitekturasynyň kömeg bilen her ädimde birnäçe görkezmeler bolýar. Altynjy nesiliň prosessorynda ýerine ýetirilmeli amallaryň tertibi üýtgäp biler we ýatdan okamak daşky şinadaky beýleki amallardan öň ýerine ýetirilip biliner. Eger amallaryň



operatiw huşa öňki ýukläp bilýar. Şeýle operatiw huşyň göwürümi diskiň hasabyna ulaldyp bolýar ýöne onda huşuň orta çalt işleýişi peselýar.

## 11. Mikroprocessor sistemalary. Mikroprocessor sistemalarynda analog we sanly signallar.

Belli bolşy ýaly prosessorlaryň x86 maşgalasyndaky ähli prosessorlar özünden öňki çykan prosessorlar bilen ylalaşykly işleýär. Bu bolsa 8088 prosessor üçin ýazylan maksatnama belgiy, 386 we Pentium Pro prosessorlarynda hemde islendik olar bilen ylalaşykly prosessorlarda işleýänligi aňladylýar. Bu köplenç şeýle hem bolýar, emma soňky çykan prosessorlarda maksatnamanyň ýerine ýetiriş çaltlygy uly bolýar. Edil şu ýerde tizlik bilen baglanşyklylykda prosesosyryň ylalaşykly işleýişi barada bir göze ilmän duran zat bar. Personal kompýuterler üçin ýazylan komýuterlaryň köpüsi diňe hasaplama geçirmän eýsem içki we daşky gurluşlary dolandyryrlar. Bu ýerde gurluşlar gatnaşygyň belli bir tertibini we wagtyň dowamlylygyny talap edýärler. Eger gurluşyň tizligi kiçi bolsa onda ol oňa ýüzlenendäki maglumatlar bilen işläp bilmeýär. Şu nukdaý nazardan gurluşlaryň kysymy barada maglumatlar gerek. Şu baglanşykda arakesmelerin wagty kesgitlenilýär.

netijesinde tekstdäki başlangyç simwollaryň yzygiderliligi simwollaryň ahyrky yzygiderliligine öwürülmelidir.

Şol bir algoritmiň, dürli başky berlenler bilen ulanmak mümkinçiligi bolmalydyr. Bu häsiýete, algoritmiň **köpçülikleýinligi** diýilýär

C++programmirleýji ulgamyň ýönekeý komponentalaryny ulanyp berlenleri girizmek-çykarmak üçin wiziual gurşawyň esasy elementlerini öwreneliň.

### C++ -niň wiziual gurşawy

DELPHI işe girizilenden soň interfeýsiň paneli ekrana çykýar. Onuň ýüzünde birnäçe iş penjiresi şekillendirilýär. Olaryň mukdaty, ýerleşýän ýeri, ölçegleri programmist tarapyndan üýtgedilip bilner.

Esasy penjire ekranda elmydama durýar. Onuň kömegi bilen goşundy döretmek prosesi dolandyrylýar.

Esasy menýu proekt taýýarlamak üçin gerek bolan xerur serişdeleri saklaýar.

Esasy menýunyň piktogramalary köp ulanylýan buýruklary almaklygy ýeňilleşdirýär.

Komponentalaryň Palitrasy DELPHI kitaphanasynda ýerleşdirilen maksatnamalara barmaklygy amala aşyrýar. Komponentalaryň häsiýetnamalary HELP bölümde ýerleşdirilen.

Obýektleriň Inspektorynyň penjiresi saýlanan komponentalaryň häsiýetlerini üýtgetmek üçin peýdalanylýar we iki sahypadan durýar: *Properties* (Häsiýetleri), *Events* (Hadysalar).

Formanyň penjiresi döredilýän Windows-goşundynyň interfeýsi bolup hyzmat edýär. Onuň yüzüne ulanyljak komponentalar ýerleşdirilýär. Proektiň her bir formasyna Unit moduly degişlidir, onuň Pascal dilindäki teksti Kodyň redaktorynyň penjiresinde ýerleşdirilýär.

Kodyň redaktory penjire proektiň modulynyň tekstini seretmek, döretmek we düzetmek üçin niýetlenen.

Hadysa	Hadysanyň ýazgysy
OnActivate	Forma aktiwizirlende ýüze çykýar.
OnCreate	Forma döredilende ýüze çykýar.
OnClick	Syçanjygyň düwmesini komponentiň üstünde basylanda ýüze çykýar.
OnDbClick	Syçanjygyň düwmesini komponentiň üstünde iki gezek basylanda ýüze çykýar.
OnKeyPress	Klawiaturada düwme basylanda ýüze çykýar.
OnKeyDown	Klawiaturada düwme basylanda ýüze çykýar.

Her hadysa üçin modulyň tekstinde *procedure* döredilýär we programmist *begin* we *end* sözleriň arasyna gerek algoritmi ýazýar. Forma bilen Kodyň redaktory penjireleri çalyşmak üçin F12 düwme basylýar.

Mesele :Aşakdaky aňlatmany hasaplamak üçin Widows-goşundy döretmeli

$$u = tg^5(\sqrt{x} - y^3) + e^{y/z} \cdot \sin z^2$$

X, y we z ululyklaryň bahalaryny klawiaturadan girizmeli.

Proýekti ýatda saklatjak bolsaň esasy menýüdaki File punkty almaly we Save Project As... Proýekte belli bir at dakmaly.

UnLinAlg modulyň teksti  
**Unit** UnLinAlg;

**interface**

ulanylyar. Faýllaryň mazmunyna prosessor (maksatnama) diňe olaryň operatiw huşyň käbir ýaýlasyndaky şekillendirmesi arkaly baryp bilýar. Gös-göni diskda ýerleşýan maksatnama belgiy ýerine ýetirmegi ýa-da maglumatlara ýuzlenmegi prosessor aslynda edip bilenok. Bu lenta huşyň hem degişli. Emma tejribelikde disk we operatiw huşy özara çylşyrymly kesişýärler. Disk huşyň esasy ýetmezçilikleri- uly barma wagty we çalyşmalaryň pes tizligi-wertikal disk ulanylanda aýyrlýarlar. Wertikal disk-bu operatiw huşyň ýörite ulanylyan ýaýlasy. Bu ýaýlada faýllar saklanýar we opersion ulgaamyň we amaly maksatnamanyň nukdaý nazaryna görä ol hemişelik, emma örän çalt disk hökmünde özüni alyp barýar. Wertikal disk köp ýagdaýlarda kompýuteriň işleýişiniň tizligini güýçli disk çalyşmalarda ýokarlandyryp bilýar.

Operatiw huşyň esasy ýetmezçiligi - onuň göwrümi köp esse diskiň göwrüminden az. Disk huşyň hasapyna operatiw huşyň göwrümini ulaltmak meseläni çözmek üçin wertikal huş ulanylyarlar. Wertikal huş-bu diskda operatiw huşy keşirlemek. Keşirlemegiň manysy indikiden ybarat. Maksatnamalar kompýuterde fiziki operatiw huşyň göwrüminden uly bolýan operatiw huşyň giňişligini peýdalanyň bilýärler. Bu wertikal giňişlik kesgitli ölçegli sahypalara bölünen, fiziki operatiw huşda her wagtda olaryň diňe bir bölegi ýerleşýär. Galan sohypalar diskda saklanýar, olary operasion ulgam fiziki

Huşyň ulgamy üçin indiki häsiýetnamalar möhüm bolýar.

1. Saklanýan maglumatlaryň göwrümi näçe ol uly bolsa, sonça gowy. Iň uly göwrümi lenta we disk gurluşlar (çalşyrylýan göterijilerli) eýelearler, onsoň disk huşlar we soň operatiw huş.
2. Barma wagty - maglumatlara talap edilen soň maglumatlary iberip başlamagynyň arasyndaky wagtyň dowamy. Iň kiçi ýuzlenme wagty operatiw huşy eýeleýar, soň disk huşy, soň lenta huşy.
3. Maglumatlaryň geçirmeginiň çalşyрма tizligi. Iň uly tizligi operatiw huş eýeleýar, soň disk huşym soň - lenta huşy.
4. Maglumatlaryň birliginiň saklamagyň udel gymmaty toplaýjyň bahasy saklamagyň birligine (baýta ýa-da megabaýta) gatnaşdyrylan. Iň kiçi saklanma gymmaty çalşyrylýan göterijili lenta gurluşlar eýeleýärler, soňra disk toplaýjylar, iň gymmat-operatiw huşy.

Içki we daşky huşlar düýpden dürli usullar arkaly ulanylýarlar. Içki huşda (operatiw we hemişelik) maksatnamalar saklanýar. Olary prosessor gös-göni ýerine ýetirip bilýar. Onda hem prosessor we ýerine ýetirilýan maksatnama gös-göni baryp bilýerleşýar. Daşky huş islendik görnüşli faýllary saklamaga

## uses

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

## type

```
TForm1 = class(TForm)
    Label1: TLabel;
    Edit1: TEdit;
    Label2: TLabel;
    Edit2: TEdit;
    Label3: TLabel;
    Edit3: TEdit;
    Label4: TLabel;
    Memo1: TMemo;
    Button1: TButton;
procedure FormCreate(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
```

## var

Form1: TForm1;

## implementation

```
{ $R *.DFM }
// Forma döretmek hadysasynyň işlenişprosedurasy:
procedure TForm1.FormCreate(Sender: TObject);
begin
    Edit1.Text:='0.4';           // X başlangyç bahasy
    Edit2.Text:='8.75e-2';       // Y başlangyç bahasy
```

```

Edit3.Text:='-19.63'; // Z başlangyç bahasy
Memo1.Clear; // Memo1 arassalamk
// Memo1 setir çykarmak:
Memo1.Lines.Add('Çyzykly algoritim ');
end;
// Button1 düwmäni basmak:
procedure TForm1.Button1Click(Sender: TObject);
var
  x,y,z,a,b,c,u : extended; // lokal üýtgeýän ululyklar
begin
  x:=StrToFloat(Edit1.Text); // X-e Edit1 bahasy berilýär
  y:=StrToFloat(Edit2.Text); // Y-e Edit2 bahasy berilýär
  z:=StrToFloat(Edit3.Text); // Z-e Edit3 bahasy berilýär
  // Aňlatma hasaplanýar
  a:=sqrt(x)-y*y*y;
  b:=sin(a)/cos(a);
  c:=Exp(5*Ln(b));
  u:=c+exp(y/z)*sin(z*z);
  Memo1.Lines.Add('X = '+Edit1.Text+' Y = '+Edit2.Text+'
    ' Z = '+Edit3.Text);
  // Memo1 netije çykarylýar:
  Memo1.Lines.Add(' U = '+FloatToStrF(u,ffFixed,8,3));
end;

end.

```

Kompýuter huşuň iki görnüşini bolýar - içki we daşky. İçki huş - elektron (ýarymgeçiriji) huş, sistem ýa-da giňeltme platalarda ýerleşýar.

Daşky huş - maglumatlary saklamagyň dürli prinsipli enjamlar görnüşine ýerine ýetiribin huş we köplenç hereketlenýän göterijiler bilen. Häziriki zaman muňa magnit (disk we lenta) huşuň, optiki we magnitoptiki huşuň gurluşlary girýar.

Prossessor göni diňe içki huşa ýüzberip bilýar, maksatnama tarapyn berlen salgı boýunça. İçki huş üçin bir ölçegli (çyzyk) salgı ulanylýar, bu kesgitli zarýady ikilik san. İçki huş iki gömüşe bölünýar: operatiw we hemişelik huşlar. Operatiw huşdaky maglumatlar prossessor arkaly islendik wagtda üýtgedilip bilýar. Operatiw huşuň öýjüklerine ýüzlenme islendik tertipli ýerine ýetirip bolýar, okamakda we ýazmakdada. Daşky huş çylşyrymlyrak salgylanýar. Onuň her öýjiginiň käbir blogyň içinde öz salgysy bar. Salgıň özi hem köpölçegli bolýar. Maglumatlar çalşyrmalaryň fiziki akalaryň ýerine ýetirilen her blok bütinleýin ýazylyp ýa-da okalyp bilýar. Bir disk üçin blogyň salgysy üç ölçegli bolýar, üstini (kellejigini) belgisi, silindriň belgisi we sektoryň belgisi. Kompýuterde disk huşlaryň sany köp bolmagynyň sebäbi üçin huşy salgylanmaga diskiň belgisi we interfeýsiniň kanalyň belgisi gatnaşýarlar.

izygiderlikde huşdan buýruklary saýlap alýar we olary ýerine ýetirýär. Prosessoryň buýruklary huşuň giňişliklerinde we giriş/çykyş portlarda ýerleşýän berlenleri derňemek, gaýtadan işlemek we geçirmek üçin niýetlenen. Kompýuterde esasy maksatnamany ýerine ýetirýän merkez prosessor (CPU – Central Processing Unit) hökman bar bolmaly. Kompýuterde köplenç merkez prosessora kömek üçin soproessorlar girizilýärler, olar käbir aýratyn meseleleri amala aşyrmaga niýetlenen. Ýüzýan nokatly formatdaky san maglumatlary has netijeli gaýtadan işleýän matematiki soproessorlar, grafik şekilleri gaýtadan işleýän we geometriki gurluşlary ýerine ýetirýän grafik soproessorlar, daşky gurluşlar bilen giriş-çykyş amallary ýerine ýetirýän giriş-çykyş soproessorlar giňden peýdalanýar.

Kompýuteriň huşy maglumatlary – buýruklaryň we berlenleriň belgilaryny - az wagtlap we köp wagtlap saklamaga niýetlenen. Huşda maglumatlar ikilik belgilarda (bitlerde) saklanýar, her bit – huşuň iň ýönekeý öýjügi – “0” ýada “1” bahalary kabul edip bilýar. Her öýjügiň öz salgysy (adresi) bar, ol kesgitli koordinatalar ulgamynda degişli öýgügi birbahaly belleýar. Huşda maglumatlaryň iň kiçi salgylanýan saklanma birligi bolup adatyň baýt bolýar, ol 8 bitdrn düzülýar.

## 6. C++ diliniň grafiki mümkinçilikleri. C++ diliniň grafiki operatorlary.

Şahalanýan algoritmleri programmirmlemede ýönekeý geçiriji komponentalary ulanyp Windows goşulma döretmeli. Onuň üçin şu aşakdaky meselä seredeliň:

Mesele: görkezilen aňlatmany işlemek üçin Windows goşundy döretmeli:

$$Z = \begin{cases} f(x), & x < y \\ y, & \text{basgaca} \end{cases}, \quad \text{bu ýerde}$$

$$f(x) = \begin{cases} \sin(x) \\ \cos(x) \end{cases} \text{ ulanyjynyň}$$

islegi boýunça. Interfeýsiň panelinde başlangyç bahalary çykarmak mümkinçiligini göz önünde tutmaly.

### Formada komponentalary ýerleşdirmek

DELPHI goşundylar döredilende geçiriji düwme görnüşindäki komponentalar köp ulanylýar. Düwmäniň ýagdaýy (öçürilgi/ýakylgy) formada görüňän şekilde peýdalanylýar. Biz CheckBox we RadioGroup düwmeleri ulanmaly bolýarys.

**CheckBox** komponentasy garaşsyz düwmäni gurnaýar. **RadioGroup** komponentasy garaşly geçirijiler bolan düwmeleriň toparyny gurnaýar.

Forma Label, Edit ı Memo komponentalary ýerleşdirip



Standard sahypadan komponentany alyp Forma ýerleşdirmeli. Caption setirde **CheckBox1** ýazgyny **Berlenleriň barlag üçin çykarylyşy** ýazgy bilen çalyşmaly. Чтобы при

запуске приложения кнопка CheckBox  $\hat{e}\hat{a}\hat{c}\hat{a}\hat{e}\hat{a}\hat{n}\hat{u} \hat{a}\hat{e}\hat{e}\hat{p}\hat{d}\hat{a}\hat{i}\hat{a}$ ,  $\hat{c}\hat{a}\hat{i}\hat{e}\hat{n}\hat{o}\hat{a}\hat{i}$  Checked уснановиѣ равным True.



Standard sahypadan piktogrammany alyp, ony formanyň gerek ýerinde goýmaly. Caption häsiýetinde **RadioGroup1** adyny **f(x)** at bilen çalyşmaly. Bir sütünde ýerleşer ýaly Columns häsiýeti 1 deň etmeli. Items häsiýete iki sapar urup setir redaktorynda iki setirde -  $\sin(x)$  we  $\cos(x)$  ýazyp OK düwmäni basmaly.

**UnRazvAlg modulyň teksti**

**Unit** UnRazvAlg;

**interface**

**uses**

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

**type**

```
TForm1 = class(TForm)
  Label1: TLabel;
  Edit1: TEdit;
  Label2: TLabel;
  Edit2: TEdit;
  Label4: TLabel;
  Memo1: TMemo;
  Button1: TButton;
  RadioGroup1: TRadioGroup;
  CheckBox1: TCheckBox;
procedure FormCreate(Sender: TObject);
procedure Button1Click(Sender: TObject);
private
  { Private declarations }
```

**begin**

vper:=MZap[j-1];

MZap[j-1]:=MZap[j];

MZap[j]:=vper;

**end;**

**for** i:=1 to 9 do

**with** StringGrid1,MZap[i] do

**begin**

Cells[1,i]:=fio;

Cells[2,i]:=IntToStr(mat);

Cells[3,i]:=IntToStr(fiz);

Cells[4,i]:=IntToStr(soch);

Cells[5,i]:=FloatToStrF(srbal,ffFixed,5,2);

**end;**

**end;**

## 10. Kompýuteriň esasy gurluşlary.

### Kompýuterleriň funksional bölekleri.

Kompýuter – bu maksatnama arkaly berlen buýrukларыň izygiderligini ýerine ýetirip bilýan gurluş. Islendik kompýuter üç sany esasy düzüjileri eýeleýär: prosessor, huş we goşmaça gurluşlar. Olar şinalar arkaly özara täsir edýärler.

Processor esasy “beýni” düwün bolup durýar, onuň esasy wezipesi huşda saklanýan maksatnama belgiy amala aşirmek. Häzirki zamanda prosessor diýip mikroprocessory aňladýarlar – bu mikroçyzgy diňe processoryň özüni dälde başga gurluşlary hem saklap bilýar, mysal üçin käş-huşy. Processor kesgitlenen

```

Cells[1,7]:= 'Ýedinji
Q.Q.';Cells[2,7]:='5';Cells[3,7]:='5';Cells[4,7]:='4';
Cells[1,8]:='Sekizinji
Cells[2,8]:='5';Cells[3,8]:='5';Cells[4,8]:='5';
Cells[1,9]:='Dokuzynjy
D.D.';Cells[2,9]:='3';Cells[3,9]:='5';Cells[4,9]:='5';
for i:=1 to 9 do
  with MZap[i] do
    begin // ýazgylaryň toplumynyň meýdanyny döretmek
      fio:=Cells[1,i];
      mat:=StrToInt(Cells[2,i]);
      fiz:=StrToInt(Cells[3,i]);
      soch:=StrToInt(Cells[4,i]);
      srbal:=(mat+fiz+soch)/3;
      Cells[5,i]:=FloatToStrF(srbal,ffFixed,5,2);
    end;
  end;
end;
procedure TForm1.Button1Click(Sender: TObject);
var
  i,j :integer;
  vper:zap;
begin
  for i:=1 to 9 do
    with StringGrid1,MZap[i] do
      begin
        fio:=Cells[1,i];
        mat:=StrToInt(Cells[2,i]);
        fiz:=StrToInt(Cells[3,i]);
        soch:=StrToInt(Cells[4,i]);
        srbal:=(mat+fiz+soch)/3;
        Cells[5,i]:=FloatToStrF(srbal,ffFixed,5,2);
      end;
    for i:=2 to 9 do // sortlaşdyrmak
      for j:=9 downto i do
        if MZap[j-1].srbal<MZap[j].srbal then

```

```

public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation
  {$R *.DFM}
  // Forma döretmek
procedure TForm1.FormCreate(Sender: TObject);
begin
  Edit1.Text:='0.5'; // X-iň başlangyç bahasy
  Edit2.Text:='1.8'; // Y başlangyç bahasy
  Memo1.Clear; // Memo larassalamak
  Memo1.Lines.Add(' şahalanýan algoritm');
end;
procedure TForm1.Button1Click(Sender: TObject);
var
  x,y,z,fx : extended; // lokal üýtgeýän ululyklar
begin
  x:=StrToFloat(Edit1.Text);
  y:=StrToFloat(Edit2.Text);
  fx:=sin(x);
  case RadioGroup1.ItemIndex of
    0:fx:=sin(x);
    1:fx:=cos(x);
  end;
  // Aňlatmany hasaplamak
  if x<y then
    z:=fx
  else
    z:=y;
  if CheckBox1.Checked then
    Memo1.Lines.Add('X = '+Edit1.Text+
      ' Y = '+Edit2.Text);

```

//Netijäni Memo1 çykarmak:

Memo1.Lines.Add(' Z = '+FloatToStrF(z,ffFixed,8,3));

end;

end.

## 7. Radiofiziki meselelerde sanly differensirlemek üçin ulanylýan algoritmler.

Sazlamak işi proektde goýberilen ýalňyşlyklary tapmak, lokallaşdyrmak we aradan aýyrmak işlerden durýar. Ol programmistiň lomaý wagtyny alýar, hatda proekt düzmekden hem köp wagt sarp etmeli bolýar. Köplenç in ýönekeý proyekt hem öz düzüminde ýalňyş saklaýar.

Sintaksis ýalňyşlary düzetmek diýmek operatorlaryň ulanyşynyň formal düzgünleriniň pozulan ýerlerini düzetmek diýmekdir. Programmistiň programmirleji dili gowşak bilýänligini görkezýär.

Sintaksis ýalňyşlyklar kompilyator tarapyndan gözlenýär. Ol goýberilen ýalňyşlary belleýär we olara häsiýetnama berýär. Bellemeli zatlar:

- 1) kompilyator hemme ýalňyşlary aňsyryp bilmeýär;
- 2) Käbir ýalňyşlar dogry operatorlara hem ýalňyşlyk girizip biler, tersine kompilyator dogry operatorlara hem ýalňyş diýip biler;
- 3) modulyň bir ýerindäki ýalňyşlyk kompilyatora modulyň başga ýerlerinde hem ýalňyş bar diýen habary çykardyp biler;
- 4) käbir ýalňyşlyklar sebäpli kompilyatoryň işi togtadylmagy mümkin.

implementation

{ \$R\*.DFM }

type

zap=record // ýazgyny beýan etmek

fio : string[20];

mat,fiz,soch:integer;

srbal: extended;

end;

var

Mzap:array[1..9] of zap // ýazgy toplumyny beýan etmek

Procedure TForm1.FormCreat(Sender:TObject);

var

i:=integer;

begin

with StringGrid1 do

begin // StrinGrid öýjüğine maglumatlary girizmek

Cells[0,0]:='Netb';

Cells[1,0]:='famiýasy,ady';

Cells[2,0]:='matematika';

Cells[3,0]:='fizika';

Cells[4,0]:='Orta bal';

for i:= 1 to 9 do

Cells[0,i]:=IntToStr(i);

Cells[1,1]:='Birinji

P.P.';Cells[2,1]:='3';Cells[3,1]:='3';Cells[4,1]:='3';

Cells[1,2]:='Ikinji

W.W.';Cells[2,2]:='3';Cells[3,2]:='3';Cells[4,2]:='4';

Cells[1,3]:='Üçünji

T.T.';Cells[2,3]:='3';Cells[3,3]:='4';Cells[4,3]:='4';

Cells[1,4]:='Dördünji

C.C.';Cells[2,4]:='4';Cells[3,4]:='4';Cells[4,4]:='4';

Cells[1,5]:='Bäşinji

P.P.';Cells[2,5]:='3';Cells[3,5]:='4';Cells[4,5]:='5';

Cells[1,6]:='Altynjy

S.S.';Cells[2,6]:='5';Cells[3,6]:='4';Cells[4,6]:='3';



komponentasynyň fiksirlenen zolaklary ulanylýar. şonuň üçin obýektleriň inspektorynda FixedCols we FixedRows häsiýetnamalaryň bahasyny 1-e deň diýip almaly. Berlen meseläniň şertine görä ColCount=6 we ColCount=10 bahalary goýmaly.

StringGrid komponentinde ähli okuwçylaryň sanawyny görmek üçin dik çyzykly lineýka aýlawy ulanmak amatlydyr. Onuň üçin ScrollBars häsiýetnamasyny ssVertikal ýagdaýda goýmak gerek. Opsiýalar sanawynda + Options häsiýetnamany açyp goEditing bahany True belgide goýmaly. Munuň özi StringGrid komponentinde maglumatlary "syçanjygyň" we klawiaturanyň kömegi bilen redaktirlämäge mümkinçilik berer.

Hadysalary işläp geçmekde FormCreate we ButtonClick proseduralary döretmek.

Syçanjygyň düwmesine iki gezek basmak arkaly, ýagny Forma we Button1 düwmelerine basmak arkaly degişli hadysalary işläp geçmek proseduralaryny döretmek bolýar. UnZap modulynyň tekstlerini peýdalanyp, üns bilen proseduranyň operatorlaryny saýlap almaly.

Unzap modulynyň teksti

```
Unit UnZap
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, Buttons, Grids;
type
TForm1=class(TForm)
  StrinGrid1:TStringGrid1;
  Button1:TButton;
  procedure FormCreate(Sender:TObject);
  procedure Button1Click(Sender:TObject);
private
public
end;
var
  Form1:TForm1;
```

Modulda hemme göyberilen ýalňyşlar baradaky habar ekranyň aşagyndaky ýörite penjirede çykýar. Has giň maglumatlary almak üçin F1 düwmäni basmaly.

Ýalňyşy lokallaşdyrmak üçin müňkür ýere kursory goýup F4 düwmäni basmaly (kursora çenli ýerine ýetirmeli diýdigi). F8 düwmäni basyp maksatnamanyň ädimme-ädim ýerine ýetiriler ýaly edip bolýar.

$$\text{Mesele: } Y(x) = \left(1 - \frac{x^2}{2}\right) \cos x - \frac{x}{2} \sin x$$

funksiýanyň bahalarynyň tablisasyny we

$$S(x) = \sum_{n=0}^n (-1)^n \frac{2n^2 + 1}{(2n)!} x^{2n} \text{ jem görnüşinde hatara}$$

dagydýan Windows goşundy döretmeli.

### *UnCiklAlg modulyň teksti*

**Unit** UnCiklAlg;

**interface**

**uses**

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

**type**

```
TForm1 = class(TForm)
  Memo1: TMemo;
  Button1: TButton;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Edit1: TEdit;
  Edit2: TEdit;
  SpinEdit1: TSpinEdit;
```

```

    CheckBox1: TCheckBox;
    CheckBox2: TCheckBox;
    CheckBox3: TCheckBox;
    procedure FormCreate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
implementation
{$R *.DFM}
procedure TForm1.FormCreate(Sender: TObject);
begin
    SpinEdit1.text:='3';
    Edit1.text:='0.1';
    Edit2.text:='2.0';
    Memo1.Clear;
    Memo1.Lines.Add('Siklikli algoritm');
end;
procedure TForm1.Button1Click(Sender: TObject);
var xn,xk,x,h,c,s,y,al,del:extended;
    n,k:integer;
begin
    n:=StrToInt(SpinEdit1.Text);
    xn:=StrToFloat(Edit1.Text);
    xk:=StrToFloat(Edit2.Text);
if CheckBox1.Checked then
        Memo1.Lines.Add('Başlangıç bahalar: n='+IntToStr(n)+
            ' Xn='+FloatToStrF(xn,ffFixed,6,1)+
            ' Xk='+FloatToStrF(xk,ffFixed,6,1));
    h:=(xk-xn)*0.1;    // adım
    x:=xn;
repeat                // x boyunca sikl

```

```

StringGrid1.ColCount:=m;
StringGrid2.RowCount:=m;
p:=StrToInt(SpinEdit3.Text);
q:=StrToInt(SpinEdit4.Text);
for i:=1 to n do
    for j:=1 to m do
        A[i,j]:=StrToFloat(StringGrid1.Cells[j-1,i-1]);
for j:=1 to m do
    begin
        X[j]:=A[p,j];
        StringGrid2.Cells[0,j-1]:=FloatToStrF(X[j],ffFixed,3,1);
    end;
for i:=1 to n do
    begin
        Y[i]:=A[i,q];
        StringGrid3.Cells[0,i-1]:=FloatToStrF(Y[i],ffFixed,3,1);
    end;
end;
end.

```

## 9. Monte-Carlo hasaplaýyş algoritmleri. Monte-Carlo hasaplaýyş usulynyň aýratynlyklary.

Ýazgylar bilen işlenende maglumatlary ekranda girizmek we çykarmak üçin StringGrid komponentini ulanmak has amatlydyr.

Mesele: dokuz okuwçydan ybarat synpyň ýetişik sanawnamasyny işläp taýýarlamagyň Windows-goşundysyny döretmeli. Her bir ýazgy özünde okuwçynyň familiýasyny, adyny, fizika, matematika, himiýa dersleri boýunça bahalary saklamaly. Okuwçylaryň sanawyny orta bahalaryň kemelýän tertibinde çykarmaly.

Döredilýän goşundylaryň biriniň interfeýs paneli kesgitlenen şekilde bolmaly. Berlen meselede setirlere we sütünlere degişli ýazgylary ýazmak üçin StringGrid

```

var
  A:array[1..6,1..8] of extended;
  X:array[1..8] of extended;
  Y:array[1..6] of extended;
  n,m,p,q:integer;
procedure TForm1.FormCreate(Sender: TObject);
begin
  SpinEdit1.Text:='4';
  SpinEdit2.Text:='6';
  SpinEdit3.Text:='2';
  SpinEdit4.Text:='3';
  StringGrid1.RowCount:=4;
  StringGrid1.ColCount:=6;
  StringGrid2.RowCount:=6;
  StringGrid3.RowCount:=4;
end;
procedure TForm1.SpinEdit1Change(Sender: TObject);
begin
  n:=StrToInt(SpinEdit1.Text);
  StringGrid1.RowCount:=n;
  StringGrid3.RowCount:=n;
end;
procedure TForm1.SpinEdit2Change(Sender: TObject);
begin
  m:=StrToInt(SpinEdit2.Text);
  StringGrid1.ColCount:=m;
  StringGrid2.RowCount:=m;
end;
procedure TForm1.Button1Click(Sender: TObject);
var
  i,j:integer;
begin
  n:=StrToInt(SpinEdit1.Text);
  StringGrid1.RowCount:=n;
  StringGrid3.RowCount:=n;
  m:=StrToInt(SpinEdit2.Text);

```

```

  c:=-x*x*0.5;
  S:=1;
for k:=1 to n do
begin
  s:=s+c*(2*k*k+1);
  c:=-c*x*x/((2*k+1)*(2*k+2));
end;
y:=(1-x*x*0.5)*cos(x)-0.5*x*sin(x);
if CheckBox2.Checked then
if CheckBox3.Checked then
begin
  al:=s-y; // абсолютный
  del:=abs((s-y)/y)*100; // относительный
  Memo1.Lines.Add('x='+FloatToStrF(x,ffFixed,6,2)+
    ' S='+ FloatToStrF(s,ffFixed,6,3)+
    ' Y='+ FloatToStrF(y,ffFixed,6,3)+
    ' A='+ FloatToStrF(al,ffFixed,6,3)+
    ' D='+ FloatToStrF(del,ffFixed,6,0)+'%');
end
else
begin
  al:=s-y;
  Memo1.Lines.Add('x='+FloatToStrF(x,ffFixed,6,2)+
    ' S='+ FloatToStrF(s,ffFixed,6,3)+
    ' Y='+ FloatToStrF(y,ffFixed,6,3)+
    ' A='+ FloatToStrF(al,ffFixed,6,3));
end
else
if CheckBox3.Checked then
begin
  del:=abs((s-y)/y)*100;
  Memo1.Lines.Add('x='+FloatToStrF(x,ffFixed,6,2)+
    ' S='+ FloatToStrF(s,ffFixed,6,3)+
    ' Y='+ FloatToStrF(y,ffFixed,6,3)+
    ' D='+ FloatToStrF(del,ffFixed,6,0)+'%');
end

```

```

    else
      Memo1.Lines.Add('x='+FloatToStrF(x,ffFixed,6,2)+
        ' S='+ FloatToStrF(s,ffFixed,6,3)+
        ' Y='+ FloatToStrF(y,ffFixed,6,3));
      x:=x+h;
    until x>xk;
  end;
end.

```

## 8. Radiofiziki meselelerde sanly integrirlemek üçin ulanylýan algoritmler.

**Mesele:** Windows-goşundy  $x=\{x_1, x_2, \dots, x_m\}$  wektory we  $y=\{y_1, y_2, \dots, y_n\}$  wektory hasaplamaly. Hasaplamada  $A=\{a_{ij}\} (x_j=a_{pj}, j=1, 2, \dots, m)$  we  $A=\{a_{ij}\} (y_i=a_{iq}, i=1, 2, \dots, n)$  ( $n \leq 6, m \leq 8$ ) matrisalary peýdalanmaly.

Massiwler bilen işlenende StringGrid komponentany ulanmak amatly bolýar. Bu komponenta maglumatlary tablisa görnüşinde şekillendirmek üçin ulanylýar. Setirleriň we sütünleriň mukdary FixedRows ۽ FixedCols häsiýetlerde kesgitlenýär.

Maglumatlary almak üçin Cells[ACol, ARow: integer]: string häsiýetler ulanylýar. Bu ýerde ACol-sütün nomeri,



ARow – setir nomeri. StringGrid piktogramma Additional sahypada ýerleşýär.

SpinEdit1Change we SpinEdit2Change hadysalar redaktoryň meýdanyndaky maglumat üýtgände ýüze çykyar.

### UnMas modulyň teksti

```

Unit UnMas;
interface

```

### uses

```

  Windows, Messages, SysUtils, Classes, Graphics, Controls,
  Forms, Dialogs,
  StdCtrls, Spin, Grids;

```

### type

```

TForm1 = class(TForm)
  Label1: TLabel;
  SpinEdit1: TSpinEdit;
  SpinEdit2: TSpinEdit;
  Label8: TLabel;
  StringGrid1: TStringGrid;
  StringGrid2: TStringGrid;
  StringGrid3: TStringGrid;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  SpinEdit3: TSpinEdit;
  SpinEdit4: TSpinEdit;
  Label6: TLabel;
  Label7: TLabel;
  Button1: TButton;

```

```

  procedure FormCreate(Sender: TObject);
  procedure SpinEdit1Change(Sender: TObject);
  procedure SpinEdit2Change(Sender: TObject);
  procedure Button1Click(Sender: TObject);

```

```

private
  { Private declarations }
public
  { Public declarations }
end;

```

### var

```

  Form1: TForm1;

```

### implementation

```

{$R *.DFM}

```