

B.Atajanow, O.Nurgeldiyew

PROGRAMMIRLEMENIŇ ESASLARY

Ýokary okuw mekdepleri üçin okuw kitaby

*Türkmenistanyň Bilim ministrligi
tarapyndan hödürilenildi*

Aşgabat
Türkmen döwlet neşirýat gullugy
2017

Atajanow B., Nurgeldiyew O.

A 85 Programmirlmegiň esaslary. Ýokary okuw mekdepleri üçin okuw kitaby. – A.: Türkmen döwlet neşirýat gullugy, 2017.

Okuw kitabynda Paskal diliniň esasynda programma düzmegiň esaslary beýan edilýär. I bap algoritmi düşünjesine, algoritmiň esasy häsiýetlerine we beýan ediliş usullaryna bagyşlanýar. II bapda Paskal dilinde programmanyň gurluşy we esasy operatorlar beýan edilýär. III bap simwol we setir ululyklar bilen işlemeklige, IV bap faýllar bilen işlemeklige, V bap modullary döretmeklige bagyşlanýar. VI bapda öňki baplarda öwrenilen materiallaryň esasynda dürli amaly programmalary düzmekligiň meselelerine seredilip geçilýär.

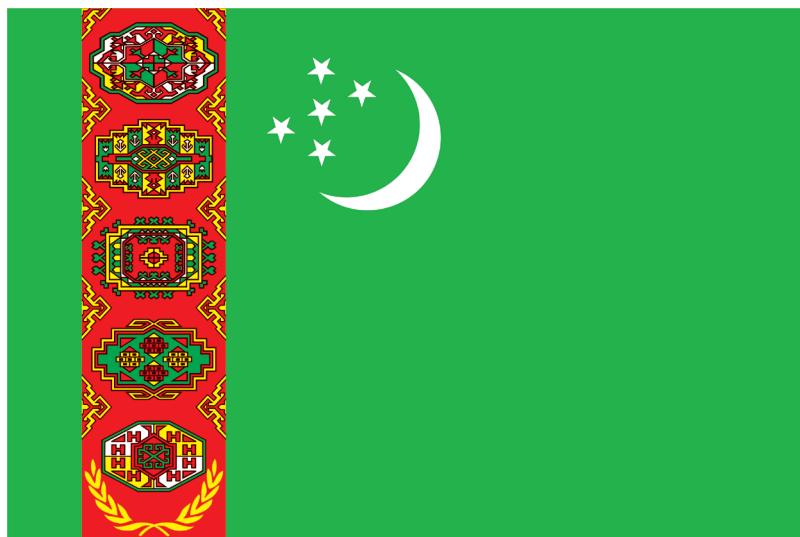
Okuw kitaby programma düzmek bilen gyzyklanýan giň okyjylar köpçüligi üçin hem peýdaly bolar diýip hasap edýäris.



**TÜRKMENISTANYŇ PREZIDENTI
GURBANGULY BERDIMUHAMEDOW**



TÜRKMENISTANYŇ DÖWLET TUGRASY



TÜRKMENISTANYŇ DÖWLET BAÝDAGY

TÜRKMENISTANYŇ DÖWLET SENASY

Janym gurban saňa, erkana ýurdum,
Mert pederleň ruhy bardyr köňülde.
Bitarap, garaşsyz topragyň nurdur,
Baýdagyň belentdir dünýäň öňünde.

Gaýtalama:

Halkyň guran Baky beýik binasy,
Berkarar döwletim, jigerim - janym.
Başlaryň täji sen, diller senasy,
Dünýä dursun, sen dur, Türkmenistanym!

Gardaşdyr tireler, amandyr iller,
Owal-ahyr birdir biziň ganymyz.
Harasatlar almaz, syndyrmaz siller,
Nesiller döş gerip gorar şanymyz.

Gaýtalama:

Halkyň guran Baky beýik binasy,
Berkarar döwletim, jigerim - janym.
Başlaryň täji sen, diller senasy,
Dünýä dursun, sen dur, Türkmenistanym!

GIRIŞ

Ýokary okuw mekdepleriň maglumat tehnologiýalarynyň programma üpjünçiligi hünärinde okaýan talyplar üçin «Programmirmegiň esaslary» dersi aýratyn orun tutýar. Bu dersde programma düzmeğiň esaslary öwredilýär. Programma düzmek birnäçe ädimlerden durýar. Adatça ol ädimler aşakdakylardan ybaratdyr:

1. Meseläni goýmak;
2. Meseläni çözmek üçin gerek bolan başlangyç maglumatlary bilmek;
3. Meseläni çözmegiň algoritmini işläp taýýarlamak;
4. Programmany taýýarlamak;
5. Programmany testirmek we ýüze çykan ýalňyşlary düzetmek;
6. Programmany işe goýbermek we goýlan meseläniň çözüwini almak.

Köplenç halatlarda ýokardaky sanalan 3-nji hem-de 4-nji ädimler birikdirilýär we meseläni çözmegiň algoritmi gös-göni haýsy hem bolsa algoritmik dilleriň birinde ýazylýar.

«Programmirmegiň esaslary» dersi maglumat tehnologiýalarynyň programma üpjünçiligi hünärinde 1-nji we 2-nji ýarym ýyllykda geçilýär. Dersde dürli meseleleri çözmekde ulanylýan algoritmler, Paskal algoritmik diliniň operatorlary hem-de bu operatorlara degişli mysallar özleşdirilýär. Hususan-da, talyplara programmirmekde wajyp orun tutýan şertli operator, geçiş operator, gaýtalamak operatorlary, kömekçi programmalar, massiwler, köplükler, ýazgylar öwredilýär. Şeýle hem kitapda simwollar we setir ululyklar bilen işlemekde ulanylýan esasy usullar, maglumatlary gaýtadan işlemeklige degişli durmuşda duş gelýän käbir meseleler beýan edilýär. Okuw kitaby programma düzmek bilen gyzyklanýan giň okyjylar köplügi üçin hem peýdaly bolar diýip hasap edýäris.



Algoritm düşünjesi. Algoritmeleriň esasy häsiýetleri

Algoritm düşünjesi häzirki döwürde ylymda we durmuşda giňden ýaýran düşünjeleriň biridir. Algoritm sözünüň gelip çykyşy IX asyrdä ýaşap geçen Gündogaryň görnükli alymy Muhammet al-Horezminiň ady bilen baglanyşyklydyr. Al-Horezmi öz işleriniň birinde ilkinji gezek hakyky sanlaryň üstünde arifmetiki amallaryň ýerine ýetiriliş düzgünlerini beýan edipdir. Soň-soňlar algoritm sözünü diňe bir matematiki manysynda däl, eýsem, has giň manylarda ulanyp başlapdyrlar. Şunlukda, gutarnykly bir netijä getirýän hereketleriň zygiderlilikini algoritm, her bir hereketi bolsa algoritmiň ädimi diýip atlandyrypdyrlar.

Elbetde, algoritm düşünjesine bellibir kesgitleme bermek kyn, ýöne algoritme bellenilen maksada ýetmek ýa-da goýlan meseläni çözmek üçin ýerine ýetirilmeli hereketleriň (işleriň) zygiderlilikiniň takyk we düşnükli ýazgysy hökmünde garamak mümkin. Algoritm düzülende goýlan meseläni çözmek üçin ýerine ýetirilmegi zerur bolan işleriň (ädimleriň) zygiderli ýazgysy düzülýär. Her bir indiki ädim üçin onuň öň ýanyndaky ädimleriň netijeleri peýdalanylýar.

Taýýar algoritmi ýerine ýetirmegi islendik ýerine ýetirijä (adama, kompýutere) tabşyryp bolar. Şunlukda, ýerine ýetirijiniň çözülyän mesele bilen tanyş bolmagy hökman däldir. Ýerine ýetiriji algoritmdäki görkezilen işleri doly we dürs berjaý edip, meseläniň çözüwini tapyp biler. Eger algoritmi ýerine ýetirmek kompýutere tabşyrylmaly bolsa, onda ol kompýutere düşnükli dilde ýazylyar. Kompýuterde çözülmeli islendik mesele üçin, ilki bilen, onuň çözülişiniň algoritmini düzmek gerek bolýar. Islendik ädimleriň sanawy algoritm bolmaýar. Onuň algoritm bolmagy üçin birnäçe häsiýetleriň ýerine ýetmegi hökmandyr. Bu häsiýetler barada aýratyn durup geçeliň.

Algoritmiň kesgitlilik häsiýeti. Algoritmde görkezilen her bir iş anyk, ýeke-täk manyly bolmalydyr. Bu häsiýete baglylykda algorit-

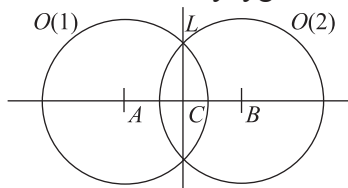


miň ýerine ýetirilişi mehaniki häsiýetdedir. Algoritm boýunça mesele çözmek pikirlenmäni talap etmeyär, ol diňe ädimleri tertip boýunça ýerine ýetirmeklige syrykdyrylýar. Çözülýän mesele barada hiç hili goşmaça görkezmeler talap etmezden, islendik ýerine ýetiriji ony ýerine ýetirip biler.

1-nji mysal. Sirkulyň we çyzgyjyň kömegi bilen AB kesimi ýarpa bölmeli.

Çözülişi. Mysalyň çözülişiniň algoritmini ädimleriň yzygiderligi görnüşinde ýazalyň:

1-nji ädim. Çyzgyjyň kömegi bilen AB kesimi gurmaly.



2-nji ädim. Sirkulyň kömegi bilen merkezi A nokatda bolan $R > AB/2$ radiusly töweregi gurmaly. Ony $O(1)$ bilen bellemeli.

3-nji ädim. Sirkulyň kömegi bilen merkezi B nokatda bolan R radiusly töweregi gurmaly. Ony $O(2)$ bilen bellemeli.

4-nji ädim. Çyzgyjyň kömegi bilen $O(1)$ we $O(2)$ töwerekleriň kesişýän nokatlaryny birleşdirýän göni çyzyk geçirmeli. Ony L bilen bellemeli.

5-nji ädim. AB kesim bilen L göni çyzygyň kesişme nokadyny C bilen bellemeli. C nokat - AB kesimi ýarpa bölýän nokatdyr.

Algoritmiň umumylyk häsiýeti. Her bir algoritm ýeke-täk meseläni çözmeklige niýetlenilen bolmaly däldir. Ol şol bir görnüşli meseleleriň bütin bir toplumyny dürli başlangyç bahalarda çözmek üçin ýaramly bolmalydyr.

2-nji mysal. $ax^2+bx+c=0$ ($a \neq 0$) deňlemäni çözmeli.

Çözülişi.

1-nji ädim. a, b, c ululyklaryň bahasyny girizmeli.

2-nji ädim. Eger $a=0$ bolsa, onda 1-nji ädime dolanyp barmaly.



3-nji ädim. $D=b^2 - 4ac$ formula boýunça D diskriminanty hasaplamaly.

4-nji ädim. Eger $D \geq 0$ bolsa, onda

$$x_1 = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \quad x_2 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

formular boýunça kökleri tapmaly, bolmasa «Kök ýok» diýen habary çykarmaly.

Algoritmiň üznüklilik häsiýeti. Algoritm meseläniň çözüwine getirýän ýönekeý (ýa-da oň kesgitlenen) ädimleriň zygiderli ýerine ýetirilişini beýan etmelidir. Başgaça aýdylanda, algoritmda beýan edilýän iş aýry-aýry ädimleriň zygiderligine bölünen bolmalydyr. Şunlukda, her bir ädimiň ýerine ýetirilmegi üçin käbir tükenikli wagt zerurdyr. Meseleleriň aglaba köpüsi üçin algoritm düzmek dördijilikli işdir, ýöne taýýar algoritm boýunça mesele çözmek hiç hili subut etmeleri, anyklamalary talap etmeýär, şonuň üçin hem ol dördijilikli iş hasaplanylmaýar. Ol diňe ädimleri ýerine ýetirmäge syrykdyrylýar.

Algoritmiň düşnüklik häsiýeti. Algoritm haýsy hem bolsa bir ýerine ýetirijä niýetlenilip düzülýär. Ýerine ýetirijiniň haýsy görkezmelere düşünjekdigini ýa-da düşünmejekdigini bilmek zerurdyr, sebäbi, her bir ýerine ýetirijiniň özüne düşnükli, ýerine ýetirip biljek görkezmeleri bolýar.

Algoritmiň netijelilik häsiýeti. Algoritmiň tükenikli ädimleri doly geçilenden soňra meseläniň haýsydyr bir kesgitli çözüwi alynmalydyr. Meseläniň çözüwiniň ýokdugyny anyklamaklyga hem çözüw hökmünde garalýar.

Algoritmiň beýan ediliş usullary

1. Algoritmeleriň söz bilen beýan edilişi. Algoritmeleriň bu hili beýan edilişine durmuşda köplenç duş gelýän düzgünler toplumu mysal bolup biler: ýoly kesip geçmegiň, telefonda gürleşmegiň, nahar taýýarlamagyň we ş.m. düzgünleri.

Goý, $y = \frac{5x}{7+x^2}$ formula boýunça hasaplamany amallaryň



zygyderligi görnüşinde ýazmak gerek bolsun. Algoritmiň söz bilen beýan edilişi şeýle görnüşde bolar:

1-nji ädim. x -iň berlen bahasyny okamaly;

2-nji ädim. x -i 5-e köpeltmeli, netijäni a_1 bilen bellemeli;

3-nji ädim. x -i kwadrata götermeli, netijäni a_2 bilen bellemeli;

4-nji ädim. 7 bilen a_2 -ni goşmaly, netijäni a_3 bilen bellemeli;

5-nji ädim. a_1 -i a_3 -e bölmeli, netijäni y -iň bahasy diýip kabul etmeli.

2. Algoritmeleriň beýan edilişiniň blok-shema usuly. Bu usulda algoritm uýy peýkamly kesimler bilen birikdirilen dürli tekiz geometrik şekillerden ybarat bolan çyzgy arkaly şekillendirilýär. Bu şekiller (bloklar) algoritmiň ol ýa-da beýleki ädimlerine degişli edilip, uýy peýkamly kesimler bolsa olaryň ýerine ýetirilýän zygiderligini kesgitläýärler. Blokly we olaryň näme üçin niýetlenendigini görkezeliň:

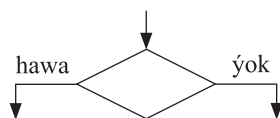
Proses. Bu bloğa arifmetiki blok hem diýilýär.



Onuň bir girelgesi we bir çykalgasy bardyr. Arifmetiki bloğuň mazmuny gönüburçluguň içinde ýazylýar.

Şertleriň barlagy. Bu bloğa logiki blok hem diýilýär. Logiki blok

algoritmelerde haýsy hem bolsa bir şerte görä, şahlanma bar bolan wagtynda peýdalanylýar. Onuň bir girelgesi we iki çykalgasy bardyr. Rombuň içinde şertiň mazmuny ýazylýar. Eger şert ýerine ýetýän (çyn) bolsa, onda «hawa» sözi bilen, ýerine ýetmeýän ýagdaýynda (ýalan) bolsa, «ýok» sözi bilen belgilenilen ugur boýunça hereket edilýär.



Maglumatlary girizmek.



Maglumatlary çap ediji gurluşa çykarmak.

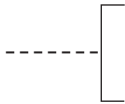


Öň döredilen we aýratyn beýan edilen algoritmelerden peýdalanmak.



Algoritmiň başlangyjy, soňy.





Düşündürişler.



Gaytalanýan proses.



Birleşdiriji.

3. Algoritmleriň ýörite algoritmik dilde beýan edilişi. Algoritmleri we olaryň ýerine ýetiriliş kadalaryny anyk we birmeňzeş beýan etmegi üpjün etmek maksady bilen, bir tarapdan adaty dile golaý, başga bir tarapdan bolsa özünde sanlary, ululyklary, amallaryň belgilerini, funksiýalary, hyzmat ediji sözleri we başgalary saklaýan ýörite algoritmik dilden hem peýdalanýarlar. Häzirki wagtda köp sanly algoritmik diller bardyr. Olardan Paskal, Basic, Java we beýlekileri görkezmek mümkindir. Algoritmleň haýsy hem bolsa bir algoritmik dilde beýan edilişine programma diýilýär.

Algoritmleriň görnüşleri

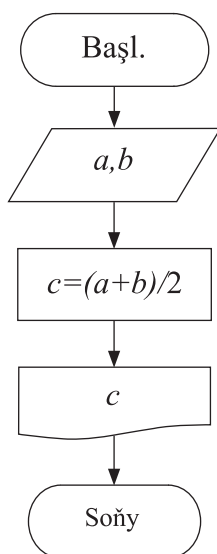
1. Çyzykly algoritmle. Bu görnüşli algoritmle gurluşlary boýunça has ýönekeýdir. Olarda hemme ädimle başdan başlanyp, ahyryna çenli biri-biriniň zyzndan zyzgiderli ýerine ýetirilýärler. Çyzykly algoritmleriň blok-shemasynda logiki blok bolmaýar. Şonuň üçin olaryň blok-shemasy biri-biri bilen baglanyşykly bloklaryň çyzykly zyzgiderlilikidir.

Çyzykly algoritmlele mysal hökmünde berlen a we b hakyky sanlaryň orta arifmetiki bahasyny hasaplamagyň blok-shemasyny getireliň.

Bu blok-shemanyň ikinji, parallelogram görnüşli blogunda a we b sanlaryň bahalary girizilýär. Üçünji, arifmetik blokda bu sanlaryň ýarym jemi hasaplanýar we ol c ululyga berilýär. Dördünji blokda bolsa hasaplanan jem çap edilýär.

Blok-shemanyň birinji we iň soňky bloklary degişlilikde onuň başlangyjyny we ahyryny aňladýarlar.



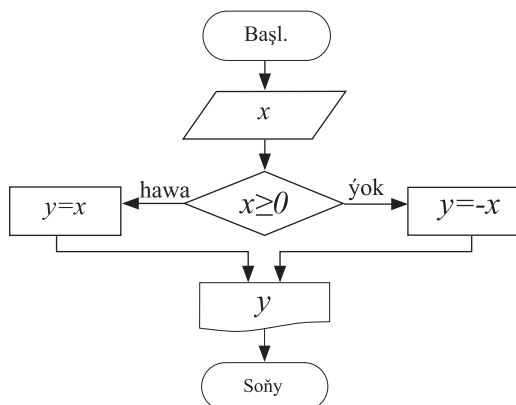


2. Şahalanýan algoritmler. Meseleleriň aglaba köpüsiniň çözülişini çyzykly algoritmler arkaly aňlatmak asla mümkin däl. Şoňa görä-de, olaryň çözüwlerini tapmak üçin dürli ýollardan peýdalanylýar. Bu ýollaryň haýsysynyň saýlanyp alynmalydygy bolsa meseläniň şertine baglydyr. Hasaplamagyň mümkin bolan her bir ýoluny şaha diýip atlandyryrs. Hasaplama işi käbir şerte baglylykda, ol ýa-da beýleki şaha boýunça gidip biler. Şu hili hasaplama işiniň algoritmi şahalanýan algoritmi diýlip atlandyrylýar. Şahalaryň sany iki ýa-da ondan köp bolup biler. Şoňa görä-de, şahalanýan algoritmleriň blok-shemasynda, iň bolmanda, bir logiki blok bolmalydyr. Bu algoritmlere mysal hökmünde sanyň modulyň kesgitlemegiň blok-shemasyny getireliň. Belli bolşy ýaly, sanyň moduly şeýle tapylýar:

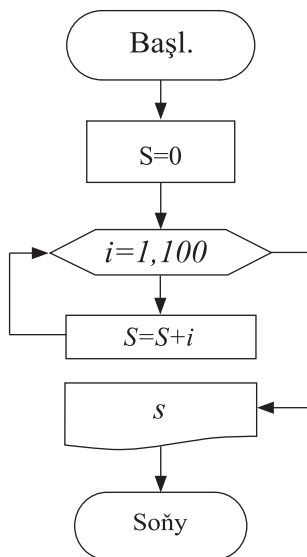
$$y = \begin{cases} x, & \text{eger } x \geq 0 \text{ bolsa;} \\ -x, & \text{eger } x < 0 \text{ bolsa.} \end{cases}$$

Blok-shemada iki sany şaha bar. Logiki blokda getirilen şerte görä, olaryň birinjisi («hawa» şahasy) x ululygyň girizilen bahasy 0-dan uly ýa-da 0-a deň bolan ýagdaýda ($y=x$ hasaplanýar), ikinjisi («ýok» şahasy) bolsa $x < 0$ bolan ýagdaýynda ($y=-x$ hasaplanýar) ýerine ýetirilýär.





3. Gaýtalanýan gurluşly algoritmler. Eger algoritmda käbir işler birnäçe gezek gaýtalanyp ýerine ýetirilýän bolsa, onda beýle algoritme gaýtalanýan gurluşly algoritmi diýilýär. Mysal üçin, 1-den 100-e çenli sanlaryň jemini tapmaklygyň algoritmi gaýtalanýan algoritmdir. Bu mysalyň blok-shemasy aşakdaky görnüşde bolýar:



Programmiremegin dillerine umumy syn

Biz algoritmi beýan etmegiň söz usuly we grafiki (blok-shema) usuly bilen has ýakýndan tanyş bolduk. Olaryň algoritm düzmegiň esaslaryny öwrenmekde uly ähmiýeti bardyr. Emma algoritmi beýan etmegiň söz we grafiki usullarynyň esasy bir uly kemçiligi hem bar: şol usullarda ýazylan algoritme kompýuter gös-göni düşünmeýär. Meseläni çözmegiň algoritmine kompýuteriň «düşünmegi» we ony ýerine ýetirmegi üçin algoritmi ýörite dilde ýazmalydyr. Şeýle dillere bolsa, programmiremegin dilleri ýa-da algoritmik diller diýip at berilýär. Programmiremegin dilinde ýazylan algoritme programma diýip at berilýändigini hem ýatladýarys.

Häzirki wagtda programmiremegin köp sanly dilleri bardyr. Ol diller, umuman, iki topara bölünýär: aşak derejeli diller we ýokary derejeli diller. Ilkinji elektron hasaplaýjy maşynlarda programmalar maşyn dilinde düzülipdir. Maşyn dili anyk kompýuteriň aýratynlyklaryna bagly bolýar. Anyk kompýuteriň aýratynlyklaryna bagly bolan programmirme dillerine aşak derejeli diller diýilýär. Aşak derejeli dillere mysal edip maşyn dilini, Assembler dilini getirmek bolar. Adatça, aşak derejeli dillerde programma düzmeklik çuň bilimi we köp zähmeti talap edýär. Anyk kompýuteriň aýratynlyklaryna bagly bolmadyk we ulanyjylaryň giň köplügi üçin niýetlenen algoritmik dillere ýokary derejeli (maşyn buýruklarynyň derejesine görä) diller diýilýär. Bu görnüşli dillere mysal edip, Basic, Paskal, Fortran, C, Java dillerini getirmek bolar. Bu dilleriň her biriniň öz elipbiýi, operatorlary, programma ýazmagyň düzgünleri bardyr.

Paskal dili 1971-nji ýylda şweýsar alymy Niklaus Wirt tarapyndan döredildi. Bu dil özüniň ýönekeýligi we öwrenmek üçin aňsatlygy bilen tapawutlanýar. Ol belli fransuz matematigi we fizigi Blez Paskalyň (1623 – 1662) hatyrasyna Paskal diýlip atlandyryldy. Paskal dilinde ýazylan programmany kompýutere girizmäge, kompilyasiýa



etmäge we ýerine ýetirmäge mümkinçilik berýän dürli programmalar döredilendir. Olardan Turbo-Pascal, Dev Pascal, Pascal ABC we başga programmalary mysal getirmek mümkindir. Delphin Object Pascal dili Turbo Pascala esaslanýar we onuň ideýalaryny ösdürýär. Bu bolsa Paskal diliniň çäksiz mümkinçilikleriniň bardygyny görkezýär.

Paskal diliniň esasy simwollary. Kömekçi sözler

Paskal diliniň elipbiýi aşakdaky simwollardan, kömekçi sözlerden we belgilerden durýar:

1) Latyn elipbiýiniň uly we kiçi harplary:

A,B,C,..., X,Y,Z, a,b,c,... x,y,z;

2) Onluk hasaplaýyş sistemasynyň sanlary: 0,1,2,... , 9;

3) Arifmetiki amallaryň belgileri:

+ (goşmak), - (aýyrmak), * (köpeltmek), / (bölmek), Div (bitin sanlary bölmek), mod (bitin sanlar bölünende galýan galyndyny kesgitlemek). Div we mod amallary bitin san köplüginde ýerine ýetirmek bilen, degişlilikde paýy we galyndyny kesgitleýär.

Meselem:

$20 \text{ div } 3 = 6;$ $16 \text{ div } 3 = 5;$ $25 \text{ div } 4 = 6;$

$13 \text{ mod } 5 = 3;$ $24 \text{ mod } 5 = 4;$ $12 \text{ mod } 3 = 0;$

4) Gatnaşyk belgileri:

> (uly), >= (uly ýa-da deň), < (kiçi), <= (kiçi ýa-da deň), = (deň),
◇ (deň däl);

5) Logiki amallaryň belgileri:

not (inkär etmek), or (ýa-da, dizýunksiýa - logiki goşmak), and (we, konýunksiýa - logiki köpeltmek), xor;

6) Baha bermek operatorynyň belgisi:

7) Biri-biriniň yzyndan gelýän iki gurluşyň arasyny aýyrmak üçin ulanylýan belgiler:



• (nokat), , (otur), : (goşa nokat), ; (nokatly otur).

8) Beýleki ýörite belgiler: ´ (apostrof), () (açylýan we ýapylýan ýaý), [] (kwadrat ýaý), (boş öýjük) we ş.m.

Hemişelikler

Ulanyjy tarapyndan ýazylan programma ýerine ýetirilýän wagtynda öz bahasyny üýtgetmeýän ululyklara hemişelikler diýilýär. Paskal algoritmik dilinde hemişelikleri aşakdaky toparlara bölmek mümkin:

san hemişelikleri;

liter hemişelikleri;

setir hemişelikleri;

logiki hemişelikler.

San hemişelikleri iki görnüşde bolup bilýär:

bitin san hemişelikleri;

hakyky san hemişelikleri.

Bitin san hemişelikleriniň tipi INTEGER, BYTE, SHORTINT, WORD, LONGINT, hakyky san hemişelikleri bolsa REAL görnüşinde bellenilýär. Meselem, Paskal dilinde bitin san hemişelikleri 23, 2001, -97, -465 we ş.m. ýaly ýazmak bolýar. Hakyky sanlar kompýuteriň ýadynda ýakynlaşan, ýagny käbir takyklyk bilen ýazylýar. Meselem: 21.0, 21.000001, 20.999999 sanlaryň ählisi hakyky tipde berlen 21-i aňladýar. Hakyky sanlary iki görnüşde ýazyp bolýar:

1) fiksirlenen nokat arkaly;

2) ýüzýän nokat arkaly.

Meselem, 0.161, - 19.74, - 4.0 we 141.69 sanlar fiksirlenen nokat arkaly ýazylandyr. Programmada ulanylýan absolyut ululygy boýunça has kiçi we has uly sanlar ýüzýän nokat arkaly aňladylýar. Meselem, 0.00075 hakyky sany ýüzýän nokat arkaly 75E-05, 7.5E-04, 0.75E03 we ş.m. görnüşinde ýazmak mümkindir. Hakyky san he-



mişelikleri absolýut ululygy boýunça (10^{-38} , 10^{38}) aralykda çäklendirilendir. Diliň elipbiýine girýän diňe bir sany simwoldan ybarat bolan ululyga liter hemişeligi diýilýär. Liter hemişelikleri « ' » apostrof belgisiniň arasynda ýazylýar. Meselem 'L', '*', 'C', '8' we ş.m. liter hemişelikleridir. Liter hemişelikleriň tipi CHAR görnüşde ýazylýar. Paskal dilinde iki sany logiki hemişelik ulanylýar:

1) TRUE (çyn);

2) FALSE (ýalan).

Logiki hemişelikleriň tipi BOOLEAN görnüşinde bellenilýär. Setir hemişeligi diýip diliň elipbiýine girýän harplaryň, sanlaryň we beýleki simwollaryň toplumyna aýdylýar. Setir hemişeligi hem apostrof belgisiniň içinde ýazylýar. Meselem: 'Y=', 'NETIJE', 'JEM' we ş.m. setir hemişeliklerine mysaldyr. Paskal dilinde setir hemişeliginiň tipi STRING görnüşinde ýazylýar. Setirdäki ähli simwollaryň sanyna setir hemişeliginiň uzynlygy diýilýär. Uzynlygy diňe 1-e deň bolan setir hemişeligine liter hemişeligi hökmünde seretmek mümkindir.

Üýtgeýän ululyklar

Kompýuterde programma ýerine ýetirilýän mahalynda öz bahasyny üýtgedip bilýän ululyklara üýtgeýän ululyklar diýilýär. Üýtgeýän ululyklary atlandyrmak üçin ulanylýan belgilere bolsa identifikatorlar diýilýär. Identifikator üýtgeýän ululyga berilýän bahanyň kompýuteriň ýadynyň haýsy öýjüginde ýerleşýändigini görkezmek üçin hyzmat edýär. Identifikatoryň birinji simwoly hökman latyn harpy bolmalydyr. Paskalda identifikatoryň uzynlygy 63 simwol bilen çäklendirilýär. Meselem: a , $x1$, delta we ş.m. identifikatorlara mysal bolup biler. 1p ýazgy identifikator bolup bilmeyär, sebäbi, onuň birinji simwoly sandyr.

Üýtgeýän ululyklaryň atlaryny Paskal dilinde ulanylýan ýörite sözleriň hiç birine hem gabat gelmez ýaly edip saýlap almaly. Paskal dilinde programmada ulanylýan hemişelikleriň, funksiýalaryň we proseduralaryň atlaryna hem identifikatorlar diýlip düşünilýär. Identifikatorlary standart we standart däl identifikatorlara bölmek bolýar. Standart identifikatorlar diýip, diliň özünde öňden kabul edilen identifikatorlara aýdylýar.



Meselem:

Standart identifikator	Manysy
FALSE, TRUE	standart logiki hemişelikler
MAXINT,PI	standart san hemişelikleri
ABS, SQR, COS, SIN, LN, EXP, SQRT, ARCTAN, TRUNC, ROUND, PRED, SUCC, ORD, CHR, ODD, EOF, EOLN we ş.m.	standart funksiýalaryň atlary
GET, PUT, NEW, REWRITE, READ, PACK, UNPACK, READLN, RESET, WRITE, WRITELN, PAGE we ş.m.	standart proseduralaryň atlary

Standart däl identifikator diýlip, programmany düzýän ulanyjynyň girizýän identifikatoryna aýdylýar. Meselem:

```
VAR A100, OMEGA, K : INTEGER; Y5, X10, Z : REAL;  
C,T : BOOLEAN; D10 : CHAR;
```

Gurluşy boýunça üýtgeýän ululyklary iki topara bölmek bolýar:

Ýönekeý üýtgeýän ululyklar;

Indeksli üýtgeýän ululyklar.

Ýönekeý üýtgeýän ululyklara mysallar:

X2, Y5, Z1, Q3 we ş.m.

Olar INTEGER, REAL, BOOLEAN, CHAR,...tipleriň islendigine degişli bolup bilýärler.



Standart funksiýalar

Köp ulanylýan elementar funksiýalaryň bahalaryny hasaplamak üçin standart funksiýalardan peýdalanylýar. Hasaplamalarda köp duş gelyän standart funksiýalar: $\sin x$, $\cos x$, \exp , $\ln x$, \sqrt{x} , $|x|$, x^2 we ş.m.

Standart funksiýalaryň bahalaryny hasaplamak üçin niýetlenen programmalar kompýuteriň ýadynda saklanylýar. Olara ýüzlenmek üçin funksiýalaryň adyny ýazyp, ýaýyň içinde argumentiň bahasy görkezilýär. Standart funksiýalara programmanyň islendik ýerinde baha bermek operatorlarynyň sag tarapyndaky aňlatmadan ýüzlenip bolýar.

Meselem:

$$y := a^3 * \sin(x).$$

Paskal algoritmik dilinde ulanylýan esasy standart funksiýalar:

Funksiýanyň ady	Argumentiň tipi	Netijäniň tipi	Funksiýanyň manysy
ABS (X)	bitin/hakyky	Argument-däki ýaly	X -iň absolyút bahasy
SIN (X)	hakyky	hakyky	X radiandan alnan sinus
COS (X)	hakyky	hakyky	X radiandan alnan cosinus
ARCTAN (X)	hakyky	hakyky	X radiandan alnan arktangens
SQRT (\underline{X})	bitin/hakyky	hakyky	X -dan alnan kwadrat kök
SQR (X)	bitin/hakyky	argument-däki ýaly	X -iň kwadraty
EXP (X)	hakyky	hakyky	e-niň derejesindäki X
LN (X)	hakyky	hakyky	X -iň natural logarifmi
TRUNC (X)	hakyky	Longint	X -iň bitin bölegi



Funksiýanyň ady	Argumentiň tipi	Netijäniň tipi	Funksiýanyň manysy
FRAC (X)	hakyky	hakyky	X -iň drob bölegi
INT (X)	hakyky	hakyky	X -iň bitin bölegi
ROUND (X)	hakyky	Longint	X -i iň ýakyn bitin sana çenli tegeleklemek
RANDOM	-	hakyky	(0...1) aralykdan tötänleýin sanlar
RANDOM (X)	Word	Word	(0... X -1) aralykdan tötänleýin sanlar
ODD (X)	Bitin	Logiki	Eger X -iň bahasy täk bolsa netije TRUE, bolmasa FALSE

Trigonometrik funksiýalaryň bahalary radianlarda hasaplanylýar. Eger burç graduslarda berlen bolsa, onda ony aşakdaky formula boýunça radianlara öwürmeli:

$$\text{radian} = \text{gradus} * \text{PI} / 180$$

Ters trigonometrik funksiýalary hasaplamak üçin aşakdaky formulalardan peýdalanmak bolýar:

$$\arcsin x = \arctg(x / \sqrt{1-x^2});$$

$$\arccos x = \arctg(\sqrt{1-x^2} / x);$$

$$\text{arctg } x = \arctg(1/x).$$

Islendik esasy logarifmi hasaplamakda bir esasan beýleki esasa geçmek üçin

$$\log_a b = \ln b / \ln a$$

formuladan peýdalanýlar.

Standart funksiýalaryň bahasy hasaplanylanda onuň argumentini ýaýyň içinde ýazmaly we açylýan-ýapylýan ýaýlaryň sany deň bolmalydyr.



Meselem: $\cos^2 x^2$

funksiýanyň bahasy hasaplanylanda aňlatmany $\text{sqr}(\cos(\text{sqr}(x)))$ görnüşde ýazyp bolýar.

Tablisada beýan edilişi ýaly, TRUNC standart funksiýasy hakyky sanyň drob bölegini kesip taşlaýar.

Meselem:

TRUNC (-12.75)=-12 ýa-da TRUNC (46.69)=46.

ROUND standart funksiýasy argumenti iň ýakyn bitin sana çenli tegelekleyär. Meselem:

ROUND (-9.5)=-10.0 ýa-da ROUND (35.49)=35.0.

PRED we SUCC standart funksiýalary bitin, CHAR we BOOLEAN tipli argumentler üçin degişlilikde berlen elementiň öň ýanyndaky we yz ýanyndaky elementi kesgitleýär.

Meselem:

PRED (6)=5; SUCC (9)=10 (Argument INTEGER tipe degişli bolanda);

PRED ('Y')= 'X'; SUCC ('Y') = 'Z' (argument CHAR tipe degişli bolanda).

ORD we CHR standart funksiýalar bir-birine ters funksiýalar bolup, olara başgaça özgerdiji funksiýalar hem diýilýär. ORD (X) standart funksiýa X simwolyň koduny kesgitleýär. Meselem:

ORD ('0')=48;

ORD ('A')=65;

ORD (' ')=32;

ORD('B')=66; we ş.m.

CHR (I) standart funksiýa kody boýunça oňa degişli bolan simwoly kesgitleýär.

Meselem:

CHR (48)= '0'; CHR (65)= 'A' we ş.m.

ODD (X) standart funksiýa X bitin sanyň jübütligini ýa-da täkligini kesgitleýär:



Eger x san ták bolsa, onda $ODD(x)$ standart funksiýa «TRUE» baha eýe bolýar. Meselem, $ODD(11)=TRUE$; $ODD(19)=TRUE$.

Eger x san jübüt bolsa, onda $ODD(x)$ standart funksiýa «FALSE» baha eýe bolýar. Meselem, $ODD(8)=FALSE$; $ODD(36)=FALSE$.

$ODD(x)$ standart logiki funksiýalara degişlidir. $EOLN(x)$ -logiki funksiýa x -faýlda setiriň soňuny kesgitlemek üçin ulanylýar.

$EOF(x)$ - x -faýlyň soňuny kesgitlemek üçin ulanylýar.

Aňlatmalar

Paskal algoritmik dilinde aňlatmalar hemişeliklerden, ýönekeý we indeksli üýtgeýän ululyklardan, standart funksiýalardan, amallaryň belgilerinden we ýaýlardan ybarat bolup bilerler.

Aňlatmanyň bahasy bitin ýa-da hakyky tipe degişli bolsa, onda onuň ýaly aňlatmalara arifmetiki aňlatmalar diýilýär. Meselem:

$$(1*25 * \sin(x)+PI)/SQR(A+B)/x.$$

Arifmetiki aňlatmalarda amallar aşakdaky tertipde ýerine ýetirilýär:

- ilki ýaýyň içi ýerine ýetirilýär;
- eger ýaý özünde ýene-de birnäçe ýaýlary saklaýan bolsa, onda hasaplama iň içki ýaýyň içinden başlanýar;
- ýaýyň içinde ilki standart funksiýalaryň bahalary hasaplanylýar;
- soňra $*$, $/$, DIV , MOD amallar, şondan soň $+$, $-$ amallar ýerine ýetirilýär.

Paskal dilinde derejä görtermek amaly $LN(x)$ we $EXP(x)$ standart funksiýalar arkaly aňladylýar. Meselem:

$1,6+2,75*x^{100}$ aňlatmany Paskalda $1.6+2.75*EXP(100*LN(x))$ görnüşde ýazmak bolar.

Aňlatmanyň bahasy hökmünde «TRUE» ýa-da «FALSE» logiki hemişelikler alynýan bolsa, onda onuň ýaly aňlatmalara logiki aňlatmalar diýilýär. Iki sany arifmetiki aňlatmanyň arasynda $<$, $>$, $<=$, $>=$, $=$, gatnaşyk belgileriniň birden-birini ulanmak bilen ýönekeý logiki



aňlatmany ýazmak bolýar. Meselem:

$$Y1+3.95 \leq X2 + \text{SQR}(A);$$

$$6.25 + 4.75 < 17.75.$$

Has çylşyrymly logiki aňlatmalar ýönekeý logiki aňlatmalardan we NOT, AND, OR, XOR logiki amallardan düzülýär. Meselem:

$$(B > 0.85) \text{ AND } (I < 4.5)$$

Logiki amallar aşakdaky hakykylyk tablisasy bilen kesgitlenilýär.

<i>A</i>	<i>B</i>	<i>A</i> and <i>B</i>	<i>A</i> or <i>B</i>	<i>A</i> xor <i>B</i>	NOT <i>A</i>
<i>T</i>	<i>T</i>	<i>T</i>	<i>T</i>	<i>F</i>	<i>F</i>
<i>T</i>	<i>F</i>	<i>F</i>	<i>T</i>	<i>T</i>	<i>F</i>
<i>F</i>	<i>T</i>	<i>F</i>	<i>T</i>	<i>T</i>	<i>T</i>
<i>F</i>	<i>F</i>	<i>F</i>	<i>F</i>	<i>F</i>	<i>T</i>

Bu ýerde *A* we *B* – logiki tipli operantlar, *T* – True, *F* – False, and, or, xor – logiki amal belgileri.

PASKAL dilinde programmanyň gurluşy

Paskal algoritmik dilinde programma sözbaşydan we bölümlerden ybarat bolýar. Programmanyň sözbaşysy umumy görnüşde aşakdaky ýaly ýazylýar:

```
PROGRAM <programmanyň ady>;
```

Bu ýerde, <programmanyň ady> - ulanyjy tarapyndan girizilýän islendik at.

Bölümleriň sany 7-ä deňdir:

- Modullaryň beýan edilýän bölümi;
- Belgileriň beýan edilýän bölümi;
- Hemişelikleriň kesgitlenyän bölümi;
- Tipleriň kesgitlenyän bölümi;
- Üýtgeýän ululyklaryň beýan edilýän bölümi;



- Funksiýalar we proseduralar bölümi;
- Operatorlar bölümi.

Modullaryň beýan edilýän bölümi Uses sözi bilen başlanýar we onuň yzyndan programmada ulanyljak modullaryň atlary ýazylýar.

Mysal üçin,

Uses crt;

Belgileriň beýan edilýän bölümi umumy ýagdaýda

LABEL n1,n2,...,nk;

görnüşde ýazylýar. Bu ýerde LABEL-bölümiň ady bolup «belgi» diýen manyny aňladýar; n1,n2,...,nk – programmada ulanylýan belgileriň sanawy. Paskal algoritmik dilinde belgi hökmünde atlary we bitin položitel sanlary ulanmak mümkin. Belgileriň sanawyny tertip boýunça ýazmak hökman däl, ol islendik tertipde ýazylyp bilner. Programma ýazylanda belgi bilen operatoryň arasynda « : » – goşa nokat belgisi goýulýar.

Meselem:

LABEL 10;

.....

10:X:=X+1;

.....

Eger programmada belgi ulanylmaýan bolsa, onda bu bölüm ýazylymaýar. Hemişelikleriň kesgitlenilýän bölümi umumy ýagdaýda

CONST a1=c1; a2=c2; ..., an=cn;

görnüşde ýazylýar. Bu ýerde: CONST – bölümiň ady, a1, a2, ..., an – programmada ulanylýan hemişelikleriň atlary, c1, c2, ..., cn – deňşlilikde a1, a2, ..., an hemişelikleriň eýe bolýan bahalary.

Her bir hemişelige baha berlenden soň hökman « ; » – nokatly otur belgisi goýulmagy zerurdyr. Hemişelik bilen onuň bahasynyň arasynda « = » – deňdir belgisi goýulýar.



Meselem:

CONST NMIN = 1; NMAX = 100; a = 5.23;

Hemişelikleriň tipi olaryň eýe bolýan bahalarynyň tipleri boýunça kesgitlenýär. Ýokardaky ýazgyda NMIN we NMAX - bitin tipe, a bolsa hakyky tipe degişlidir. Eger programmada hiç hili hemişelik girizilmedik bolsa, onda bu bölüm ýazylmaýar.

Tipleriň kesgitlenýän bölümi umumy ýagdaýda aşakdaky ýaly ýazylýar:

TYPE T1= <tipiň mazmuny>; T2= <tipiň mazmuny>; ...,
Tn=<tipiň mazmuny>;

Bu ýerde: TYPE – bölümiň ady bolup «tip» diýen manyny aňladýar; T1, T2,..., Tn – kesgitlenýän tipleriň atlary.

Meselem:

TYPE

TOMUS = (Iýun, Iýul, Awgust);

MASHGALA = (Kaka, Eje, Dogan, Jigi);

VAR

A: TOMUS; B=MASHGALA;

Üýtgeýän ululyklaryň beýan edilýän bölümi umumy ýagdaýda:

VAR X11, X12,..., X1n: T1; X21, X22,..., X2n: T2;..., Xm1,
Xm2,...,Xmn: Tm;

görnüşde ýazylýar. Bu ýerde: VAR – bölümiň ady bolup «üýtgeýän ululyk» diýen manyny aňladýar. X1i (i=1, n) T1 tipe degişli bolan üýtgeýän ululyklaryň atlary, X2i (i=1, n) T2 tipe degişli bolan üýtgeýän ululyklaryň atlary, Xmi (i=1,n) Tm tipe degişli bolan üýtgeýän ululyklaryň atlary.

Meselem:

VAR

I, J, K: INTEGER; S, T: REAL; H: CHAR; B1, B2 : BOOLEAN;

ýazgy I, J, K – üýtgeýän ululyklaryň diňe bitin tipe degişli bolan ba-



halary, S, T – üýtgeýän ululyklaryň diňe hakyky tipe degişli bahalary, H – üýtgeýän ululygyň diňe liter tipe degişli bolan bahalary we B1, B2 – üýtgeýän ululyklaryn bolsa diňe logiki tipe degişli bolan bahalary kabul edip bilýändigini aňladýar.

Funksiýalar we proseduralar bölüminde programmada ulanylýan standart däl funksiýalar we proseduralar beýan edilýär. Funksiýanyň sözbaşysy aşakdaky ýaly ýazylýar:

FUNCTION <funksiýanyň ady> (q1:T1; q2:T2; ...; qn:Tn): T;

Bu ýerde: q_i ($i=1,n$) – formal parametrler; T_i ($i=1,n$) degişlilikde q_i ($i=1,n$) parametrleriň tipleri; T – funksiýanyň ýerine ýetirilmeginde alynýan netijäniň tipi.

Meselem:

FACTORIAL n!-i hasaplamak üçin düzülen funksiýa bolsa, onda ol funksiýalar we proseduralar bölüminde

FUNCTION FACTORIAL (K:INTEGER): INTEGER;
sözbaşy görnüşinde ýazylýar. Eger-de programmada prosedura ulanylýan bolsa, onda ol

PROCEDURE <proseduranyň ady> (formal parametrleriň sanawy); görnüşindäki sözbaşy bilen ýazylýar. Formal parametrleriň sanawynda giriş-çykyş parametrleri we olaryň tipleri görkezilýär.

Meselem:

KWU – kwadrat deňlemäniň hakyky köklerini tapmagyň prosedurasyna bolsa, onda ol funksiýalar we proseduralar bölüminde aşakdaky ýaly ýazylýar:

PROCEDURE KWU (a, b, c: REAL; VAR X1, X2: REAL);

Bu ýerde: a, b, c – giriş parametrleri. Çykyş parametrleri üýtgeýän ululyk hökmünde ygylan edilýär. Şonuň üçin olaryň önünden VAR sözi goýulýar. Proseduranyň adynyň tipi kesgitlenilmeýär, sebäbi, ol bu ýerde hiç hili baha eýe bolmaýar.

Operatorlar bölümi BEGIN sözi bilen başlanýar we END sözi bilen gutarýar. Bölümiň ahyryny görkezýän END sözünden soň hökman « . » – nokat belgisi goýulmaly. Bu bölüm programmanyň ýerine ýetirilýän bölümi bolup, ol arasynda « ; » – nokatly otur belgisi goýulýan operatorlaryň zygiderlilikinden durýar. Operatorlar özleri-



niň gurluşy boýunça ýönekeý we düzme operatorlara bölünýär.

Özünde diňe bir sany operator saklaýan operatorlara ýönekeý operatorlar diýilýär. Meselem: geçiş operatory, baha bermek operatory we ş.m.

Paskal algoritmik dilinde birnäçe operatorlary birikdirip, bir operator hökmünde hem ýazmak bolýar we netijede alnan operatora düzme operator diýilýär. Düzme operator umumy ýagdaýda

BEGIN P1; P2; ...; Pn END.

görnüşinde ýazylýar. Bu ýerde P1; P2; ...; Pn – Paskal diliniň islendik operatorlary. Başgaça, BEGIN we END sözleriň jübütine operatorlar ýaýy hem diýilýär, BEGIN – açylyan ýaýy, END – ýapylyan ýaýy aňladýar.

Meselem:

BEGIN Y: =SQR(x)/2; x:=x+1 END;

BEGIN K: =5; BEGIN J:=0; AK: =0 END; END.

Operatorlar bölümüniň özüne hem düzme operator hökmünde setretmek mümkindir.

Paskal dilinde tipler

Skalýar tipler

Tip düşüňjesi programmirlenmegiň ähli dillerinde hem esasy düşüňjeleriň biri hasaplanylýar. Tip diýip ululyklaryň kabul edip bilýän bahalarynyň köplüğine düşünilýär. PASKAL algoritmik dilinde tipleri iki uly topara bölmek mümkin: 1) skalýar tipler; 2) strukturaly tipler. Eger obýekt diňe bir sany komponentden ybarat bolsa, onda oňa skalýar tipe degişli diýilýär. Birden köp, birnäçe komponentden ybarat bolan obýektler strukturaly tipe degişlidir.

Skalýar tipleri hem standart we standart däl skalýar tiplere bölmek bolýar. Standart skalýar tipe bitin, hakyky, CHAR we BOOLEAN tipler degişlidir. INTEGER tipi (-32768, 32767) aralyga degişli bolan ähli bitin sanlaryň köplügi bilen kesgitlenilýär. Bitin tip

< at, ... > : INTEGER;

görnüşinde yglan edilýär.



Meselem:

VAR

BAHA, SANAW_N : INTEGER;

X_KOORD, Y_KOORD, Z_KOORD: INTEGER;

Bu ýazgy BAHA, SANAW_N, X_KOORD, Y_KOORD, Z_KOORD – 5 sany üýtgeýän ululygyň diňe (-32768,32767) aralykda bolan bitin bahalara eýe bolup bilýändigini aňladýar. Mysal üçin: BAHA:=4.45 baha bermek operatorynyň ýazylyşy nädogry. SANAW_N:=295 operator dogry ýazylan.

BYTE (byte) tipine [0;255] aralykdaky bitin sanlar degişlidir. Bu tipdäki ululyklar üýtgeýän ululyklaryň beýan edilýän bölümünde aşakdaky ýaly yglan edilýär:

VAR <ady,...>:BYTE;

Meselem:

VAR JEM : BYTE;

MIN, MAX, INDEKS: BYTE;

Bu ýazgylarda JEM,MIN,MAX we INDEKS atly ululyklara diňe [0;255] aralykdaky bitin bahalary berip bolýandygyny aňladýar. Arifmetiki aňlatmalarda BYTE we INTEGER tipli ululyklaryň utgaşyp gelmegine-de rugsat edilýär. Bitin tipleriň arasynda iň uly bahalara eýe bolýany LONGINT tipidir. Onuň çäkleri -2147483648-den +2147834647-ä çenlidir.

Hakyky tipe degişli bolan üýtgeýän ululyklar aşakdaky görnüşde yglan edilýär:

VAR <ady>: REAL ;

Meselem:

VAR JEM, NETIJE : REAL ;

ýazgy JEM we NETIJE ütgeýän ululyklaryň diňe hakyky tipe degişli bolan bahalara eýe bolup bilýändigini aňladýar. Arifmetiki aňlatmalarda bitin we hakyky tipli ululyklaryň utgaşyp gelmegine hem rugsat edilýär. Hakyky tipli ütgeýän ululyklary massiwiň indeksi we



köplügiň elementi hökmünde ulanmak bolmaýar. Logiki tipe degişli bolan üýtgeýän ululyklar aşakdaky ýaly yglan edilýär:

VAR <ady> : BOOLEAN ;

Meselem:

VAR S, T : BOOLEAN ;

ýazgy S we T üýtgeýän ululyklaryň logiki tipe degişlidigini aňladýar. Logiki tipe degişli bolan üýtgeýän ululyklar diňe «TRUE» (çyn) ýa-da «FALSE» (ýalan) bahalaryň birden-birini kabul edip bilýär.

Meselem:

eger S: = 12 < 17 we T: = 10 > 14 bolsa, onda S - «TRUE», T - «FALSE» baha eýe bolýar. Paskal algoritmik dilinde TRUE we FALSE standart logiki hemişelik hökmünde hem ulanylýar.

Liter (simwol) tipe degişli bolan üýtgeýän ululyklar aşakdaky ýaly yglan edilýär:

VAR <ady> : CHAR ;

Meselem:

VAR HARP, SIMWOL : CHAR ;

ýazgy HARP we SIMWOL üýtgeýän ululyklaryň diňe simwol bahany kabul edip bilýändigini aňladýar.

Paskalda simwol baha diýlende diliň elipbiýine girýän islendik simwola – latyn harplaryna, onluk sanlara, amallaryň belgilerine, ýörite belgilere we ş.m. düşünilýär.

Meselem:

HARP: = 'M' ;

SIMWOL: = '*' ;

ýa-da

SIMWOL: = ' ' – boş öýjük we ş.m.

CHAR tipe degişli bolan üýtgeýän ululyklary arifmetiki aňlatmalarda ulanmak rugsat edilmeyär. Paskal algoritmik dilinde CHAR tipe degişli bolan üýtgeýän ululyklary aşakdaky ýaly deňeşdirmek bolýar:

'k' > 'b' – TRUE baha eýe, sebäbi, kod tablisasynda 'k' harpy 'b'-den soň gelýär. Şoňa görä-de, liter ululyklaryň üstünde CHR,



ORD, PRED we SUCC funksiýany ýerine ýetirip bolýar.

Paskal algoritmik dilinde ulanyjynyň özüne hem tip kesgitlemäge rugsat edilýär. Şonuň ýaly tiplere standart däl tipler ýa-da ulanyjynyň tipi diýilýär. Standart däl skalýar tipleri girizmeklik ýazylýan programmany sadalaşdyrýar we maşynyň ýadyny tygşytlamaga mümkinçilik berýär.

Elementleri sanalyp geçilýän tip aşakdaky ýaly yglan edilýär:
TYPE < tipiň ady > = (< 1-nji element, 2-nji element, ... , n-nji element >);

VAR < ady > : < tipiň ady > ;

Meselem:

TYPE

HEPDE = (DS, SS, CS, PS, AN, SB, YS);

FAKULTET = (FIZ, MAT, HIM, TAR, BIOL, GEOGR);

TOPAR = (G101, G102, G103, G104);

Bu ýerde: ýaýyň içinde ýazylan bahalara görkezilen tipe degişli bolan hemişelikler hökmünde hem seretmek bolýar. Şonuň üçin hem bu bahalar ulanylanda hemişelikler bilen işlemegiň düzgüninden ugur almaly. Elementleri sanalyp geçilýän tip tertipleşdirilen tiplere degişlidir, ýagny onuň elementlerini deňeşdirip bolýar.

Meselem:

G102 < G104 – TRUE baha eýe, G103 < G101 bolsa FALSE baha eýedir. Diýmek, bu tipe degişli bolan bahalary SUCC, PRED, ORD standart funksiýalarda hem ulanyp bolýar.

Bellik. Elementleri sanalyp geçilýän tipe degişli bolan bahalary giriş-çykyş operatorlarynda ulanmak bolmaýar.

Çäklendirilen tipli üýtgeýän ululyklar aşakdaky ýaly yglan edilýär:

TYPE < tipiň ady > = m1 ... m2 ;

VAR < üýt. ulul. ady > : < tipiň ady > ;

Bu ýerde: m1 we m2 hemişelikler, degişlilikde berlen tipe degişli



bolan üýtgeýän ululyklaryň kabul edip biljek iň kiçi we uly bahalaryny görkezýärler. m1 we m2 hemişelikler hakyky tipe deňişli bolmaly däl, CHAR, BOOLEAN, bitin tipleriň islendigine deňişli bolup biler.

Meselem:

```
TYPE DAY = 1.. 31 ;
```

```
Harp = ' a ' .. ' e ' ;
```

```
VAR WORKING_DAYS, FREE_DAYS :DAY;
```

```
X1, X2 : harp ;
```

ýazgy WORKING_DAYS, FREE_DAYS- atly üýtgeýän ululyklaryň diňe 1-den 31-e çenli aralykdaky bitin bahalara eýe bolýandygyny, X1 we X2 üýtgeýän ululyklaryň bolsa, diňe ' a ', ' b ', ' c ', ' d ' we ' e ' simwollaryň birden-birine eýe bolýandygyny aňladýar. Köplenç çäklendirilen tip aşakdaky görnüşde yglan edilýär:

```
CONST MIN = 1; MAX = 31;
```

```
TYPE
```

```
DAY := MIN .. MAX ;
```

```
VAR WORKING_DAYS, FREE_DAYS : DAY ;
```

Beýan edilen ýagdaýda interwalyň çäklerini çalyşmak üçin diňe hemişelikler bölümini üýtgetmek ýeterlik bolýar.

Strukturaly tipler barada düşüňje

Strukturaly tipler skalýar tipe deňişli bolan üýtgeýän ululyklaryň toplумы hökmünde kesgitlenilýär. Strukturaly tip özüniň komponentleriniň tipleri bilen häsiýetlendirilýär. Paskal algoritmik dilinde ulanylýan strukturaly tiplere setir tipini, köplükleriň kombinirlenen tipini, massiwleri, faýllary deňişli etmek mümkin.

Diliň elipbiýine girýän simwollaryň tükenikli yzygiderliligine setir hemişeligi diýilýär. Setir hemişelikleri hem edil simwol hemişelikleri ýaly « ' » - apostrof belgisiniň içinde ýazylýar. Meselem:



'Jemi', 'Y=', 'PASKAL' we ş.m.

Baha hökmünde diňe setir hemişelikleri kabul edip bilýän üýtgeýän ululyklara setir üýtgeýän ululyklary diýilýär.

Paskal algoritmik diliniň esasy operatorlary

Baha bermek operatory

Baha bermek operatory aňlatmanyň bahasyny üýtgeýän ululygyň öýüne ýazmak üçin ulanylýar. Ol umumy ýagdaýda aşakdaky görnüşde ýazylýar:

$$A:=B;$$

bu ýerde «:=» – baha bermek operatorynyň belgisi, A – üýtgeýän ululygyň ady; B – aňlatma, hususy ýagdaýda üýtgeýän ululyk ýa-da hemişelik hem bolup biler. Baha bermek operatoryny ýerine ýetirmek iki tapgyrdan durýar: birinji tapgyrda B aňlatmanyň bahasy hasaplanylýar; ikinji tapgyrda – alnan netije A – üýtgeýän ululygyň bahasy hökmünde kabul edilýär.

Baha bermek operatorynyň sag tarapyndaky B aňlatma we çepindäki A üýtgeýän ululyk şol bir tipden bolmaly. Eger B aňlatma bitin ýa-da hakyky tipe degişli bolsa, onda baha bermek operatoryna arifmetiki baha bermek operatory diýilýär.

Eger arifmetiki aňlatmada A – üýtgeýän ululyk hakyky tipe degişli bolsa, onda B – aňlatma hakyky we bitin tipleriň islendigine degişli bolup bilýär. Eger A – üýtgeýän ululyk bitin tipe degişli bolsa, onda B – aňlatma hem hökman şol tipe degişli bolmaly. Arifmetiki baha bermek operatoryna mysallar:

VAR

K:INTEGER;

A,B,Y,V, R:REAL;

.....

K:=K+1;

Y:=SQRT(SQR(A)+SQR(B)*B);



V:=2*PI*R;

Eger B - logiki aňlatma bolsa, onda baha bermek operatoryna logiki baha bermek operatory diýilýär.

Meselem:

VAR

S, T :BOOLEAN;

.....

S:=(3.6+4.3)<> (4.9+0.2);

T:=(6>2) AND (7<0.9);

operatorlar ýerine ýetirilende S - logiki üýtgeýän ululyk TRUE baha, T - logiki üýtgeýän ululyk bolsa FALSE baha eýe bolar. Görnüşi ýaly, logiki aňlatmalar şol bir wagtda özünde arifmetiki amallary saklap bilýärler.

Logiki aňlatmalarda amallar aşakdaky tertipde ýerine ýetirilýär:

NOT (inkär etme);

*, /, DIV, MOD, AND – multiplikatiw amallar;

+, -, OR, XOR – additiw amallar;

<, <=, >, >=, <> - gatnaşyk amallary.

Bu ýerde gatnaşyk amallaryna iň pes dereje berilýär. Deň derejeli amallar çepden saga tertip boýunça ýerine ýetirilýär. Amallaryň ýerine ýetiriliş tertibini üýtgetmek üçin ýaýlardan peýdalanylýar.

A:= B baha bermek operatorynda B - aňlatmanyň CHAR ýa-da STRING tipe degişli bolmagy hem mümkin.

Meselem:

VAR

M, N : CHAR; C1, C2 :STRING [10];

BEGIN

N:= 'E' ; C1:= 'TOPAR' ;

C2:=C1+'L' ; M:=SUCC(L);



Giriş-çykyş operatorlary

Kompýuterde giriş gurluşy hökmünde esasan klawiaturalardan peýdalanylýar. Üýtgeýän ululyklara klawiaturadan baha girizmek üçin aşakdaky operatorlar ulanylýar:

```
READ (X1,X2,...,Xn);
```

```
READLN (X1,X2,...,Xn);
```

Bu yerde $X_i(i=1,n)$ – giriş sanawynyň elementleri ýa-da bahalary girizilýän üýtgeýän ululuklaryň atlary. READ – operatoryň ady bolup, «OKAMAK» diýen manyny aňladýar. READ sözünüň yzyndaky «LN» – setir manyny aňladýar. Ýagny, READLN operatorynyň sanawyndaky üýtgeýän ululyklara baha girizilenden soň, şol setirden başga baha girizip bolmaýar.

READ we READLN operatoryndaky üýtgeýän ululyklaryň tipi degişlilikde olaryň eýe bolýan bahalarynyň tipi bilen gabat gelmeli. Bir setirden birnäçe baha girizilende olaryň arasynda « » - boşluk goýulýar. Mysal hökmünde A, B, C, D hakyky üýtgeýän ululyklara degişlilikde 3.8, -9.6, 12.52 we 495.0 bahalaryň berilmegine seredeliň:

```
a) VAR A, B, C, D: REAL;  
BEGIN
```

```
    READ (A, B, C, D);...
```

Bu ýerde ýokardaky bahalar klawiaturada aşakdaky ýaly girizilýär:

```
3.8 -9.6 12.52 495.0 «ENTER» – klawiaturanyň düwmejigini basmaly.
```

```
b) READ(A,B,C); READ(D);
```

Bu ýagdaýda bahalar aşakdaky ýaly girizilýär:

```
3.8 -9.6 12.52 «ENTER»
```

```
495.0 «ENTER»
```

```
ç) READLN; READLN(A,B); READ(C,D);
```

Bu ýagdaýda bahalar aşakdaky görnüşde girizilýär:



«ENTER»

3.8 –9.6 «ENTER»

12.52 495.0 «ENTER»

Çykyş operatory aşakdaky ýaly ýazylyar:

```
WRITE (a1[:b1[:c1]], a2[:b2[:c2]],...an[:bn[:cn]]);
```

```
WRITELN (a1[:b1[:c1]], a2[:b2[:c2]],...,an[:bn[:cn]]);
```

```
WRITELN;
```

bu ýerde: a_1, a_2, \dots, a_n – bahalary çap edilýän üýtgeýän ululyklaryň atlary; b_1, b_2, \dots, b_n – degişlilikde a_1, a_2, \dots, a_n üýtgeýän ululyklaryň bahalaryny çap etmäge niýetlenen öýjüklerniň sany; c_1, c_2, \dots, c_n – bolsa, degişlilikde a_1, a_2, \dots, a_n üýtgeýän ululyklaryň bahalarynyň drob (galyndy) bölegi üçin niýetlenen öýjüklerniň sanyny görkezýär. a_1, a_2, \dots, a_n bitin, hakyky, CHAR, BOOLEAN, STRING tipleriň islendigine degişli bolup biler. b_i, c_i ($i=1, n$) parametrler hususy ýagdaýda bitin tipe degişli hemişelikler, umuman, bitin tipli üýtgeýän ululyklar, aňlatmalar görnüşinde berlip bilner. b_i we c_i ($i=1, n$) parametrleriň üýtgeýän ululyklary özünde saklap bilmegi netijäni tablisalar, çyzgylar görnüşinde çap etmäge mümkinçilik berýär. Meselem, goý, $[1, 20]$ kesimde $y = -x$ funksiýanyň çyzgysyny şekillendirmeli bolsun. Onda programmany aşakdaky görnüşde ýazmak mümkin:

```
PROGRAM A1;
```

```
VAR X: INTEGER;
```

```
BEGIN
```

```
FOR X: =1 TO 20 DO WRITELN ('X': X) ;
```

```
END.
```

Bellik: c_1, c_2, \dots, c_n – parametrler diňe REAL tipe degişli bolan bahalary çap etmekde ulanylýar; Operatoryň umumy ýazgysynda « [» , «] » – kwadrat ýaýlaryň goýulmagy şol parametrleriň hökmany dældigini aňladýar.

Hususy ýagdaýda çykyş operatoryny aşakdaky görnüşde ýazmak mümkin:



WRITE (a1, a2,... an) ;

WRITELN (a1, a2,..., an) ;

Mysallar.

1) Goý, A, B, C bitin üýtgeýän ululyklar deňlikde 27, 18 we -12 bahalara eýe bolsun. Aşakdaky ýagdaýlara seredeliň:

a) WRITE (A,B) ; WRITE (C) ;

Bu ýagdaýda bahalar bir setirde yzygider çap edilýär, ýagny olaryň arasynda boş öýjük goýulmaýar:

2718-12

b) WRITE (A,B,C) ;

Bu ýagdaýda bahalar edil ýokarda görkezilişi ýaly çap edilýär, hiç hili üýtgeşiklik bolmaýar.

ç) WRITELN (A,B) ; WRITE (C) ;

Bu ýagdaýda aşakdaky görnüşde çap edilýär:

2718

-12

d) WRITE (A:2, B:5 , C:4) ;

Bu ýagdaýda bahalar aşakdaky görnüşde çap edilýär:

27 18 12

e) WRITE ('A=' , A :2, 'B=' , B:2 , 'C=' , C:3) ;

Bu ýagdaýda bahalar aşakdaky görnüşde çap edilýär:

A = 27 B = 18 C = -12

2) Goý, A,B,C hakyky üýtgeýän ululyklar deňlikde 370.555 , 34365.40 we -4.2375 bahalara eýe bolsun.

Aşakdaky ýagdaýlara seredeliň:

a) WRITELN (A:6:2, B:9:2, C:8:4);

Bu ýagdaýda netije aşakdaky görnüşde çap edilýär:

370.56 34365.40 -4.2375



b) WRITELN (A,B,C);

beýle ýagdaýda netije aşakdaky ýaly çap edilýär:

3.7055500000E+2 3.4365400000E+04 -7.2375000000E00.

3) Goý, $A < B$ logiki aňlatmanyň bahasyny çap etmeli bolsun:

a) Eger $A:=8$; $B:=12$ bolsa, onda WRITELN ($A < B:8$); operator netijäni TRUE görnüşinde çap edýär.

b) $A:=5$; $B:=2$ bolsa, onda netije FALSE bolar. Bir setirde birnäçe baha çap edilende olaryň arasyňy açmak üçin « » - boş öý belgiden peýdalanylýar.

Meselem:

WRITE (A,B); operatoryň ornuna WRITE (A, ' ': 4, B); operatory ýazmak mümkin.

Soňky ýagdaýda A we B üýtgeýän ululyklaryň bahalarynyň arasynda 4 sany boş öýjük galdyrylýar. Mysallardan görnüşi ýaly, WRITELN operatory WRITE operatoryndan tapawutlanyp, onuň sanawyndaky üýtgeýän ululyklaryň bahalary çap edilenden soň, täze setire geçilýär, ýagny şol setirden başga hiç hili baha çap edip bolmaýar.

Çyzykly algoritmleri programmirmek üçin diňe baha bermek we giriş-çykyş operatorlaryny bilmek ýeterlidir.

Mysal. Geronyň formulasyndan peýdalanyňp, taraplary A, B, C bolan üçburçlugyň meýdanyny hasaplamak üçin programma düzmeli.

Programması:

```
PROGRAM GERON;
```

```
USES CRT;
```

```
VAR A,B,C,P,S: REAL;
```

```
BEGIN
```

```
CLRSCR;
```

```
WRITELN (' A,B,C, bahasyny giriz');
```



```

READ (A,B,C) ;
P:= ( A+B+C) / 2;
S:=SQRT (P* (P-A) * (P-B)*(P-C));
WRITELN ('P=',P:5:2, ' S=', S:7:3);
END.

```

Geçiş operatory

Eger-de algoritmda käbir şertiň ýerine ýetýändigine ýa-da ýetmeýändigine baglylykda hasaplaýyş prosesi iki ýa-da birnäçe şaha bölünýän bolsa, onda onuň ýaly algoritmlere şahalanýan gurluşly algoritmler diýilýär.

Programmanyň içinde geçiş operatory, esasan, şahalanýan gurluşly algoritmleri programmirmek üçin ulanylýar. Başgaça aýdylanda, geçiş operatory programmadaky operatorlaryň ýerine ýetiriliş tertibini dolandyrmak üçin hyzmat edýär.

Geçiş operatory umumy ýagdaýda aşakdaky görnüşde ýazylýar:

```
GOTO <belgi>;
```

Bu ýerde GOTO – operatoryň ady bolup, «geçmek» diýen manyny aňladýar; <belgi> – salgylanylýan operatoryň belgisi, ol LABEL bölümünde yglan edilýän bolmaly. Meselem:

```

LABEL 20;
.....GOTO 20;
.....
20: X:=X+1;
.....

```



Şertli operator

Şertli operator özünde käbir logiki aňlatmany saklap, onuň «TRUE» ýa-da «FALSE» bahalaryň haýsysyny kabul edýändigine baglylykda iki şahadan birini saýlap almaga mümkinçilik berýär.

Paskal algoritmik dilinde şertli operator aşakdaky görnüşlerde ýazylýar:

a) IF <şert > THEN S1 ELSE S2;

b) IF <şert> THEN S1,

bu ýerde IF – operatoryň ady bolup, ol «eger» diýen manyny aňladýar, <şert> – käbir logiki aňlatma, S1, S2 – Paskal algoritmik diliniň islen-dik operatorlary.

Bu operatorlar ýerine ýetirilende ilki bilen <şert> - logiki aňlat-manyň bahasy hasaplanylýar. Eger logiki aňlatma «TRUE» baha eýe bolsa, onda a)-ýagdaýda S1-operator; b)-ýagdaýda hem S1-operator ýerine ýetirilýär. Ondan soňra IF operatoryň yz ýanyndaky ilkinji operatora geçilýär. Eger-de logiki aňlatma «FALSE» baha eýe bol-sa, onda a)-ýagdaýda S2-operator ýerine ýetirilýär we ondan soňra IF operatorynyň yzyndaky ilkinji operatora geçilýär; b)-ýagdaýda bolsa, hiç hili operator ýerine ýetirilmezden, gönüden-göni IF operatorynyň yzyndaky ilkinji operatora geçilýär. Paskal algoritmik dilinde bir IF operatorynyň düzüminde başga IF operatorynyň hem ulanylmagyna rugsat edilýär. Bu şahalanmanyň içinde ýene-de şahalanma ulanylýan ýagdaýynda ýüze çykýar.

Meselem:

```
IF X<15 THEN IF X>10 THEN WRITELN (' 10<X<15')
                ELSE WRITELN ('X<=10').
```

IF operatoryny biri-biriniň içinde ulanmaklyk programma düşün-mekligi kynlaşdyrýar. Şoňa görä-de, IF operatoryny biri-biriniň için-de iki-üç gezekden köp ulanmak maslahat berilmeýär.

Mysallar.

1. Funksiýanyň bahasyny hasaplamak üçin programma düzmeli:

$$a) \quad Y = \begin{cases} \sqrt{X}, & \text{eger } X \geq 0 \\ X^3, & \text{eger } X < 0 \end{cases}$$



Programmasy:

Program funksiya;

```
Uses crt;  
Var x, y: real;  
Begin  
  Clrscr;  
  Write('x=');  
  Readln(x);  
  If x>=0 then y:=sqrt(x)  
  Else y:=sqr(x)*x;  
  Writeln('y=',y:8:3);  
End.
```

2. a, b, t ululyklaryň berlen bahalarynda S ululygyň bahasyny hasaplamaly:

$$S = \begin{cases} at + b, & \text{eger } at < 1; \\ \cos(at), & \text{eger } at = 1; \\ e^{-at}\cos(at), & \text{eger } at > 1. \end{cases}$$

Programmasy:

```
Program funk1;  
Uses crt;  
Var a, b, t, s: real;  
BEGIN  
  Clrscr;  
  Write('a, b, t bahasyny giriz');  
  Readln(a,b,t);  
  IF A*T<1 then S:=A*T+B  
  Else if A*T=1 then S:=cos(A*T)
```



```
Else S:=exp(-A*T)*cos(A*T);  
Writeln('S=', S:8:3);  
End.
```

Bellik.

Birnäçe IF operatory biri-biriniň içinde ulanylanda her bir ELSE kömekçi sözi özüniň öň ýanyndaky THEN kömekçi sözüne degişli edilýär.

Mysal. Kwadrat deňlemäniň hakyky köklerini tapmak üçin programma düzmeli.

Programmasy:

```
PROGRAM KWADRATD;  
Uses CRT;  
Var A,B,C,D,X1,X2 : real;  
BEGIN  
CLRSCR;  
Write('A,B,C bahasyny giriz');  
Readln(A,B,C);  
D:=SQR(B)-4*A*C;  
IF D<0 then Writeln('hakyky köki ýok')  
ELSE begin  
X1:=(-B+SQRT(D))/(2*A);  
X2:=(-B-SQRT(D))/(2*A);  
Writeln('X1=',X1,'X2=',X2);  
end;  
END.
```



Saýlaw operatory

Belli bolşy ýaly, IF operatory iki şahadan birini saýlap almaga mümkinçilik berýär. CASE saýlaw operatory bolsa ikiden hem köp birnäçe şahadan birini saýlap almaga mümkinçilik berýär. Saýlaw operatory umumy görnüşde aşakdaky ýaly ýazylyar:

CASE C OF

N1:p1;

N2:p2;

....

Nm:pm;

Else p

End,

bu yerde C – saýlaýjy parametr, başgaça selektor hem diýilýär. C selektor bitin, CHAR, BOOLEAN standart skalýar tiplere we standart däl skalýar tipler bolan, elementleri sanalyp geçilýän ýa-da çäklendirilen tipe degişli bolup biler. Islendik ýagdaýda C-selektor bilen N1, N2,..., NM hemişelikleriň tipi gabat gelmeli. N1, N2,..., NM hemişelikleri degişlilikde operatorlaryň belgileri hökmünde göz öňüne getirmek mümkin; P1, P2,..., PM - Paskal algoritmik diliniň islendik operatory.

Saýlaw operatory ýerine ýetirilende ilki C-selektoryň bahasy hasaplanylýar, ony CB arkaly belgiläliň. Soňra CB bahanyň N1, N2,..., Nm - belgileriň haýsysy bilen gabat gelyändigine baglylykda P1, P2,..., Pm operatorlaryň birden - biri ýerine ýetirilýär, eger CB N1, N2,..., Nm belgileriň hiç biri bilen gabat gelmese, onda else sözüň zyndaky P operator ýerine ýetirilýär, eger-de else sözi ýok bolsa, onda bu ýagdaýda hiç hili iş edilmezden CASE operatory tamamlanylýar we dolandyryş CASE operatorynyň zyndaky ilkinji operatora geçirilýär.

Islendik ýagdaýda hem, saýlanyp alnan operator ýerine ýetirilenden soň, beýleki operatorlardan sowlup CASE operatorynyň zyndaky ilkinji operatora geçirilýär.



Bellik. CASE operatoryndaky belgileri LABEL bölümünde ygılan etmeli däl we olara programmanyň başga böleginden GOTO operatory arkaly salgylanmak bolmaýar.

Mysallar

1. Çärýek nomeri boýunça nokadyň koordinatalarynyň almatyny kesgitlemeli.

Programmasy:

```
PROGRAM NOMER;
```

```
USES CRT;
```

```
VAR N:INTEGER;
```

```
BEGIN
```

```
  CLRSCR;
```

```
    WRITE('N=');
```

```
    READLN(N);
```

```
    CASE N OF
```

```
      1:WRITELN('X>0 we Y>0');
```

```
      2: WRITELN('X<0 we Y>0');
```

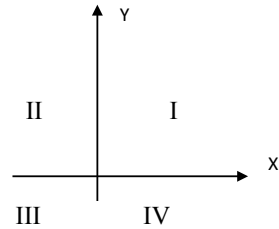
```
      3: WRITELN('X<0 we Y<0');
```

```
      4: WRITELN('X>0 weY<0');
```

```
    ELSE WRITELN ('N-e 1,2,3,4 bahalaryň birini ber');
```

```
    END;
```

```
  END.
```



2. Hepdäniň günleriniň atlaryny çap etmek üçin programma düz-meli.

Programmasy:



```

PROGRAM HEPDE;

USES CRT;

VAR N : INTEGER;

BEGIN

CLRSCR;

    READ (N);
    CASE N OF
        1: WRITELN ('Duşenbe');
        2: WRITELN ('Sişenbe');
        3: WRITELN ('Çarşenbe');
        4: WRITELN ('Penşenbe');
        5: WRITELN ('Anna');
        6: WRITELN ('Şenbe');
        7: WRITELN ('Ýekşenbe');
    ELSE WRITELN ('N-e 1,2,3,4,5,6,7, bahalaryň
birini giriz');
    END;          END.

```

Saýlaw operatorynda n_1, n_2, \dots, n_m hemişelikleriň her biriniň bir belgiden däl-de, eýsem arasynda «,» – otur belgisi goýlan birnäçe belgiden, ýagny belgileriň sanawyndan ybarat bolmagy-da mümkin.

Mysala ýüzlenenliň:

```

PROGRAM MYSAL;

USES CRT;

VAR N :INTEGER;

BEGIN

CLRSCR;

```



```

READ (N);
CASE N OF
1: WRITE ('N=1');
2: WRITE ('N=2');
3,4: WRITE ('N=3 ýa-da N=4');
ELSE WRITE ('N-e 1,2,3,4 bahalaryň
birini bermeli');
End;           END.

```

Bu ýerde WRITE ('n=3 ýa-da n=4'); operatoryň ýerine ýetirilmeği üçin N-iň 3 ýa-da 4 bahalaryň birine deň bolmagy ýeterlidir.

Gaýtalanmak operatorlary. FOR operator

Eger programmanyň haýsy-da bolsa bir bölegi käbir üýtgeýän ululygyň dürli bahalarynda birnäçe gezek gaýtalanyp ýerine ýetirilýän bolsa, onda onuň ýaly programmalara gaýtalanýan gurluşly programmalar diýilýär. Programmanyň gaýtalanyp ýerine ýetirilýän bölegine, ýagny operatorlaryň toplumyna gaýtalanmanyň göwresi diýilýär. Gaýtalanma mahalynda her gezek öz bahasyny üýtgedip durýan ululyga gaýtalanmagyň parametri diýilýär. Ýörite gaýtalanmany gurnamak üçin niýetlenen operatorlara gaýtalanmak operatorlary diýilýär.

Paskal algoritmik dilinde gaýtalanmak operatorlaryny iki topara bölmek mümkin:

- 1) Parametrli gaýtalanmak operatory.
- 2) Şertli gaýtalanmak operatory.

Parametrli gaýtalanmak operatory aşakdaky iki görnüşde ýazylýar:

- a) FOR i:= m1 TO m2 DO s ;
- b) FOR i:=m1 DOWNT0 m2 DO S;



Bu ýerde FOR – operatoryň ady, «üçin» diýen manyny aňladýar; i – gaýtalanmagyň parametri, m1 we m2 bolsa degişlilikde onuň başlangyç we ahyrky bahalary. Bu parametrleriň üçüsi hem şol bir tipden bolmaly: bitin, CHAR, BOOLEAN ýa-da standart däl skalýar tipler – elementleri sanalyp geçilýän tip, çäklendirilen tip; i-görkezilen tipe degişli bolan üýtgeýän ululyk bolmaly; m1 we m2 bolsa şol tipe degişli bolan hemişelik, üýtgeýän ululyk ýa-da aňlatma hem bolup biler. Eger FOR operatorynda «TO» - kömekçi sözi ulanylýan bolsa, onda a)- ýagdaýda gaýtalanmagyň parametri bir - birden artýar. Hususy ýagdaýda, eger i bitin tipleriň birine degişli bolsa, onda gaýtalanmagyň ädimi $h=1$ hasap edilýär. Eger-de FOR operatorynda «DOWNTO» - kömekçi sözi ulanylýan bolsa, onda gaýtalanmagyň parametri bir - birden kemelýär. Eger i bitin tipleriň birine degişli bolsa, onda $h=-1$ hasap edilýär.

Parametrli gaýtalanmak operator ýerine ýetirilende ilki m1 we m2-aňlatmalaryň bahalary hasaplanylýar. Soňra gaýtalanmagyň parametri m1b baha eýe bolýar ($i:=m1$) we a) ýagdaýda $i\leq m2$ şert barlanylýar, b) ýagdaý üçin bolsa $i\geq m2$ şert barlanylýar. Eger şert ýerine ýetse, onda S-gaýtalanmagyň göwresi ýerine ýetirilýär we a) ýagdaýda gaýtalanmagyň parametri $i:=i+1$ baha eýe bolýar, b) ýagdaý üçin $i:=i-1$ baha eýe bolýar, soňra ýene-de gaýtalanmanyň başyna dolanyp gelinýär-de proses dowam etdirilýär. a) ýagdaýda $i\leq m2$ şert ýerine ýetmese, ýagny $i>m2$ bolsa, onda operatorndan çykylýar we FOR operatoryň yzyndaky ilkinji operatora geçilýär. b) ýagdaýda $i\geq m2$ şert ýerine ýetmese, ýagny $i<m2$ bolsa, onda gaýtalanmadan çykylýar.

Mysallar:

1) FOR k:=1 to 10 DO WRITE (' * ');

Netije:

* * * * * (10 sany ýyldyzjyk)

2) FOR i:=1 to 5 DO WRITELN (sqr(I));

Netije:

1



4

9

16

25

3) FOR i:=3 to 10 DO WRITE (I:3);

Netije: 3 4 5 6 7 8 9 10

4) FOR i:=10 DOWNT0 3 DO WRITE (I:3);

Netije: 10 9 8 7 6 5 4 3

5) FOR k:='B' TO 'F' DO WRITE (K:2);

Netije: B C D E F

6) FOR k:='F' DOWNT0 'B' DO WRITE (k:2);

Netije: F E D C B.

Käbir ýagdaýlarda gaýtalanmanyň sany öňünden berilmeyär, ýagny haçan gaýtalanmak prosesinden çykalydygy programma ýerine ýetirilip duran mahaly belli bolýar. Beýle ýagdaýlarda gaýtalanýan gurluşly algoritmleri programmirmek üçin şertli gaýtalanmak operatorlaryndan peýdalanylýar.

WHILE we REPEAT operatorlary

Şertli gaýtalanmak operatory gurluşy boýunça iki hilli bolýar:

1) Şertli öňünde goýulýan gaýtalanmak operatory;

2) Şerti soňunda goýulýan gaýtalanmak operatory.

Şerti öňünde goýulýan gaýtalanmak operatory umumy görnüşde aşakdaky ýaly ýazylýar:

WHILE <şert> DO S;

Bu ýerde: WHILE – operatoryň ady bolup, «entek» diýen



manyny aňladýar; <şert> – käbir logiki aňlatma; S – ýönekeý ýa-da düzme operator.

WHILE operatorynda ilki şert barlanylýar, eger şert ýerine ýetse, ýagny logiki aňlatma «TRUE» baha eýe bolsa, onda S-gaýtalanmagyň göwresi ýerine ýetirilýär we ýene-de gaýtalanmaga dolanyp gelinýär. Haçan-da logiki aňlatma «FALSE» baha eýe bolsa, onda WHILE operatoryndan çykylýar we onuň yzyndan gelýän ilkinji operatora geçilýär. Bu ýerde gaýtalanmagyň göwresine degişli bolan operatorlaryň iň bolmanda biri logiki aňlatmanyň bahasyna täsir edip bilýän bolmaly. Eger şeýle bolmasa, onda gaýtalanmakdan çykyp bolmaýar.

Mysal. WHILE operatoryny ulanyp, $S = \sum_{k=1}^{30} \frac{1}{k^3}$ jemi hasaplamak üçin programma düzmeli.

Programması:

```
PROGRAM JEM1;  
USES CRT;  
VAR  
JEM: REAL;  
K: INTEGER;  
BEGIN  
CLRSCR;  
JEM:=0; K:=1;  
WHILE K<=30 DO  
BEGIN  
JEM:=JEM+1/(SQR(K)*K);  
K:=K+1;  
END;  
WRITELN('JEM=',JEM:8:3);  
END.
```



Şerti soňunda goýulýan gaýtalanmak operatory umumy görnüşde aşakdaky ýaly ýazylýar:

REPEAT S UNTIL <şert > ,

bu ýerde REPEAT operatoryň ady bolup, ol «gaýtalanmak» diýen manyny aňladýar, UNTIL – kömekçi söz «çenli» diýen manyny aňladýar, S – ýönekeý ýa-da düzme operator, < şert > - käbir logiki aňlatma.

REPEAT operatory WHILE operatoryndan tapawutlanyp, onda ilki bilen gaýtalanmaklygyň göwresi ýerine ýetirilýär, soňra şert barlanylýar. Eger şert ýerine ýetmese, gaýtalanmaklyk dowam etdirilýär, ýagny S-gaýtalanmaklygyň göwresine dolanyp gelinýär. Eger şert ýerine ýetse, onda REPEAT operatoryndan çykylýar we onuň yzindaky ilkinji operatora geçilýär.

Bu ýerde hem edil WHILE operatoryndaky ýaly, gaýtalanmaklygyň göwresine degişli operatoryň iň bolmanda biri barlanylýan şerte täsir edýän bolmaly. Mysal hökmünde ýokarda hasaplanan jeme seredeliň:

$$S = \sum_{k=1}^{30} \frac{1}{k^3}$$

Programmasy:

PROGRAM JEM2;

VAR K: INTEGER;

JEM: REAL;

BEGIN

CLRSCR;

K:=1; JEM:=0;

REPEAT

JEM:=JEM+1/(SQR(K)*K);

K:=K+1 UNTIL K>30;

WRITELN('JEM=', JEM:9:4);

END.



REPEAT operatory WHILE operatory bilen deňeşdirilende aşakdaky üç sany aýratynlyga eýe:

1) REPEAT operatorynda gaýtalanmaklygyň göwresi iň bolmanda bir gezek ýerine ýetirilýär;

2) REPEAT operatorynda şert ýerine ýetmese, ýagny logiki aňlatma «FALSE» bahany alanda gaýtalanmaklyk dowam etdirilýär;

3) REPEAT operatorynda gaýtalanmaklygyň göwresinde düzme operator ýazylanda hem BEGIN, END sözleri goýmak hökman däl.

Käbir ýagdaýlarda gaýtalanmaklygyň göwresiniň bir gezek hem ýerine ýetirilmeyän ýagdaýyny hasaba almaly bolýar. Beýle ýagdaýda diňe WHILE operatoryny ulanyp bolýar. Meselem:

$$\left\{ \frac{1}{K^3} \right\}$$

tükeniksiz san hataryň berlen Eps položitel sandan uly bolan agzalarynyň jemini hasaplamaly.

Programmasy:

```
PROGRAM JEM3;
USES CRT;
VAR K:INTEGER;
      Eps, JEM:REAL;

BEGIN
  CLRSCR;
  K:=1; JEM:=0;
  READ(Eps);
  WHILE(1/(SQR(K)*K))>Eps DO

  BEGIN
    JEM:=JEM+1/(SQR(K)*K);
    K:=K+1; END;
```



```
WRITELN('JEM=', JEM);
```

```
END.
```

Düşündirişler (kommentariler)

Düşündirişler programmanyň ýerine ýetiriliş tertibine hiç hili täsir etmeýär we ol diňe ulanylýan ýazgylara düşündiriş berýär. Paskal algoritmik dilinde düşündirişler { , } ýa-da (* , *) belgileriň kömegi bilen ýazylýar. Düşündirişler özlerinde latyn, rus harplaryny, onluk sanlary we diliň elipbiýine degişli bolan beýleki simwollary saklap biler. Düşündirişleri programmada boş öýjük goýmaga rugsat edilýän islendik ýerde goýup bolýar.

Meselem:

(* Funksiýanyň bahasyny hasaplamaly*)

ýa-da

{Aşgabat-10.01.2015ýyl} we ş.m.

Düşündirişleriň soňundan « ; »-nokatly otur belgisi goýulmaýar. Islendik tekstde düşündirişleriň başlangyjyny ýa-da ahyryny aňladýan belgileri ulanmaly däldir. Mysal üçin, {{ kwadrat deňlemäniň köki ýok}} ýazgy nädogrudyr. Emma (*,*) belgileriň arasynda {,} belgileri ulanmak we tersine rugsat edilýär.

(*{1-nji mysal}*) ýazgy dogrudyr.

Mysal.

```
PROGRAM SYNAG;
```

```
USES CRT;
```

```
{Şaryň göwrümini hasaplamagyň programmasy}
```

```
{PI – standart hemişelik}
```

```
VAR R : INTEGER; {R – radius}
```

```
V : REAL; {V - göwrüm}
```



BEGIN

CLRSCR;

R:=5;

V:=PI*R*SQR(R)*(4/3); {şaryň göwrümi}

WRITELN('V=',V:7:3); END.

Massiwler

Paskal algoritmik dilinde diňe aýratyn alnan bir sany üýtgeýän ululyk bilen işlemek mümkin bolman, eýsem, olaryň toplумы bilen hem işlemek mümkindir. Şeýle toplumlaryň bir görnüşi hem massiwlerdir.

Massiw diýlip, şol bir tipe degişli bolan üýtgeýän ululyklaryň tertipleşdirilen toplumyna düşünilýär. Massiwi düzýän üýtgeýän ululyklar zygyderli, tükenikli we tertipleşdirilen bolmalydyr. Massiwiň elementleriniň sany ol yglan edilen wagtynda fiksirlenýär we programma ýerine ýetirilen döwründe üýtgemeyär. Massiwiň her bir elementine aýratynlykda ýüzlenip bolar ýaly, onuň elementlerine indeks degişli edilýär.

Paskal algoritmik dilinde indeks bitin we CHAR tipleriň islendigine degişli bolup bilýär.

Massiw programmada aşakdaky ýaly beýan edilýär:

TYPE <tipiň ady> =ARRAY [indeksiň çäkleri] OF <komponentleriň tipi>; VAR <massiwiň ady>: <tipiň ady>;

Massiwi başgaça, gönüden-göni üýtgeýän ululyklar bölümünde hem yglan etmek mümkindir:

VAR < massiwiň ady> : ARRAY [indeksiň çäkleri] OF < komponentiň tipi >;

Meselem:

TYPE

VEKTOR = ARRAY [1..15] OF REAL;

VAR



X,Y:VEKTOR;

A,B: ARRAY [1..20] OF INTEGER;

Eger massiwiň komponenti hökmünde ýene-de massiw ulanylýan bolsa, onda onuň ýaly massiwlere iki ölçegli massiw diýilýär. Meselem:

TYPE

MATR : ARRAY[1..10,1..20] of integer; {2- ölçegli massiw}

Iki ölçegli massiw başgaça aşakdaky görnüşde hem yglan etmek mümkin:

VAR

MATR : ARRAY [1..4,1..10] OF INTEGER;

Adatça, bir ölçegli massiwler wektorlar bilen, iki ölçegli massiwler bolsa matrisalar bilen işlemek üçin ulanylýar. Meselem:

VAR

WEKTOR : ARRAY [1..20] OF REAL;

MATR: ARRAY [1..4, 1..6] OF INTEGER,

bu ýerde WEKTOR massiwi 20 sany hakyky komponentden ybarat bolan wektor hökmünde göz önüne getirmek mümkin, MATR massiwe 4 setirden, 6 sütünden ybarat bolan bitin elementli gönüburçly matrisa hökmünde seretmek mümkin.

Köplenç massiwiň indeksiniň çäklerini hemişelikler bölümünde görkezmek amatly bolýar:

CONST

NMIN=1; NMAX=30;

VAR

X : ARRAY[IMIN..IMAX] OF REAL.

Bir ölçegli massiwiň elementleri kompýuteriň ýadynda indeksleri artýan tertipde yzygider, matrisanyň elementleri bolsa setirleri boýunça ýerleşdirilýär.

Meselem, eger

VAR



B : ARRAY[1..4,1..4] OF INTEGER;

bolsa, onda B massiwiň elementleri kompýuteriň ýadynda aşakdaky tertipde ýerleşdirilýär:

$$A[1,1], A[1,2], \dots, A[1,4], A[2,1], A[2,2], \dots, A[2,4], \dots, A[4,1], A[4,2], \dots, A[4,4].$$

Massiwiň ady yglan edilenden soň onuň her bir elementini aýratyn üýtgeýän ululyk hökmünde ulanmak mümkin. Onuň üçin massiwiň adyny we kwadrat ýaýyň içinde elementiň indeksini görkezmek ýeterlikdir. Meselem, MASS[4] ýazgy MASS-bir ölçegli massiwiň 4-nji elementine ýüzlenmäge mümkinçilik berýär, VEKTORA[20] ýazgy bolsa VEKTORA atly bir ölçegli massiwiň 20-nji elementine ýüzlenmäge mümkinçilik berýär, MATRX[4,6] ýazgy MATRX iki ölçegli massiwiň, ýagny gönüburçly matrisanyň 4-nji setiri bilen 6-njy sütüniň kesişmesinde ýerleşen elementine ýüzlenmäge mümkinçilik berýär we ş.m.

Massiwe başgaça indeksli üýtgeýän ululyk hem diýilýär. Indeksli üýtgeýän ululyklary hem edil ýönekeý üýtgeýän ýaly ulanmak mümkin. Meselem, olary operand hökmünde aňlatmanyň düzümine girizip bolýar; FOR, WHILE, REPEAT operatorlarynda sanawyň elementi hökmünde ulanyp bolýar we ş.m.

Massiwiň elementleri bilen işlenilende köplenç ýüze çykaýmagy mümkin bolan ýazgylara seredeliň. Onuň üçin ilki 2 sany massiw we 3 sany kömekçi üýtgeýän ululyk girizeliň:

VAR A : ARRAY[1..5] OF REAL;

Y : ARRAY[1..15,1..20] OF INTEGER;

I,J,K : INTEGER;

- 1) Bir ölçegli A massiwi nullamak (ähli elementlerine 0 baha berilýär);

FOR I:=1 TO 5 DO A[I]:=0;

bu operator A[1]:=0; A[2]:=0; A[3]:=0; A[4]:=0; A[5]:=0; baş sany operatoryň zygider yerine ýetirilmegi bilen deň güýçlüdir.

- 2) Iki ölçegli Y massiwi nullamak:

FOR I:=1 TO 15 DO



```
FOR J:=1 TO 20 DO Y[I,J]:=0;
```

3) Massiwiň elementlerini klawiaturadan girizmek:

```
FOR I:=1 TO 15 DO
```

```
FOR J:=1 TO 20 DO READLN(Y[I,J]),
```

bu ýerde iki ölçegli Y massiwiň elementleriniň bahalary girizilýär.

4) Massiwiň elementlerini çapa çykarmak:

Massiwiň elementleriniň bahalaryny çapa çykarmak hem edil ýokardaka meňzeşlikde amala aşyrylýar: diňe READ, READLN operatorlarynyň ornuna WRITE, WRITELN operatorlary ulanylýar. Me-selem:

```
FOR I:=1 TO 5 DO WRITELN(A[I]);
```

ýa-da

```
FOR I:=1 TO 15 DO
```

```
FOR J:=1 TO 20 DO WRITELN(Y[I,J]);
```

Bu ýagdaýda her setirde bir baha çap edilýär. Matrisany setir-me-setir çap etmek üçin, programmany aşakdaky ýaly özgertmeli:

```
FOR I:=1 TO 15 DO
```

```
BEGIN writeln;
```

```
FOR J:=1 TO 20 DO Write(Y[I,J]:5);
```

```
END;
```

5) Käbir ýagdaýlarda massiwiň elementleriniň arasyndan haýsy hem bolsa bir şerti kanagatlandyryanlaryny gözlemeli bolýar.

Mysal üçin, goý $A[N]$ massiwiň elementleriniň arasynda näçe sany otrisatel elementiň bardygyny kesgitlemek gerek bolsun.

Onda programmany aşakdaky ýaly ýazmak mümkin:

```
K:=0;
```

```
FOR I:=1 TO N DO
```

```
IF  $A[I]<0$  THEN  $K:=K+1$ ;
```



gaýtalanmak ýerine ýetirilenden soň, K-üýtgeýän ululyk massiwiň ähli otrisatel elementleriniň sanyna deň bolýar. Bu ýerde K-parametre sanajy diýilýär.

Mysal. A(4X5) gönüburçly matrisa berlen bolsun. Bu matri-sanyň položitel elementleriniň jemini hasaplamagyň programmasyny düzmeli:

```
PROGRAM JEM1;
USES CRT;
CONST IMAX=4; JMAX=5;
TYPE MATR=ARRAY[1..IMAX,1..JMAX] OF REAL;
VAR A : MATR; JEM : REAL;
      I,J : INTEGER;
BEGIN CLRSCR;
      JEM:=0;
      FOR I:=1 TO IMAX DO
        FOR J:=1 TO JMAX DO
          BEGIN
            READLN(A[I,J]);
            IF A[I,J]>0 THEN JEM:=JEM+A[I,J];
          END;
        END;
      WRITE('JEM=',JEM:8:3); END.
```

Funksiýalar we proseduralar

Käbir meseleler çözülide parametrleriň dürli bahalarynda şol bir algoritmi birnäçe gezek gaýtalaý ýerine ýetirmeli bolýar. Beýle ýagdaýda programmanyň şol algoritme degişli bölegini aýratyn bölüp alyp, oňa gerek wagty programmanyň islendik böleginden ýüzlenmek



mümkindir.

Programmanyň özbaşdak programma birligi hökmünde ulanyp bolýan bölegine bölek programma diýilýär. Paskal algoritmik dilinde bölek programmanyň iki görnüşi ulanylýar:

1. Funksiýalar;
2. Proseduralar.

Eger hasaplama ýerine ýetirilenden soň alnan netije diňe bir baha bolsa, onda onuň ýaly hasaplamany funksiýa görnüşinde ýazyp bolýar. Funksiýa hem üç bölekden durýar:

1. Sözbaşy
2. Yglan ediş bölümi
3. Funksiýanyň göwresi.

Funksiýa umumy görnüşde aşakdaky ýaly ýazylyar:

FUNCTION F($q_1:T_1;q_2:T_2; \dots; q_n:T_n$):T; {Funksiýanyň sözbaşy})

<Yglan ediş bölümi>

BEGIN

P1;

P2;

... {Funksiýanyň göwresi}

F:=...

END;

bu ýerde F – funksiýanyň ady; $q_i(i=1, n)$ – formal parametrleriň atlary; $T_i(i=1, n)$ – degişlilikde olaryň tipleri; T – F funksiýanyň adynyň tipi ýa-da başgaça netijäniň tipi; P1, P2, ... - funksiýanyň göwresini emele getirýän operatorlar. Yglan ediş bölümünde diňe funksiýanyň çäginde ulanylýan parametrler we olaryň tipleri görkezilýär. Eger formal parametrleriň birnäçesi şol bir tipe degişli bolsa, onda olary toparlap, tipi diňe bir gezek ýazmaklyga rugsat edilýär. Meselem:

FUNCTION FAKT(K1,K2,K3 : INTEGER) : REAL;



Funksiýa programmanyň islendik ýerinden ýüzlenmek üçin baha bermek operatorynyň sag tarapyndaky aňlatmada onuň adyny we ýaýyň içinde formal parametrleriň iş ýüzündäki bahalaryny görkezmeýerlikdir.

F(b1, b2, ..., bn);

Bu ýerde F – funksiýanyň ady; b1, b2, ..., bn – degişlilikde q1, q2, ..., qn – formal parametrleriniň iş ýüzündäki bahalary. Formal parametrleri bilen onuň iş ýüzündäki bahalarynyň sany deň bolmaly we olar degişlilikde tipleri boýunça gabat gelmeli.

Mysal.

$$C_n^m = \frac{n!}{m!(n-m)!} \{n, m \in \mathbb{N}, n > m\} \quad \text{formula boýunça } n \text{ ele-}$$

mentli köplükdäki m elementli utgaşdyrmalaryň sanyny hasaplamak üçin programma düzmeli. Faktorial hasaplamagy bölek programma – funksiýa görnüşinde ýazmaly.

```
PROGRAM UTGSANY;
```

```
USES CRT;
```

```
VAR CNM : LONGINT;
```

```
  N,M,L : INTEGER;
```

```
  FUNCTION FACT(K : INTEGER) : LONGINT;
```

```
    VAR P,I : LONGINT;
```

```
    BEGIN
```

```
      P:=1;
```

```
      FOR I:=1 TO K DO P:=P*I;
```

```
      FACT:=P
```

```
    END;
```

```
  BEGIN
```



```

WRITE('N,M='); READ(N,M);
      CNM:=FACT(N) div FACT(M) div FACT(N-M);

      WRITE('Utgaşmanyň sany=',CNM);

END.

```

Eger algoritm ýerine ýetirilenden soň alynýan netije birden köp – birnäçe baha bolsa, onda bölek programmanyň prosedura görnüşinden peýdalanylýar. Proseduralar umumy görnüşde aşakdaky ýaly ýazylýar:

PROCEDURA F (formal parametrleriň sanawy); {Proseduranyň sözbaşy}

<Yglan ediş bölümi>

BEGIN

P1;

P2;

... {Operatorlar bölümi}

Pn; END;

Bu ýerde F – proseduranyň ady, ol funksiýanyň adyndan tapawutlanyp hiç hili baha eýe bolmaýar. Şoňa görä-de, onuň üçin tip kesgitlenmeýär. P1, P2, ..., Pn – proseduranyň göwresini emele getirýän operatorlar. Formal parametrleriň sanawynda prosedura ýüzlenmek üçin gerek bolan parametrleriň atlary we olaryň tipleri görkezilýär. Prosedura aşakdaky ýaly ýüzlenilýär:

F(b1, b2, ..., bn);

Bu ýerde b1, b2, ..., bn – formal parametrleriň iş ýüzündäki bahalary. Edil funksiýadaky ýaly, bu ýerde hem formal parametrler bilen onuň iş ýüzündäki bahalarynyň sany gabat gelmeli we olar degişlilikde şol bir tipe degişli bolmalydyr.

Mysallara seredeliň:

1) $\text{Sh } x = (e^x - e^{-x})/2$ formulany ulanyp $z = \text{Sh}^2 a + \text{Sh}(a-b)/\text{Sha} + \text{Sh}(a^2 - b^2)$ anlatmany hasaplamak üçin programma düzmeli.



Programmasy:

```
PROGRAM PROSEDUR;  
  
USES CRT;  
  
VAR A,B, Z,T1, T2, T3:REAL;  
  
    PROCEDURE SH (X: REAL; VAR R: REAL );  
BEGIN  
  
    R:= (EXP(X) – EXP(-X))/2;  
  
END;  
  
BEGIN CLRSCR;  
  
    WRITE('A we B bahasyny giriz ');  
    READ (A, B);  
    SH (A, T1);  
    SH ( A-B,T2);  
    SH(A*A-B*B,T3);  
    Z:=T1*T1+T2/T1+T3;  
  
    WRITE('Z=' ,Z:12:4);      END.
```

Eger prosedurada we baş programmada şol bir atly parametrler ulanylýan bolsalar, onda prosedura ygylan edilende formal parametrleri görkezmek hökman däldir.

2) Goý, tekizlikde N-sany nokat özleriniň gönüburçly koordinatalary bilen berlen bolsun:

$$(X_i, Y_i), X_i > 0, i=1, N$$

Bu nokatlaryň polýar koordinatalaryny kesgitlemegi parametrsiz görnüşde ýazmaly:

$$R = \sqrt{\sqrt{x} + \sqrt{y}};$$



$\text{tg}(\alpha) = y/x;$
 $\alpha = \text{arctg}(y/x).$

Programasy:

```
PROGRAM POLKORD;  
USES CRT;  
VAR X,Y,R,alpha:REAL;  
PROCEDURE POLAR;  
BEGIN  
R:=SQRT(SQR(x)+SQR(y));  
alpha:=ARCTAN(Y/X);  
END;  
BEGIN  
CLRSCR;  
WRITELN('N=');  
READ(N);  
FOR I:=1 TO N DO  
BEGIN  
WRITELN('X,Y bahasyny giriz');  
READ(x,y);  
POLYAR;  
WRITELN('R=',R:8:3,' alpha=', alpha:8:3);  
END;  
END.
```



Funksiyada we prosedurada formal parametrleri islendik tertipde ýerleşdirmek mümkin. Bölek programma ýüzlenilen mahalynda formal parametrler nähili tertipde ýerleşdirilen bolsa, olaryň iş ýüzündäki bahalary hem şol tertipde ýerleşdirilen bolmaly.

Prosedurada formal parametrleriň iki görnüşini biri-birinden tapawutlandyrmak gerek:

1) Baha hökmünde ulanylýan parametrler;

2) Üýtgeýän ululyk hökmünde ulanylýan parametrler;

Baha hökmünde ulanylýan parametrler proseduranyň ýerine ýetirilmegi üçin zerur bolan başlangyç maglumatlary - berlenleri kesgitlemek üçin hyzmat edýärler. Olar formal parametrleriň sanawynda:

$(q1:t1; q2:t2; \dots)$ ýa-da $(q1, q2; \dots)$

görnüşinde ýazylýar. Bu ýerde: $q1, q2, \dots$ - parametrleriň bahalary hemişelik, üýtgeýän ýa-da aňlatma görnüşinde berlip bilner.

Üýtgeýän ululyk hökmünde ulanylýan parametrler adatça prosedura ýerine ýetirilenden soň, alnan netijelere baha bermek üçin hyzmat edýärler. Proseduradan alynýan maglumatlar şol üýtgeýän ululyklar arkaly esasy programma berilýär. Üýtgeýän ululyk hökmünde ulanylýan parametrler formal parametrleriň sanawynda aşakdaky görnüşde ýazylýar:

$(\dots; \text{VAR } q1 : T1, q2 : T2, \dots, qn : Tn),$

bu ýerde $q1, q2, \dots, qn$ – üýtgeýän ululyk hökmünde ulanylýan parametrleriň atlary; $T1, T2, \dots, Tn$ - degişlilikde olaryň tipleri. Paskal algoritmik dilinde bir bölek programmanyň içinde ýene-de şol bölek programmanyň özüne ýüzlenmäge rugsat edilýär. Bölek programma ýüzlenmekligiň bu görnüşine rekursiw ýüzlenme diýilýär. Köplenç matematiki formulalary rekursiýa görnüşde ýazyp bolýar.

Meselem:

$$N! = \begin{cases} 1, & \text{eger } n=0 \text{ bolsa,} \\ N(N-1)!, & \text{eger } n>0 \text{ bolsa.} \end{cases}$$



Mysal.

Rekursiw ýüzlenmäni ulanyp $k!$ -y hasaplamak üçin bölek programma düzmeli.

```
FUNCTION FAKTOR ( K: INTEGER ): INTEGER;
```

```
    BEGIN.
```

```
    IF K=1 THEN FAKTOR:=1 ELSE
```

```
    FAKTOR:= FAKTOR ( K-1 ) *K END;
```

Programmanyň islendik ýerinde $n!$ hasaplamak üçin

```
nfaktor:= fakt (n);
```

operatory ýazmak ýeterlikdir.

Käbir ýagdaýlarda bir bölek programmada beýleki bir entek yglan edilmedik bölek programmany çagyrmaly bolýar. Onuň üçin şol çagyryljak bölek programmanyň sözbaşysynyň yzyndan FORWARD-kömekçi sözünü ýazmaly. Bu ýagdaýy aşakdaky mysalyň üsti bilen düşündirmek mümkin:

```
PROGRAM A5;
```

```
VAR X,Y: real;
```

```
PROCEDURE P1(A:REAL); FORWARD;
```

```
PROCEDURE P2(B:REAL);
```

```
    BEGIN P1(X) END;
```

```
        PROCEDURE P1;
```

```
            BEGIN P2(Y) END;
```

```
        BEGIN P2(X); P1(Y); END.
```

Prosedura, funksiýadan tapawutlanyp, birden köp - birnäçe netijäni gaýtaryp, käbir ýagdaýlarda bolsa, hiç hili netijäni gaýtarman hem biler. Meselem:




```

PROGRAM A6;
USES CRT;
VAR I: BYTE;

PROCEDURE LINIA (N: INTEGER);
VAR      I: INTEGER;
BEGIN
FOR J=1 TO N DO
        WRITE( '-'); END;
BEGIN
FOR I:=1 TO 6 DO
        BEGIN LINIA (I); WRITELN; END
END.

```

Görnüşü ýaly, bu ýerde her gezek bölek programma ýüzlenilende yzyna netijä gaýtarylmaýar-da, diňe käbir operatorlaryň toplумы ýerine ýetirilýär. Programmada WRITELN operatory her gezek täze setire geçmek üçin hyzmat edýär.



Simwollar bilen işlemek

Paskalda simwol (char) we setir (string) tipli üýtgeýänler bilen işlemek mümkindir. Kompýuterde ulanylýan her bir simwol kompýuteriň ýadynda sanamagyň ikilik ulgamyndaky bir bitin san görnüşinde saklanýar. Simwollar bilen olara degişli sanlaryň arabağlanyşygyny görkezýän tablisa kod tablisasy diýilýär. Häzirki wagtda köp sanly kod tablisalary bardyr.

ASCII (American Standard Code for Information Interchange) – munuň özi maglumatlary alyşmagyň standart amerikan kodudyr. Belli bolşy ýaly, kompýuterde ulanylýan ähli simwollaryň öz kodlary bardyr. Ähli baş we setir latyn harplaryň, sanlaryň we klawiaturadaky beýleki dolandyryjy klawişalaryň kodlary 0-127 aralykda ýerleşýändir. Kod tablisalarynyň (sahypalarynyň) birinji bölegi (128 simwoly) ähli kod tablisalarynda birmeňzeşdir. Kod tablisalarynyň galan böleginde milli elipbiýiň harplary we başga simwollar ýerleşdirilýär. MS DOS operasion sistemasy ASCII kodlaryny ulanýar. Windows operasion sistemasynda bolsa esasan ANSI we UniCode kodlary ulanylýar. Diske faýl ýazylanda simwollaryň deregine olaryň kodlary ýazylýar. Diskden faýl okalanda bolsa okalan kodlaryň deregine degişli simwollar ekrana çykýar. Biz kähalatlarda MS DOS-da diske ýazylan rus dilindäki teksti Windows operasion sistemasyndaky programmalar (mysal üçin, Bloknot) bilen okanymyzda düşnüksiz ýazgylary görýäris. Edil şeýle ýagdaý tersine okalanda (Windows-da taýynlanan tekst MS DOS-da okalanda) hem ýüze çykýar. Bu ýagdaý ASCII we ANSI kodlarynyň ikinji bölekleriniň bir-birine doly laýyk gelmeýändigini sebäpli bolýar. Simwollaryň ASCII we ANSI kodlary 1 baýt ýer tutýar. Windows operasion sistemasynda ulanylýan simwollaryň UniCode kodlary üçin 2 baýt ýer berilýär.

Char tipli baha – kompýuteriň elipbiýindäki boş bolmadyk, goşa apostrof belgisiniň içine alnan simwoldyr. Mysal üçin, ' ', 'A', '7'



we ş.m. Simwollaryň bu adaty ýazgysyndan başga Paskal simwollary ýazmak üçin ýene-de iki görnüşi ulanylýar:

1. Simwoly onuň ASCII kodunyň kömegi bilen aňlatmak. Bu ýagdaýda simwolyň kodunyň öňünden # belgisi goýulýar. Mysal üçin,

97 = Chr(97) = 'a' ('a' simwol),

65 = Chr(65) = 'A' ('A' simwol),

32 = Chr(32) = ' ' (boşluk, probel),

0 = Chr(0) (nulunjy simwol).

2. Kodlary 0-dan 31-e çenli bolan dolandyryjy simwollar. Olary belgilemek üçin "^" belgi we latyn elipbiýindäki şol nomerli harpy (1...26 aralykdaky kodlar üçin) ýa-da kömekçi belgi (27...31 aralykdaky kodlar üçin) ulanylýar. Mysal üçin,

^A = #1 = Chr(1) - kod 1,

^B = #2 = Chr(2) - kod 2,

...

^ = #26 = Chr(26) - kod 26,

^[= #27 = Chr(27) - kod 27,

...

^_ = #31 = Chr(31) - kod 31,

şol sanda ^G – ses (kod 7), ^I – TAB (kod 9), ^J – LF (kod 10), ^M – CR (kod 13) we ş.m.

Simwollary diňe deňşdirmek we dakmak (onuň kömegi bilen başga simwola baha bermek) bolýar. Simwollar deňşdirilende olaryň ASCII kodlary deňşdirilýär. Eger simwollaryň kodlary deň bolsa, onda olar deň hasaplanýar, eger bir simwolyň kody beýlekiniňkiden uly bolsa, onda kody uly simwol uly hasaplanýar:

'R' = 'R'



'r' > 'R' (kod 114 > kod 82).

Deñşdirmek amallary <, <=, =, >=, >, <> belgileriň kömegi bilen ýazylýar.

Simwol bahalara we üýtgeýänlere aşakdaky tablisada görkezilen funksiýalar ulanylyp bilner:

Funksiýa : Tipi	Ýerine ýetirýän işi
Chr(X:Byte): Char	ASCII kody X-e deň simwoly berýär
Ord(C:Char): Byte	C simwolyň ASCII koduny berýär
Pred(C:Char):Char	C - niň öňündäki simwoly berýär
Succ(C:Char):Char	C - niň yzyndaky simwoly berýär
UpCase(C:Char):Char	'a'...'z' simwollary degişlilikde 'A'...'Z' simwollara öwürýär

Bellik. UpCase funksiýasy diňe setir latyn harplaryny degişli baş latyn harplaryna öwürýär, galan simwollary öňküsi ýaly goýýar.

Mysal. ASCII kod tablisasyny çap etmeli.

Çözülüşi:

Var

i:byte;

begin

for i:=32 to 255 do write(i,'->',chr (i),' ');

end.

Setirler. Setirler bilen işlemek üçin proseduralar we funksiýalar

Tekstleri gaýtadan işlemek üçin STRING (setir) tipi ulanylýar. Ol öz gurluşy boýunça simwollaryň bir ölçegli *ARRAY[0..N] of CHAR* massiwine meňzeşdir. Setir tipli üýtgeýän ululyklar beýan edilende STRING ýa-da STRING[N] sözler ulanylýar. Bu ýerde N setiriň bo-



lup biljek uzynlygydyr. Eger ol görkezilmedik bolsa $N=255$ diýip hasap edilýär.

Programmada setiriň her bir simwolyna gös-göni ýüzlenip bolýar. Onuň üçin ilki setiriň ady, soňra bolsa dik ýaýlaryň içinde şol simwolyň setirdäki tertip nomeri ýazylyar. Mysal üçin, $s[5]$, $a[23]$, $d[n]$.

Setirdäki simwollaryň tertip nomeri 0-dan başlanýar. 0-njy simwol setiriň uzynlygyny aňladýar. Mysal üçin, $S[0]$ S setiriň uzynlygyny görkezýän simwoldyr. Setiriň uzynlygynda 0-njy simwol hasaba alynmaýar, şonuň üçin setiriň ilkinji simwolyň nomeri 1-e deňdir. Setiriň uzynlygynyň bitin sanda aňladylyşyny almak üçin ord ($S[0]$) funksiýasyny ulanmalydyr. Bu ýerde ord -funksiýasy simwolyň koduny kesgitleýär.

Setirlere baha bermek, birleşdirmek we deňeşdirmek mümkindir. Setirleri birleşdirmek üçin «+» amaly ulanmak bolýar. «+» amaly iki setiri birleşdirýär. Eger goşulanda alnan täze setiriň uzynlygy setiriň bolup biljek uzynlygyndan uly bolsa, onda onuň artyk bölegi zyňylýar.

Setirleri deňeşdirmek setiriň çep tarapky simwolyndan başlap simwollar boýunça amala aşyrylýar.

Setirleriň üstünde işlemek üçin aşakdaky tablisadaky görkezilen funksiýalar we proseduralar ulanylýar:

Proseduralar we funksiýalar	Ýerine ýetirýän işi
1	2
LENGTH(S:String):Byte	S setiriň uzynlygyny tapýar
CONCAT(S1, S2, ..., Sn:String): String	S1,S2, ..., SN -setirleri goşmak
COPY(S:String; Start, Len:Integer):String	S setiriň Start-nji simwolyndan başlap, Len sany simwolyň nusgasyny (kopiýasyny) alýar
Delete(Var S:String; Start, Len:Integer)	S setiriň Start-nji simwolyndan başlap, Len sany simwolyň öçürýär
Insert(Var S:String; SubS:String;Start:Integer)	S setiriň Start-nji simwolyndan başlap şol setire SubS setiri goşýar



1	2
Pos (SubS,S:string):Byte	S setiriň içinde SubS setiriň bardygyny barlaýar. Eger SubS setir S setiriň içinde bar bolsa, onda onuň S setiriň näçenji simwolyndan başlanýandygyny tapýar, eger SubS setir S setiriň içinde ýok bolsa, onda funksiýa 0 bahany çykarýar
Str (X[:F[:n]]; Var S:String)	Hakyky ýa-da bitin tipli X sany setir tipli S üýtgeýän ululyga öwürýär. Bu ýerde: F - sany ýazmak üçin berilýän öýleriň sany, n -sanyň drob bölegine berilýän öýleriň sany. n parametr diňe X hakyky tipli bolanda ýazylýar
Val (S:String; Var X; Var ErrCode:Integer)	Setir tipli S üýtgeýän ululygyň bahasyny san görnüşine öwürüp, X üýtgeýän ululygyň öýüne ýazýar. Bu ýerde: ErrCode Integer tipli üýtgeýän ululyk. Eger öwürmek üstünlikli geçse ErrCode 0-a deň bolýar. Eger öwürmek käbir sebäbe görä amala aşmadyk bolsa, onda ErrCode S setirde öwürmeklige päsgel beren ilkinji simwolyň tertip nomerine deň bolýar. X üýtgeýäniň bahasy bu ýagdaýda üýtgemän galýar.

Bellik. S setiriň başky simwollarynyň boşluk (probel) bolmagy VAL prosedurasyny ýerine ýetirilende ýalňyşlyk döretmeýär, ýöne S setiriň soňky simwollarynyň boşluk bolmagy ýalňyşlyk döredýär. Mysal üçin, Val (' 304', n, k) prosedura dogry ýerine ýetiriler. Netijede n-iň bahasy 304, k-nyň bahasy bolsa 0 bolar, Val ('304 ',n, k) - prosedura ýalňyş bolar. Netijede n-iň bahasy üýtgemez, k-nyň bahasy bolsa 4-e deň bolar.



1-nji mysal. Setir berlen. Onuň içinde näçe sany boşluk belgisi-niň bardygyny tapmaly.

Çözülişi.

Var

S:string;

n, k,i:integer;

begin

write('s=');

readln(s);

n:=length(s);

k:=0;

for i:=1 to n do

if s[i]=' ' then k:=k+1;

write('k=',k);

end.

Köplükler

Köplük diýlip, bir bitewi zat hökmünde seredip bolýan şol bir häsiýet, şol bir nyşan ýa-da nyşanlar boýunça ýygnalan obýektleriň toplumyna düşünilýär. Meselem, 1-den 99-a çenli zygider gelýän ähli jübüt sanlaryň köplügi, türkmen diliniň elipbiýine girýän ähli çekimli harplaryň köplügi we ş.m.

Paskal dilinde köplügiň elementlerine birnäçe çäklendirmeler goýulýar. Meselem, köplügiň elementleri REAL tipe degişli bolmaly däl, köplügiň elementleriniň sany 256-dan geçmeli däl we ş.m.

Köplügiň elementleri dik ýaýlaryň içinde ýazylýar.

Meselem:

['A', 'B', 'C']

elementleri CHAR tipe degişli bolan köplük.



[-3, 1, 3, 5]

elementleri INTEGER tipe degişli bolan köplük.

[FALSE, TRUE]

elementleri BOOLEAN tipe degişli bolan köplük.

[1..100]

elementleri çäklendirilen tipe degişli bolan köplük we ş.m. Hiç bir elementi bolmadyk köplüğe boş köplük diýilýär we ol PASKAL-da [] görnüşde bellenilýär.

Boş köplük islendik köplügiň bölek köplügidir. Baha hökmünde diňe köplükleri kabul edip bilýän üýtgeýän ululyklar aşakdaky ýaly yglan edilýär:

TYPE

<tipiň ady>=SET OF <elementleriň tipi>;

VAR

<üýtgeýän ululygyň ady>: < tipiň ady>;

ýa-da başgaça, gönüden-göni üýtgeýän ululyklar bölümünde hem yglan etmek mümkin:

VAR

<üýtgeýän ululygyň ady>: SET OF <elementleriň tipi>;

Meselem:

VAR X:SET OF 1..3;

ýazgy X üýtgeýän ululygyň elementleri diňe 1..3 aralykdan bolan köplük bolup bilýändigini aňladýar. Bu ýerde: X üýtgeýän ululyk {1,2,3} köplügiň islendik bölek köplüğine eýe bolup bilýär. Ýagny X üýtgeýän ululyk aşakdaky köplükde kesgitlenen:

{{1,2,3}, {1,2}, {1,3}, {2,3}, {1}, {2}, {3}, {}}

Köplük yglan edilende elementleriň tipi hökmünde BYTE, BOOLEAN, CHAR tipleri we standart däl skalýar tipleri ulanyp bolýar.

VAR

A,B:SET OF 0..49;



HARP:SET OF CHAR;

X:SET OF (MART, APREL, MAY);

[] boş köplük tipleriň islendigine degişli edip bolýan ýeke-täk köplükdir.

Eger köplük [e1..e2] görnüşde berlip, $e1 > e2$ bolsa, onda ol köplük boş köplük hasap edilýär. Meselem: [35..-1] ýa-da ['m'..'s'] boş köplüklerdir.

Edil matematikadaky ýaly, Paskal algoritmik dilinde hem köplükleriň üstünde kesişme, birleşme, tapawut alamatlaryny ýerine ýetirip bolýar. Bu amallar degişlilikde '*', '+', we '-' simwollar bilen belgilenýär. Ýagny,

VAR

A,B,X,Y,Z : SET OF INTEGER;

BEGIN

X:=A*B; Y:=A+B; Z:=A-B;

operatorlary ýerine ýetenlerinde X üýtgeýän ululyk şol bir wagtda A we B köplükleriň ikisinde hem bar bolan elementlerden düzülen köplüğe deň bolar, Y-üýtgeýän ululyk A we B köplükleriň iň bolmanda birine degişli bolan elementlerden durýan köplüğe deň bolýar, Z-üýtgeýän ululyk bolsa, A köplügiň B köplükde bar bolmadyk elementlerinden durýan köplüğe deň bolýar. Bu ýerde A we B şol bir tipe degişli bolan köplükler bolmaly.

Köplügiň üstünden bu amallardan başga-da, baş sany logiki amaly hem ýerine ýetirmek mümkin:

1) '='; 2) '<>'; 3) '<='; 4) '>='; 5) 'IN';

Bu belgiler degişlilikde "=", "≠", "≤", "≥", "∈" belgileri aňladýarlar.

Goý, A we B – şol bir tipe degişli bolan köplükler bolsun. Ol köplükler şol bir elementlerden duran bolsa, onda $A=B$ logiki aňlatma «TRUE» baha eýe bolýar, beýleki ýagdaýlarda «FALSE» baha eýedir.

Eger A we B köplükler özara gabat gelmeseler, onda $A <> B$ - logiki aňlatma «TRUE» baha eýe bolýar, beýleki ýagdaýlarda «FALSE» baha eýe bolýar.



Eger B köplük özünde A köplügi saklaýan bolsa, onda $A \leq B$ – logiki aňlatma «TRUE» baha eýe, beýleki ýagdaýlarda «FALSE» baha eýe.

Eger A köplük özünde B köplügi saklaýan bolsa, onda $A \geq B$ - logiki aňlatma «TRUE» baha eýe, beýleki ýagdaýlarda «FALSE» baha eýe.

Eger X element S köplüğe degişli bolsa, onda $X \in S$ – logiki aňlatma «TRUE» bahany kabul edýär, eger-de X element S köplükde ýok bolsa, onda «FALSE» bahany kabul edýär.

Mysal üçin, goý,

```
VAR A,B,C : SET OF CHAR;
```

```
BEGIN
```

```
  A:=[ 'a', 'b', 'c', 'd'];
```

```
  B:=[ 'b']; C:=[ 'c', 'e']; ...
```

bolsun. Onda $A+C$, $A-C$ we $A*B$ aňlatmalar degişlilikde $['a'..'e']$, $['a', 'c', 'd']$ we $['b']$ bahalara eýe bolýar; $B \leq A$, $A \geq B$, $C \leq A$ logiki aňlatmalar degişlilikde «TRUE», «TRUE» we «FALSE» bahalara eýe bolýarlar.

$'a' \in A$ we $'a' \in C$ logiki aňlatmalar degişlilikde «TRUE» we «FALSE» bahalary kabul ederler.

IN operasiýasy haýsy bolsa-da bir bahanyň berlen köplüğe degişlidigini ýa-da degişli däldigini barlamak üçin hyzmat edýär we köplenç şertli operatorlarda ulanylýar. Meselem:

```
IF 2 IN [1,2,3] THEN ...,
```

```
IF 'V' IN['a',..., 'e'] THEN ...,
```

```
IF X1 IN [X0,X1,X2,X3] THEN ...
```

Kähalatlarda IN operasiýasy NOT inkär etme operasiýasy bilen bilelikde ulanylýar. Meselem:

```
IF NOT(X IN M) THEN ...
```

Eger X element M köplükde ýok bolsa, onda bu logiki aňlatma «TRUE» bahany alýar.



Mysal.

```
PROGRAM MYSAL_1;

USES CRT;

VAR

    K,I : BYTE; TEKST : STRING[255];
    LATHARPY : SET OF CHAR; SIMWOL : CHAR;
BEGIN
CLRSCR;

    LATHARPY:=['a'..'z']; k:=0;

    READLN(TEKST);

    FOR I:=1 TO LENGTH(TEKST) DO

    BEGIN

        SIMWOL:=TEKST[I];
        IF SIMWOL IN LATHARPY THEN K:=K+1;

    END;

    WRITE('K=',K);

END.
```

Bu programma klaviaturadan girizilen TEKST atly setir üýtgeýän ululygyň bahasynda näçe sany kiçi latyn harpynyň bardygyny kesgitleýär.

Mysal.

```
PROGRAM MYSAL_2;

USES CRT;

VAR

    ELEMENT : 1...25; I,K : INTEGER;
```



```

        KOPLUK : SET OF 1..25;
BEGIN
CLRSCR;

    KOPLUK:=[ ]; RANDOMIZE;

    FOR I:=1 TO 8 DO BEGIN ELEMENT:=RANDOM(24)+1;

        KOPLUK:=KOPLUK+[ELEMENT]

    END;

    K:=0;

    FOR I:=1 TO 25 DO

        IF I IN KOPLUK THEN

            BEGIN WRITE(I, ' ':5); K:=K+1; END;

        WRITE('K=',K);

    END.

```

Netijeler:

a) 1 4 13 14 18 19 20 22

k=8

b) 7 8 9 10 11 14 23

k=7

ç) 10 11 17 20 23 24

k=6

we ş. m.

Bu programmada 1..25 aralykdan bolan tötänleýin sanlardan köplük düzülýär. Köplügiň elementleri we olaryň sany çapa çykarylýar. Bu ýerde tötänleýin sanlaryň arasynda gabat gelýänleri-de bolmagy mümkin; bu ýagdaýda $K < 8$ bolýar.

Bellik.

1) RANDOM(N) – standart funksiýasy $[0, N-1]$ aralykdan alnan



tötänleýin bitin sany kesgitleýär. Eger bu ýerde N parametr goýulmadyk bolsa, onda [0,1] aralykdaky ötänleýin hakyky sanlar alynýar;

2) Paskal algoritmik dilinde köplügiň elementi hökmünde otrisatel bitin sanlary ulanmak bolmaýar.

Paskal algoritmik dilinde ýazgylar (kombinirlenen tip)

Belli bolşy ýaly, massiw diýlende şol bir tipden bolan üýtgeýän ululyklaryň toplumyna düşünilýär. Ýazgy – bu massiwden tapawutlanyp, dürli tipden bolan üýtgeýän ululyklaryň toplumydyr. Şoňa görä-de, oňa başgaça kombinirlenen - garyşyk tip hem diýilýär. Ýazgylar tipler hökmünde aşakdaky ýaly yglan edilýär:

TYPE

P=RECORD A1:T1; A2:T2;...; An:Tn END;

Bu ýerde: P – ýazgynyň ady, A1, A2,..., An – ýazgyny düzýän meýdançalaryň atlary, T1,T2,...,Tn bolsa degişlilikde A1,A2,...,An meýdançalaryň tipleri.

Meselem, berlen topardaky talyplaryň doglan günlerini kombinirlenen tipiň üsti bilen şeýle kesgitlemek mümkin:

TYPE

DATE=RECORD

DOGLAY : 1..12;

DOGLGUN : 1..31;

DOGLYL : INTEGER;

END;

VAR

X : DATE;

Görnüşi ýaly, bu ýerde DATE atly ýazgy DOGLAY, DOGLGUN, DOGLYL diýlip atlandyrylýan üç sany meýdançadan ybarat bolup, olar degişlilikde 1..12, 1..31 çäklendirilen tiplere we INTEGER tipe degişlidirler.



X : DATE;

ýazgy X – üýtgeýän ululyga topardaky islendik talybyň doglan güni baradaky maglumatlary (doglan güni, aýy we ýyly) ýazyp bolýandygyny aňladýar. Mysal üçin, X üýtgeýän ululyga 1991-nji ýylyň 27-nji oktýabry diýen bahany bermek üçin:

X.DOGLAY:=10;

X.DOGLGN:=27;

X.DOGLYL:=1991;

baha bermek operatoryny ýerine ýetirmeli. Ýene-de mysala ýüzleneliň:

TYPE AWTOMOBIL = RECORD

NOMER : INTEGER;

MARKA : STRING[25];

FAAULAN : STRING[50]; {Ulanýanyň F.A.A.}

OYADRESULAN : STRING[50]; {Ulanýanyň
ýaşayan öý salgysy}

END;

VAR

X,Y : AWTOMOBIL;

Bu ýazgy X we Y üýtgeýän ululyklara diňe AWTOMOBIL atly tipe degişli bolan bahalary berip bolýandygyny aňladýar. Bu tipe degişli bolan her baha 4 sany komponentden ybarat bolup, olaryň birinjisi INTEGER tipe degişli, beýleki üçüsi bolsa STRING tipe degişli bolmaly. Ýazga girýän komponentler degişlilikde NOMER, MARKA, FAAULAN we OYADRESULANAN diýlip atlandyrylýar.

Z : AWTOMOBIL üýtgeýän ululyga baha bermek üçin Z.NOMER, Z.MARKA, Z.FAAULAN we Z.OYADRESULAN atly üýtgeýän ululyklara baha bermeli. Meselem:

Z.NOMER: =7462;

Z.MARKA: =‘WAZ-2107’;

Z.FAAULAN:=‘MEREDOW’;

Z.OYADRESULAN:=‘Aşgabat köçesi, 6 jaýy’;



Bu üýtgeýän ululyklaryň toplumyny 4 elementli bir ölçegli massiw bilen deňşdirmek mümkin. Eger Z massiw bolan bolsa, onda onuň komponentlerine dik ýaýyň içinde ýazylan indeks boýunça ýüzlenilerdi. Bu ýerde bolsa, komponente ýüzlenmek üçin Z-niň zyzyndan «.»-nokat belgisi goýulýar we meýdançanyň ady görkezilýär.

Z.NOMER, Z.MARKA, we ş.m. görnüşli atlara düzme atlar diýilýär. Olara baha bermek üçin giriş-çykyş operatorlaryndan peýdalanyp bolýar. Meselem:

```
READ(Z.NOMER,Z.MARKA,Z.FAAULAN,
```

```
Z.OYADRESULAN);
```

```
  ýa-da
```

```
WRITE(Z.NOMER,Z.MARKA,Z.FAAULAN,
```

```
Z.OYADRESULAN);
```

Şol bir tipden bolan X we Y ýazgylar üçin

```
X:=Y;
```

baha bermek operatoryny ulanyp bolýar. Beýle ýagdaýda Y ýazgynyň komponentleriniň bahalary deňşlilikde X ýazgynyň komponentlerine eýe bolýar.

Käbir ýagdaýlarda ýazgylardan massiw emele getirmek amatly bolýar, ýagny massiwiň her elementine kombinirlenen tipe deňişli baha berilýär. Meselem:

```
TYPE
```

```
  MAGLUMAT = RECORD
```

```
    FAA:=STRING[25];
```

```
    DY:=INTEGER;
```

```
    OA:=STRING[60];
```

```
  END;
```

```
  VEKTOR_MAGLUMAT=ARRAY[1..25] OF MAGLUMAT;
```

```
VAR
```

```
  MITDAU1 : VEKTOR_MAGLUMAT;
```



Bu ýazgy MITDAU1 – üýtgeýän ululygyň 25 elementli bir ölçepli massiwdigini we onuň her bir komponentine talybyň familiýasy, ady, atasynyň ady, doglan ýyly we onuň öýüniň salgysy baradaky maglumatlary berip bolýandygyny aňladýar. Mysal üçin:

```
MITDAU1 [1].FAA:=‘Aşyrow D.M.’;
```

```
MITDAU1 [1].DY:=1984;
```

```
MITDAU1 [1].OA:=‘Aşgabat ş. Magtymguly şaýoly 15/2’;
```

```
MITDAU1 [2].FAA:=‘Rejepowa M.O.’;
```

```
MITDAU1 [2].DY:=1984;
```

```
MITDAU1 [2].OA:=‘Aşgabat ş. B.Annanow köç. 5-jaýy’.
```

Görnüşi ýaly, düzme atlardan peýdalanmaklyk programmanyň ýazgysynyň ulalmagyna getirýär. Şeýle ýagdaýlarda programmanyň göwrümünü kiçeltmek üçin WITH operatory ulanylýar. WITH operatory aşakdaky ýaly ýazylýar:

```
WITH <kombinirlenen tipli üýtgeýän ululyk> DO <operator>;
```

WITH operatory düzmelil atlarda üýtgeýän ululygyň adynyň «.»-nokat belgisinden öňdäki bölegini taşlap ýazmaga mümkinçilik berýär. Meselem, ýokardaky baha bermek operatorynyň toplumyny şeýle ýazmak mümkin:

```
WITH MITDAU1 [1] DO
```

```
BEGIN
```

```
    FAA:=‘Aşyrow D.M.’;
```

```
    DY:=1984;
```

```
    OA:=‘Aşgabat ş. Magtymguly şaýoly 15/2’;
```

```
END;
```

ýa-da:

```
WITH MITDAU1 [2] DO
```



BEGIN

FAA:=‘Rejepowa M.O.’;

DY:=1984;

OA:=‘ Aşgabat ş. B.Annanow köç. 5-jaýy’;

END.

Paskal algoritmik dilinde özünde birnäçe warianty saklaýan ýazgylary girizmek hem göz önünde tutulandyr. Beýle ýazgylara wariantly ýazgylar diýilýär we ol umumy görnüşde aşakdaky ýaly ýazylýar:

TYPE

YAZGY=RECORD

[r1:t1;r2:t2;...;m:tn;]

CASE C:Tc OF

C1 : (R1 : T1);

C2 : (R2 : T2);

...

CM : (RM : TM);

END,

bu ýerde YAZGY – tipiň ady; r1,r2,...,rn, R1,R2,..., Rm – meýdançalaryň atlary; t1,t2,...,tn, T1,T2,..., Tm – degişlilikde olaryň tipleri; C – saýlaýjy parametr, Tc – saýlaýjy parametriň tipi; C1,C2,...,Cm – belgiler, olar Tc – tipe degişli bolan hemişelikler bolmaly. Ýazgylaryň dik ýaýlara alnan bölegine onuň fiksirlenen bölegi diýilýär. Fiksirlenen bölegiň daşyndaky dik ýaýlar, onuň hökmany dälidigini aňladýar. Ýazgynyň CASE sözünden başlanýan bölegine onuň wariantly bölegi diýilýär.

Ýazgynyň fiksirlenen böleginiň ulanylyşy edil adatdaky ýalydyr. Wariantly bölek ulanylanda ilki C-saýlaýjy parametriň bahasy hasaplanylýar. Soňra onuň kabul eden bahasyna baglylykda C1,C2,...,Cm-wariantlaryň biri saýlanyp alynýar. Eger C=C1 bolsa,



onda $R1:T1$, $C=C2$ bolsa, onda $R2:T2$ we ş.m. eger $C=Cm$ bolsa, onda $Rm:Tm$ alynýar. Diýmek, YAZGY diýen ýazgy $r1:t1$, $r2:t2, \dots$, $r_n:t_n$ -n sany komponentden we C -parametriň kömegi bilen saýlanyp alynýan ýene-de bir sany komponentden ybarat eken.

Bellik. Bu ýerde CASE bölümi üçin aýratyn end sözi ulanylmaýar.

Ýazgy doly kesgitlenenden soň, diňe bir sany END sözi goýulýar. Eger haýsy-da bolsa bir C_k ($1 \leq k \leq m$) hemişelige degişli bolan meýdança boş bolsa, ýagny şol ýagdaý üçin hiç hili meýdança kesgitlenmedik bolsa, onda C_n : (); görnüşde ýazylyar. Wariantly ýazgylar birnäçe meňzeş ýazgylary bir ýazga birleşdirmäge mümkinçilik berýär.

Mysal.

TYPE

HAT=RECORD

NOMER : BYTE;

RAZMER : INTEGER;

CASE P : BOOLEAN OF

TRUE : (BAHA1 : INTEGER);

FALSE : (BAHA2 : REAL)

END;

VAR

X : HAT,

bu ýerde: X –üýtgeýän ululyk özüne baha hökmünde HAT tipli ýazgylary kabul edip bilýär. HAT – tipe degişli bolan ýazgylaryň her biri 3 sany komponentden ybarat bolup, olaryň ikisi fiksirlenen – NOMER : BYTE; RAZMER : INTEGER. Üçünji komponentiň haýsy boljakdygy P-logiki aňlatmanyň alýan bahasyna bagly: Eger P - «TRUE» baha eýe bolsa, onda üçünji komponent hökmünde BAHA1: INTEGER; alynýar; eger P «FALSE» baha eýe bolsa, onda BAHA2: real; alynýar.



Programmireleme dilinde faýl düşünjesi

Maglumatlar adatyňda faýllarda saklanylýar. Mysal üçin, biz Microsoft Word redaktorynda tekst ýazsak maglumaty diskde faýl görnüşinde saklaýarys. Aýdym-sazlaryň hem faýl görnüşde saklanýanlygyny bilýäris. Faýl näme? – diýen sorag ýüze çykýar. Faýl – munuň özi diskiň atlandyrylan bölegidir.

Faýllaryň aýratyn bir görnüşine bukja diýilýär. Bukjalar faýllary, şeýle hem beýleki bukjalary özünde saklap bilýärler. Faýlyň ady we salgysy bolýar. Faýlyň salgysy üç bölekden ybarat bolup durýar.

Faýlyň salgysynyň birinji bölegi kompýuterde bar bolan logiki diskiň biriniň adyny saklaýar. Logiki diskiň ady bolsa biziň bilşimiz ýaly, latyn harplarynyň birisi bilen belgilenýär. Logiki diskiň adyny faýlyň salgysynyň beýleki böleklerinden bölmek üçin iki nokat «:» we «\» simwoly peýdalanylýar.

<Logiki diskiň ady>:\

Mysal üçin:

C:\ Faýl C diskde ýerleşýär.

A:\ Faýl A diskde ýerleşýär.

X:\ Faýl X diskde ýerleşýär.

Faýlyň salgysynyň ikinji bölegi faýlyň ýerleşýän bukjalarynyň zygydirliligini görkezýär: <Bukja1\...\Bukja n>\ .

Faýlyň salgysynyň üçünji bölegi faýlyň doly adyny saklaýar. Faýlyň doly ady iki bölekden ybarat bolup durýar: faýlyň ady hem-de faýlyň tipi (giňeltmesi). Faýlyň ady bilen giňeltmesiniň arasyn-da «.» nokat belgisi goýulýar. Faýlyň adynyň bolmagy hökmandyr, giňeltmesi bolsa bolman hem biler. Umumy görnüşde faýlyň ady aşakdaky ýaly bolar:

<Faýlyň ady>.<tipi>



Faýlyň salgysy doly görnüşde aşakdaky ýaly bolar:

C:\MYPROGS\P01. PAS – bu faýl C diksdäki MYPROGS bukjada ýerleşýär. Faýlyň ady P01 tipi PAS.

D:\SAVES\PROGS\synag.exe – bu faýl D diskiň SAVES bukjasynda saklanýan PROGS bukjada ýerleşýär. Faýlyň ady synag, tipi bolsa exe.

X:\start.bat – bu faýl X diskiň düyp bukjasynda ýerleşýär. Faýlyň ady start, tipi bolsa bat.

Kompýuterde işleýän adam islendik wagtda haýsy hem bolsa bir bukja bilen iş salyşýar. Şol iş salşylyan bukja işçi bukja diýilýär. Eger faýlyň diňe ady görkezilen bolsa, onda ol faýl işçi bukjada ýerleşýändir.

Faýlyň tipi (giňeldilmesi) barada

Ýokarda belläp geçişimiz ýaly, faýlyň tipi (giňeltmesi) faýlyň hökman däl bölegi bolup durýar. Ýöne faýlyň giňeltmesi ol faýl barada möhüm maglumaty özünde saklaýar. Mysal üçin, **txt** giňeltme – bu faýlyň adaty tekst maglumatyny saklaýanlygyny, **exe** giňeltme – bu faýlda maşyn dilindäki programmanyň saklanýanlygyny, **bmp** giňeltme – bu faýlyň şekili saklaýanlygyny, **mp3** giňeltme – bu faýlyň aýdym-sazy saklaýanlygyny we ş.m. aňladýar.

BASIC dilindäki programma **bas** giňeltmeli faýllarda saklanylýar. C++ dilindäki programma **cpp** giňeltmeli faýllarda saklanylýar. Paskal dilindäki programma bolsa pas giňeltmeli faýllarda saklanýar. Paskal dilindäki programma kompilirlenende maşyn diline geçirilýär. Mysal üçin, **salam.pas** faýly özünde Paskal dilindäki programmany saklaýar, kompilirlenenden soň bolsa **salam.exe** faýly emele gelýär. **exe** giňeltmeli faýllar maşyn dilindäki programmalary saklaýarlar. Maşyn dilindäki faýllara ýerine ýetýän faýllar hem diýilýär.



Fiziki we logiki faýllar. **Faýl üýtgeýän ululygy bilen diskdäki faýly bellemek** **(baglanyşdyrmak)**

Haçan-da programmada haýsydyr bir faýl bilen işlenmekçi bolsa, bu faýl üýtgeýän ululyk diýlip atlandyrylýan ululygyň kömegi bilen amala aşyrylýar. Faýl bilen ýerine ýetirilýän işleriň hemmesi soňra bu ululygyň kömegi bilen baglanyşykly ýerine ýetirilýär.

Faýl üýtgeýän ululygyny kesgitlemek ulanylmakçy bolýan faýlyň tipine bagly bolup durýar.

PASKAL dilinde faýllaryň üç görnüşi bilen işlemek mümkinçiligi bar.

- **Text** tipi;
- **File** tipi;
- **File of** <elementleriniň tipi>

Tekst faýllary bilen işlemek üçin **Text** tipi ulanylýar. Tipleşdirilmedik ikilik faýllar bilen işlemek üçin **File** tipi ulanylýar. Tipleşdirilen ikilik faýllar bilen işlenilende bolsa, ol faýlyň elementleriniň tipi anyk görkezilýär. Mysal üçin: **File of integer**.

Faýl üýtgeýän ululygy programmanyň üýtgeýän ululyklar bölümünde beýan edilýär.

Var

T1 : Text; { Tekst faýllary bilen işlemek üçin T1 faýl üýtgeýän ululygy beýan edilýär }

B1 : File; { Tipleşdirilmedik ikilik faýllary bilen işlemek üçin B1 faýl üýtgeýän ululygy beýan edilýär }

B2 : File of integer;

{ Elementleri **Integer** tipine degişli bolan tipleşdirilen ikilik faýllary bilen işlemek üçin B2 faýl üýtgeýän ululygy beýan edilýär }

Paskal programmirleme dilinde standart ASSIGN prosedurasynyň kömegi arkaly fiziki faýl logiki faýl bilen baglanyşdyrylýar.



ASSIGN(<f.ü.u.>, <faýlyň ady>),

bu ýerde <f.ü.u.> – faýl üýtgeýän ululygy. Faýl üýtgeýän ululygy ulanylmazyndan öň beýan edilýär. Faýl üýtgeýän ululygyna logiki faýl hem diýilýär. <faýlyň ady> – faýlyň adyny özünde saklaýan tekst aňlatmadyr, ýagny setir ululykdyr. <faýlyň ady>-na fiziki faýl hem diýilýär. Mysal üçin:

ASSIGN(F, 'maglumat.txt'),

bu ýerde '**maglumat.txt**' fiziki faýl, **F** bolsa logiki faýl. Logiki faýla faýl üýtgeýän ululygy hem diýilýär.

Faýldan maglumat okamak üçin, faýla maglumat ýazmak üçin hem-de faýla maglumat goşmak üçin faýl açylyp bilner.

Faýly okamak maksady bilen açmak. RESET komandasy faýly okamak maksady bilen açýar. Umumy görnüşde ol aşakdaky ýaly ýazylýar: RESET(LF). Bu ýerde: LF logiki faýlyň ady.

Faýly ýazmak maksady bilen açmak. REWRITE komandasy faýly ýazmak maksady bilen açýar. Umumy görnüşde ol aşakdaky ýaly ýazylýar: REWRITE(LF). Bu ýerde: LF logiki faýlyň ady.

Faýla maglumat goşmak maksady bilen açmak. APPEND komandasy faýlyň soňuna maglumat goşmak maksady bilen açýar. Umumy görnüşde ol aşakdaky ýaly ýazylýar APPEND(LF). Bu ýerde, LF logiki faýlyň ady. APPEND operatory diňe tekst faýllary üçin ulanylýar.

Faýldan okamak ýa-da ýazmak. Faýly ýapmak

Faýllary açmak we faýllary ýapmak hemme tipli faýllarda birmeňzeş amala aşyrylýar. Faýllardan maglumatlar okalanda we faýllara maglumatlar ýazylanda bolsa faýllaryň tiplerine baglylykda biraz tapawutlanýarlar.

Faýldan okamak READ(F,X) operatoryň kömegi bilen amala aşyrylýar. Bu ýerde: F faýl üýtgeýän ululygy, X bolsa okalýan ululyk.

Faýla ýazmak WRITE(F,X) operatoryň kömegi bilen amala aşyrylýar. Bu ýerde: F faýl üýtgeýän ululygy, X bolsa faýla ýazylyan ululyk.



Faýly ýapmak CLOSE(F) operatoryň kömegi bilen amala aşyrylýar. Bu ýerde: F faýl üýtgeýän ululygy.

Tekst faýllary

Tekst faýllaryndan maglumatlary okamaklyk **Read**, **Readln** operatorlarynyň kömegi bilen amala aşyrylýar. **Readln** operatory doly setiri okamaklygy amala aşyrýar. **Write**, **Writeln** operatorlary bolsa tekst faýllaryna maglumatlary ýazmaklygy amala aşyrýar. **Writeln** operatory doly setiri faýla ýazmaklyga mümkinçilik berýär. Bu operatorlaryň ilkinji parametri işlenilýän logiki faýlyň ady bolýar.

Mysal. Işçi katalogda **text1.txt** faýlyny döredýän we ol faýla «Salam PASCAL!» setiri ýazmaklyga mümkinçilik berýän programma ýazmaly.

```
Uses crt;
Var t : text;
begin
    Assign(t, 'text1.txt');
    rewrite(t);
    writeln(t, 'Salam PASCAL!');
    close(t);           { Faýl ýapylýar }
end.
```

Mysal. Ýokardaky programmanyň döreden faýlyndaky setiri ekrana çykarmaga mümkinçilik berýän programma ýazmaly.

```
Uses crt;
Var f1 : text;
    s : string;
begin
    Assign(f1, 'text1.txt');
    reset(f1);
    readln(f1,s);
```



```
writeln(s);
close(f1);
readkey;
end.
```

Mysal. input.txt tekst faýlynda aralary boşluk bilen bölünen iki sany bitin san berlipdir. Ol sanlaryň jemini tapmak üçin programma ýazmaly.

```
Uses crt;
Var f : text;
    a,b,s : integer;
begin
    Assign(f, 'input.txt');
    reset(f);
    read(f,a,b);
    s:=a+b;
    write('Jemi = ',s);
    close(f);
readkey;
end.
```

Ikilik tipleşdirilen faýllar

Ikilik faýllar gurluşy boýunça maglumatlary kompýuteriň operatiw (işjeň) ýadyndaky görnüşde saklaýarlar. Ikilik tipleşdirilen faýllardan maglumatlary okamak we ýazmak degişlilikde Read we Write operatorlarynyň kömegi bilen amala aşyrylýar. Bu ýerde Readln we Writeln operatorlary ulanylmaýar, onuň hem sebäbi ikilik faýllarda setir diýen düşünje ýok.

Mysal. Klawiaturadan girizilen Integer tipli 10 sany ikilik faýla ýazmak üçin programma düzmeli.

```
Uses crt;
```




```

Var f : File of Integer;
    i,a : integer;
begin
    Assign(f,'sanlar.bin');
    Rewrite(f);
    For i:=1 to 10 do
    begin
        Write(i,' - nji sany giriz: ');
        Read(a);
        Write(f,a);
    end;

    Close(f);
end.

```

Mysal. Ýokardaky döreden sanlar.bin faýlyndaky 10 sany okamak we ekrana çykarmak üçin programma düzüň.

```

Uses crt;
Var f : File of Integer;
    i,a : integer;
begin
    Assign(f,'sanlar.bin');
    Reset(f);
    For i:=1 to 10 do
    begin
        Read(f,a);
        Writeln(i,' - nji san: ',a);
    end;

    Close(f);
Readkey;
end.

```



**Faýllar hem-de bukjalar bilen işlemek üçin niýetlenen standart funksiýalar we proseduralar barada gysgaça maglumatlar
(F – faýl üýtgeýän ululygy)**

Funksiýa ýa-da prosedura	Ýerine ýetirýän işi
ASSIGN(F, 'faýlyň ady')	Faýl üýtgeýän ululygy bilen faýly baglanyşdyrmak.
REWRITE(F)	Faýly döretmek ýa-da täze faýl açmak.
RESET(F)	Bar bolan faýly açmak.
APPEND(F)	Faýlyň soňuna maglumat goşmak üçin faýly açmak.
EOF(F)	Logiki funksiýa, faýldan iň soňky komponent okalanda TRUE logiki ululygy, galan ýagdaýlarda bolsa FALSE logiki ululygy gaýtaryp bermek.
EOLN(F)	Logiki funksiýa, tekst faýllarda setiriň soňudygyny ýa-da dældigini görkezmek.
ERASE(F)	Faýly diskden öçürmek.
READ(F,X)	Faýldan X-iň bahasyny okamak.
WRITE(F,X)	Faýla X-iň bahasyny ýazmak.
FILESIZE(F)	Faýldaky komponentleriň sany barada maglumat bermek.
SEEK(F,N)	Faýlyň görkezijisini N-nji komponente geçirmek.
FILEPOS(F)	Faýlyň görkezijisiniň duran ornuny kesgitlemek.
CLOSE(F)	Faýly ýapmak.



Fayllar bilen işlemeklige degişli dürli mysallar

1-nji mysal. Natural n san berlipdir. «netije.txt» tekst faýlyny döretmek we faýla aralaryna boşluk (probel) bilen bölmek arkaly 1-den n -e çenli natural sanlary şol faýla ýazmak üçin programma düzmeli.

```
Uses crt;
var f : text;
    n,i : word;
begin
    Assign(f,'netije.txt');
    rewrite(f);
    write('n='); read(n);
    for i:=1 to n do
        write(f,i,' ');
    close(f);
end.
```

2-nji mysal. 'maglumat.txt' tekst faýlynda aralary boşluk bilen bölünen bitin sanlar berlipdir. Ol sanlary ekrana çykarmak üçin programma ýazmaly.

```
Uses crt;
var f : text;
    a : integer;
begin
    Assign(f,'maglumat.txt');

    reset(f);

    while not eof(f) do

        begin
```



```

        read(f,a);
        write(a, ' ');
    end;

    close(f);
readkey;
end.

```

3-nji mysal. 'maglumat.txt' tekst faýlynda aralary boşluk bilen bölünen bitin sanlar berlipdir. Ol sanlaryň jemini tapmak üçin programma ýazmaly.

```

Uses crt;
var f: text;
    a,s: integer;
begin
    Assign(f, 'maglumat.txt');
    reset(f);
    s:=0;
    while not eof(f) do
        begin
            read(f,a);
            s:=s+a;
        end;
    write('jemi = ',s);
    close(f);
readkey;
end.

```



4-nji mysal. 'maglumat.txt' tekst faýlynda aralary boşluk bilen bölünen bitin sanlar berlipdir. Ol sanlaryň orta arifmetiki bahasyny tapmak üçin programma ýazmaly.

```
Uses crt;
var f: text;
    a,k: integer;
s:real;
begin
    Assign(f,'maglumat.txt');
    reset(f);
    s:=0;
    k:=0;
    while not eof(f) do
    begin
        read(f,a);
        s:=s+a;
        k:=k+1;
    end;
    write('jemi = ',s/k:5:3);
    close(f);
readkey;
end.
```

5-nji mysal. 'sanlar.txt' tekst faýlynda aralary boşluk bilen bölünen hakyky sanlar berlipdir. Ol sanlaryň iň kiçisini hem-de iň ulusyny tapmak üçin programma ýazmaly.

```
Uses crt;
var f: text;
    min, max,a: real;
```



```

begin
  Assign (f,'sanlar.txt');
  reset (f);
  read (f,a);
min:=a; max:=a;;
  while not eof(f) do
    begin
      read(f,a);
      if a<min then min:=a;
      if a>max then max:=a;

    end;
  write ('min = ',min:5:2,', max = ',max:5:2);
  close (f);
readkey;
end.

```

6-njy mysal. 'text.txt' tekst faýly berlipdir. Ol faýldaky baş harplaryň, setir harplaryň hem-de sanlaryň mukdaryny kesgitlemek üçin programma ýazmaly.

```

Uses crt;
var f: text;
    c: char;
    bh,sh,sifr: integer;
begin
  Assign(f,'text.txt'); reset(f);
  bh:=0; sh:=0; sifr:=0;
  while not eof(f) do
    begin

```



```

    read(f,c);
    if (65<=ord(c) and (ord(c)<=90) then inc(bh);
    if (97<=ord(c) and (ord(c)<=122) then inc(sh);
    if (48<=ord(c) and (ord(c)<=57) then inc(sifr);
end;
writeln ('Baş harplaryň sany = ',bh);
writeln ('Setir harplaryň sany = ',sh);
writeln ('Sifrleriň sany = ',sifr);
close (f);
readkey;
end.

```

7-nji mysal. Faýlyň nusgasyny (kopiýasyny) almaklyga mümkinçilik berýän programma ýazmaly.

```

Uses crt;
var fi,fo: file of byte;
    f1,f2: string;
    b: byte;
begin
    write ('Faýlyň adyny giriziň: '); readln(f1);
    write ('Faýlyň kopiýasy: '); readln(f2);
    Assign (fi,f1); reset(fi);
    Assign (fo,f2); rewrite(fo);
    while not eof(fi) do
    begin
        read (fi,b);
        write (fo,b);
    end;
    close (fi); close (fo);
end.

```



Paskal dilinde modulyň gurluşy

Paskalda tipleriň, hemişelikleriň, üýtgeýän ululyklaryň, belgileriň, funksiýalaryň we proseduralaryň kesgitlemelerini ýörite usul bilen toplamak bolýar. Şeýle topluma **modul (UNIT)** diýip at berilýär. Modul programmadan tapawutlylykda, özbaşdak ýerine ýetirmäge goýberilip bilinmeýär. Ol diňe programmany ýa-da başga moduly döretmekde ulanylýar. Paskalyň öz standart modullary bardyr: SYSTEM, CRT, GRAPH, DOS.

Modul aýratyn faýlda ýazylýar we ol kompilýasiýa edilende TPU (Turbo Pascal Unit) giňeltmeli faýl döreyär. Programmada moduly ulanmak üçin programmanyň başynda Uses sözüniň yzýndan modulyň adyny ýazmaly. Şondan soň programmanyň içinde şol modulyň islendik funksiýasyny, prosedurasyny, hemişeliklerini, üýtgeýän ululyklaryny ulanyp bolýar.

Eger programmada modul beýan edilen bolsa, onda ol programma ýerine ýetirilmäge ugradylanda ilki başda beýan edilen modullar ýerine ýetirilýär, şondan soň programmanyň operatorlar bölümi ýerine ýetirilýär.

Modulyň umumy gurluşy aşakdaky ýaly bolýar:

```
UNIT   Modulň_ady:
INTERFACE
beýan ediji bölüm
IMPLEMENTATION
ýerine ýetirilýän bölüm
BEGIN
Modulyň inisializirleýän bölümi

END.
```

Bu ýerde UNIT – modulyň sözbaşysyny görkezýän ýörite söz;
Modulyň_ady – modulyň ady (dogry identifikator);



INTERFACE – interfeýs bölümi başlaýar;
IMPLEMENTATION – ýerine ýetirilýän bölümiň başlanýan ýeri.

Begin – modulyň inisializirleýän bölüminiň başyny görkezýär.

END – modulyň soňuny görkezýär.

Bu bölümleriň düzümleriniň hiç biri hem hökman dälendir we bolman hem biler.

Aşakda modul ýasalanda hökman bolmaly sözleri görkezýän boş modul görkezilýär:

```
UNIT Bosh;  
INTERFACE  
IMPLEMENTATION  
END.
```

Bu mysaldan görnüşi ýaly, esasy sözler bolan INTERFACE, IMPLEMENTATION sözleriň zyzndan nokatly otur belgisi goýulmaýar, eger ýerine ýetirilýän bölüm girizilmeyän bolsa, onda şol bölümiň başlaýan BEGIN sözi ýazylmaýar.

Modulyň sözbaşysynda onuň ady girizilýär. Modulyň ady unikal (ýagny modulyň içinde gaýtalanmaly dälendir) bolmalydyr we modulyň tekstini saklaýan PAS giňeltmeli (kompilýasiýa edilenden soň bolsa TPU giňeltmeli) faýlyň adyna gabat gelmelidir.

Modulyň adynyň uzynlygy identifikator hökmünde 64 simwola çenli bolup biler. Ýöne diskdäki faýlyň adynyň uzynlygy sekiz simwoldan geçmeli dälendir. Şeýle-de bolsa, modulyň adyny sekiz simwoldan geçirmejek bolmagyň zerurlygy ýokdur. Eger modulyň ady sekiz simwoldan geçse, onda onuň başky sekiz simwolynyň faýlyň ady bilen gabat gelmegi hökmanydyr. Esasy programmanyň USES bölümünde modulyň sözbaşysyndaky ýaly doly at görkezilmedir.

INTERFACE sözi bilen başlanýan beýan ediji bölümde tipler, hemişelikler we üýtgeýän ululyklar kesgitlenilýär. Esasy programma-da modul beýan edilenden soň bu kesgitlenen tipler, hemişelikler we üýtgeýän ululyklar şol esasy programma elýeterli bolýar. INTERFACE bölümünde modula girýän funksiýalaryň we proseduralaryň sözbaşylary beýan edilýär. Bu bölümde funksiýalaryň we proseduralaryň



diňe sözbaşylarynyň bolmagynyň esasy sebäbi kompilýasiýa edilýän wagtynda moduldaky proseduralaryň mazmuny ulanylmaýar we proseduralaryň mazmuny diňe programma kompanowka edilýän wagty ulanylýar. Sözbaşysy INLINE sözi bilen tamamlanýan proseduralarda bu kadadan çykyp, ol prosedura şu bölümde doly kesgitlenip bilner. INTERRUPT we FORWARD sözler bilen tamamlanýan sözbaşylar rugsat edilýän däldir.

Eger tipler, hemişelikler we kömekçi programmalar beýan edilýän wagtynda başga modullarda kesgitlenen tipler we hemişelikler ulanylýan bolsa, onda şol modullar INTEREACE sözüniň yzyndan USES bölümçede görkezilmelidir. USES bölümçe modullarda iki gezek ulanylyp bilner: 1. INTERFACE bölümünde; 2. IMPLEMENTATION bölümünde. INTERFACE bölümünde diňe zerur modullar görkezilmelidir.

Ýerleşdiriji bölüm (IMPLEMENTATION) adatça INTERFA-CE bölümünde beýan edilen funksiýalaryň we proseduralaryň doly göwresini saklaýar. Bu bölümde funksiýalaryň we proseduralaryň sözbaşylary parametrleri görkezmän, diňe gysgaça görnüşde bolsa hem bolýandyr, Ýöne, eger sözbaşy doly getirilse, onda ol öň beýan edilen bölümdäkä doly gabat gelmelidir.

IMPLEMENTATION bölümünde modulyň öz tipleri, hemişelikleri (konstantalary) we üýtgeýän ululyklary girizilip bilner. Olar bu bölümiň kömekçi programmalary, şeýle hem ýerine ýetirilýän bölümiň operatorlary (eger olar bar bolsa) üçin global diýlip hasaplanýar. Moduly ulanýan esasy programma üçin IMPLEMENTATION bölümünde beýan edilen maglumatlar we hemişelikler elýeterli däldir.

Eger proseduralaryň göwresinde tipler ýa-da üýtgeýän ululyklar kesgitlenende başga modullarda beýan edilen zatlary ulanmak gerek bolsa we ol modullar INTERFACE bölümünde beýan edilmedik bolsa, onda şol modullary IMPLEMENTATION sözüniň yzynda USES bölümçede beýan etmelidir.

IMPLEMENTATION bölümünde modulda beýan edilmedik, ýöne beýleki funksiýalaryň we proseduralaryň işine gatnaşýan, kömekçi programmalar kesgitlenip bilner.

Ýerine ýetirilýän bölüm modulyň ýazgysyny jemleýär. Eger bu bölüm ýok bolsa, onda IMPLEMENTATION bölümüniň iň soňky



bölek programmasy tamamlanandan soň END sözi ýazylyar we nokat goýulyar. Eger ýerine ýetirilýän bölüm bar bolsa, onda BEGIN sözi ýazylyar we esasy programma ýerine ýetirilip başlamanka ýerine ýetirilmeli işler programmirlenýär. Adatça bu bölümde modulyň üýtgeýän ululyklarynyň başlangyç bahalary we programma işläp başlamanka ýerine ýetirilmeli işler kesgitlenýär. Modullary ulanýan programma elmydama beýan edilen modullaryň ýerine ýetirilýän bölümi ýerine ýetirilip bolandan soň işläp başlaýar.

Modullary döretmeklige degişli mysallar

1-nji mysal. Hakyky a sany natural n derejä göterýän dereje (a,n) funksiýany, $\lg x$ funksiýany hasaplaýan $\lg(x)$ funksiýany özünde saklaýan KTS modulyňy döretmeli.

```

UNIT KTS;
INTERFACE
Function dereje (a:real,n:integer):real;
Function lg (x:real):real;
IMPLEMENTATION
Function dereje;
begin
if a>0 then dereje:= exp (n*ln(a));
if a<0 then if odd(n) then dereje:=-exp(n*ln(abs(a))) else dere-
je:= n*ln(abs(a));
if a=0 then dereje:= 1; end;
Function lg;
Begin lg:= ln(x)/ln(10); end; end.

```

2-nji mysal. Kompleks sanlaryň arifmetikasyny ornaşdyrýan modul döretmeli.

```

UNIT Cmplx;
INTERFACE
type
complex = record
re, im: real;
end;
Procedure AddC (x,y:complex; var z:complex);

```



```
Procedure SubC (x,y:complex; var z:complex);
Procedure MulC (x,y:complex; var z:complex);
Procedure DivC (x,y:complex; var z:complex);
```

IMPLEMENTATION

```
Procedure AddC;
```

```
begin
```

```
z.re:=x.re+y.re;
```

```
z.im:=x.im+y.im
```

```
end {AddC};
```

```
Procedure SubC;
```

```
begin
```

```
z.re:=x.re-y.re;
```

```
z.im:=x.im-y.im
```

```
end {SubC};
```

```
Procedure MulC;
```

```
begin
```

```
z.re:=x.re*y.re-x.im*y.im;
```

```
z.im:=x.re*y.im+x.im*y.re
```

```
end {MulC};
```

```
Procedure DivC;
```

```
Var zz:real;
```

```
begin
```

```
zz:=sqr(y.re)+sqr(y.im); z.re:=(x.re*y.re+x.im*y.im)/zz;
```

```
z.im:=(x.re*y.im-x.im*y.re)/zz end {DivC}; end.
```



Talyplaryň bilimine baha berýän testirleýji programmany döretmek

Talyplaryň bilimlerine baha bermek üçin testirlemäni kompýuterde geçirmeklik birnäçe artykmaçlyklara eýedir. Birinjiden, ol ka-gyz çykajysyny aradan aýyrýar. Ikinjiden bolsa, talyp öz bilim de-rejesini özbaşdak barlamak mümkinçiligini alýar. Üçünjiden, baha dogry goýulýar.

Soraglary ýörite tekst faýlynda saklamak amatly bolýar. Faýlyň gurluşy dürli görnüşde saýlanylýp bilner. Biz faýlyň gurluşyny şeýle saýlaýarys. Her soraga 20 setir berilýär. Şol 20 setiriň ilkinji 19-sy so-ragy beýan etmeklik we birnäçe jogap hödürlemek üçin niýetlenýär. 20-nji setire soragda hödürlenlen jogaplaryň dogrusynyň nomeri ýa-zylýar. Anyklyk üçin soraglary saklaýan faýly Test.txt diýip atlandy-ralyň we ol işjeň bukjada ýerleşen diýip hasap edeliň.

Şeýlelikde, Test.txt faýlyndaky setirleriň n sany 20-ä kratny bolýar, soraglaryň sany bolsa $(n \text{ div } 20)$ -ä deň bolýar.

Programmanyň algoritmi şeýle bolýar:

1. Soraglary saklaýan faýlyň ady soralýar. Ulanyjy faýlyň adyny girizýär.

2. Programma soraglary saklaýan faýly okamak üçin açýar we onda näçe setiriň bardygyny hasaplaýar we ol sany 20-ä bölüp, soraglaryň jemi sanyny (ssany) çykarýar;

3. M köplük boşadylýar ($M := []$). Bu ýerde: M – talyba berlen soraglaryň nomerleri).

4. Berlen dogry jogaplaryň sanyny görkezýän ululyk 0-a deň edi-lip alynýar;

5. Soňra 1- sany aralykdan bir tötänleýin san alynýar.

6. Eger alnan san M köplükde ýok bolsa, onda nomeri şol sana deň bolan sorag ekrana çykarylýar (soraga degişli 20 setiriň diňe 19-sy çykarylýar), soragyň nomeri bolsa M köplüğe goşulýar.



7. Talyb jogaby girizýär. Ol jogap soraga degiřli 20 setiriň 20-nji setirinde ýerleşen san bilen deňeşdirilýär. Eger olar deň gelseler, onda jogap dogry diýip habar çykýar we berlen dogry jogaplaryň sany bir birlik artdyrylýar, eger olar deň gelmeseler, onda dogry däl diýen habar berilýär;

8. Eger talyba berlen soraglaryň sany 10-dan az bolsa 5-nji ädime dolanyp barylýar.

9. Talybyň beren dogry jogaplarynyň sany we oňa berilýän baha ekrana çykarylýar.

Programma aşakdaky görnüşde bolýar:

```
{Testirleyji programma}
```

```
Uses crt; Label 1,2;
```

```
Var M: Set of byte; fname,ss,s:string; f:TEXT; a,ssany,n,k,l,  
j,l:integer;
```

```
BEGIN clrscr;
```

```
randomize;
```

```
ssany:=0;
```

```
write('Soraglary saklaýan faýlyň adyny giriziň');
```

```
readln(fname);
```

```
assign(f,fname);
```

```
reset(f);
```

```
repeat
```

```
readln(f,s);
```

```
ssany:=ssany+1;
```

```
Until eof(f);
```

```
SSany:=SSany div 20;
```

```
Writeln('Faýlda ',SSany,' sorag bar');
```

```
M:=[];
```

```
k:=0; {dogry jogaplaryň sany}
```

```
Writeln('Soraglar');
```

```
delay(1000);
```

```
for i:=1 to 10 do
```



BEGIN

```
clrscr;
reset(f);
writeln(l, '-nji sorag');
1: n:=random(SSany)+1;
if n in M then goto 1 else M:=M+[n];
for j:=1 to (n-1)*20 do readln(f);
for j:=1 to 19 do begin
readln(f,s);
writeln(s);
end;
writeln('Dogry jogabyň nomerini ýaz! ');
readln(l);
readln(f,a);
if l=a then begin
writeln('Dogry'); k:=k+1; end else writeln('Dogry däl');
readln(ss);
END;
Writeln('Siz 10 soragdan ',k,' sanysyny bildiňiz');
Case k of
10 : Writeln('5-lik');
9 : Writeln('4-lik');
8 : Writeln('4-lik');
7 : Writeln('3-lik');
Else Writeln('2-lik');
close(f);
end.
```



Telefon gözleg ulgamyny döretmek

Telefon gözleg ulgamyna Aşgabat şäheriniň öý telefonlarynyň mysalynda serederis. Öý telefonlary baradaky maglumat şu aşakdaky meýdanlary saklaýar:

1. Telefon nomeri;
2. Müşderiniň familiýasy, A.,A.a.;
3. Poçta indeksi;
4. Köçesi;
5. Jaýy;
6. Korpusy;
7. Öýi.

Maglumatlary saklaýan faýlda meýdanlary biri-birinden «nokatly otur» (;) bilen bölýäris. Faýllaryň adyny SPR_NN.CSV görnüşde ýazýarys, bu ýerde: NN – telefon stansiýanyň nomeri.

1-nji mesele. Berlen telefon nomeri boýunça abonent barada maglumat tapmaly.

Meseläniň algoritmi şeýle görnüşde bolýar:

1. Telefon nomeri girizilýär;
2. Programma degişli faýlyň adyny döredýär;
3. Şol faýl işjeň diskden okamak üçin açylýar;
4. Faýldan bir setir okalýar;

5. Okalan setirdäki ilkinji 6 simwoldan düzülen setir girizilen telefon nomeri bilen deňeşdirilýär. Eger olar deň gelse onda okalan setir degişli böleklere bölünip ekrana çykarylýar we programma işini tamamlýar, eger olar deň gelmeseler, onda 4-nji ädime gaýdyp barylýar we gaýtalamak tä gerekli setir tapylýança ýa-da faýlyň soňuna ýetilýänçä dowam edýär.

Programma aşakdaky görnüşde bolýar:

Uses Crt;

Label 1,2;

Var

s,st,s1,fname:string;




```

number,fam,indeks, koche, korpus,jay,oy:String;
f,f1:TEXT;
B:Boolean;
procedure Abonent(s:string; Var n,fam,ind,koch,j,kor,oy:String);
Var i,k:integer;
a:array[1..7] of string;
begin
k:=1;
for i:=1 to 7 do a[i]:='';
for i:=1 to length(s) do
if s[i]=';' then k:=k+1 else a[k]:=a[k]+s[i];
n:=a[1];
fam:=a[2];
ind:=a[3];
koch:=a[4];
kor:=a[5];
j:=a[6];
oy:=a[7];
end;
begin
1: clrscr;
writeln('Telefon nomerini giriziñ');
readln(number);
if length(number)<>6 then goto 1;
st:=copy(Number,1,2);
fname:='SPR_'+st+'.csv';
B:=False;
Assign(f,fname);
Reset(f);
while not eof(f) do begin

```



```

readln(f,s);
s1:=copy(s,1,6);
if s1=number then begin  B:=True;goto 2; end;
    end;
2: if B then begin  Abonent(s,number,fam,indeks, koche, korpus,
jay, oy);
    writeln('Telefon nomeri ',number);
    writeln('AbONENTiň familiýasy ',fam);
    writeln('Poçta indeksi ',indeks);
    writeln('Koçe ',koche);
    writeln('Korpus ',korpus);
    writeln('Jaý ',jay);
    writeln('Öý ',oy);
        end else writeln('Telefon tapylmady');
close(f);
end.

```

2-nji mesele. Müşderiniň familiýasy boýunça telefon nomerini tapmaly.

Meseläniň algoritmi şeýle görnüşde bolýar:

1. Müşderiniň familiýasy girizilýär;
2. Girizilen familiýanyň uzynlygy len tapylýar;
3. Programma bazadaky birinji faýly okamak üçin açýar;
4. Faýldan bir setir okalýar;
5. Okalan setiriň 8-nji simwolyndan başlap len sany simwolyndan s1 düzülýär. Ol setir girizilen «müşderiniň familiýasy» bilen deňeşdirilýär. Eger olar deň gelse, onda okalan setir degişli böleklerle bölünip ekrana çykarylýar, eger olar deň gelmeseler, onda ol setir ekrana çykarylmaýar. Programma ýene-de açylan faýldan bir setir okaýar we ýokardaky işi gaýtalaýar. Gaýtalamak tä faýlyň soňuna yetilýänçä dowam edýär.

6. Eger bazada ýene-de faýl bar bolsa, onda ol faýl okamak üçin açylýar we 4-nji ädime geçilýär. Eger bazadaky faýllar doly derňelen



bolsa, onda programma öz işini tamamlayar.

Goý, anyklyk üçin telefon bazasynda aşakdaky faýllar bar diýip hasap edeliň:

SPR_31.csv, SPR_32.csv, SPR_33.csv, SPR_34.csv, SPR_35.csv, SPR_39.csv, SPR_41.csv, SPR_42.csv, SPR_43.csv, SPR_44.csv, SPR_45.csv, SPR_48.csv, SPR_49.csv, SPR_50.csv, SPR_51.csv.

Programma aşakdaky görnüşde bolýar:

```
Uses Crt;
```

```
Var
```

```
M: set of byte;
```

```
s,st,s1,fname:string;
```

```
number,fam,indeks, koche, korpus,jaý, oy: String;
```

```
f,f1:TEXT;
```

```
B:Boolean;
```

```
len, i:integer;
```

```
procedure Abonent(s:string; Var n,fam,ind,koch,j,kor,oy:String);
```

```
Var i, k:integer;
```

```
a:array[1..7] of string;
```

```
begin
```

```
k:=1;
```

```
for i:=1 to 7 do a[i]:='';
```

```
for i:=1 to length(s) do
```

```
if s[i]= ';' then k:=k+1 else a[k]:=a[k]+s[i];
```

```
n:=a[1]; fam:=a[2]; ind:=a[3]; koch:=a[4];
```

```
kor:=a[5]; j:=a[6]; oy:=a[7];
```

```
end;
```

```
begin
```

```
M:=[31, 32, 33, 34, 35, 39, 41, 42, 43, 44, 45, 48, 49, 50, 51];
```

```
clrscr; B:=False;
```

```
writeln('Abonentin familiýasyny giriziň');
```

```
readln(fam); len:=Length(fam);
```

```
for i:=1 to 51 do
```

```
if i in M then BEGIN
```

```
str(i,st);
```

```
fname:='SPR_'+st+'.csv';
```



```

Assign(f,fname); Reset(f);
while not eof(f) do begin
readln(f,s);
s1:=copy(s,8,len);
if s1=fam then begin      B:=True;
Abonent(s,number,fam,indeks, koche, korpus, jay, oy);
writeln('Telefon nomeri ', number);
writeln('Abonentin familiyasi ', fam);
writeln('Poçta indeksi ', indeks);
writeln('Koçe ', koche);
writeln('Korpus ', korpus);
writeln('Jay ', jay);
writeln('Öý ', oy);
writeln('-----');
end;      end;  close(f);      END;
if not B then writeln('Abonent tapylmady');
end.

```



PEÝDALANYLAN EDEBIÝATLAR

1. *Gurbanguly Berdimuhamedow*. Ösüşiň täze belentliklerine tarap. Saýlanan eserler. I tom. – A.: Türkmen döwlet neşirýat gullugy, 2008.
2. *Gurbanguly Berdimuhamedow*. Ösüşiň täze belentliklerine tarap. Saýlanan eserler. II tom. – A.: Türkmen döwlet neşirýat gullugy, 2009.
3. *Gurbanguly Berdimuhamedow*. Garaşsyzlyga guwanmak, Watany, halky söýmek bagtdyr. – A.: Türkmen döwlet neşirýat gullugy, 2007.
4. *Gurbanguly Berdimuhamedow*. Türkmenistan – sagdynlygyň we ruhubelentligiň ýurdy. – A.: Türkmen döwlet neşirýat gullugy, 2007.
5. *Gurbanguly Berdimuhamedow*. Türkmenistanyň dermanlyk ösümlikleri. V tom. – A. : Türkmen döwlet neşirýat gullugy, 2013.
6. *Gurbanguly Berdimuhamedow*. Döwlet guşy. – A.: Türkmen döwlet neşirýat gullugy, 2013.
7. *Gurbanguly Berdimuhamedow*. Bilim – bagtyýarlyk, ruhubelentlik, rowaçlyk. – A.: Türkmen döwlet neşirýat gullugy, 2014.
8. «Türkmenistanyň Prezidentiniň obalaryň, şäherleriň, etrapdaky şäherçeleriň we etrap merkezleriniň ilatynyň durmuş-ýaşayyş şertlerini öz-gertmek boýunça 2020-nji ýyla çenli döwür üçin milli Maksatnamasy». – A.: Türkmen döwlet neşirýat gullugy, 2007.
9. Türkmenistanyň Konstitusiyasy. – A.: Türkmen döwlet neşirýat gullugy, 2016.
10. «Türkmenistanyň nebitgaz senagatyny ösdürmegiň 2030-njy ýyla çenli döwür üçin Maksatnamasy». – A.: Türkmen döwlet neşirýat gullugy, 2006.
11. *Nurgeldiyew O.* we başg. Informatika we kompýuter tehnika darsinden meseleler ýygındysy. Orta, orta hünär we ýokary okuw mekdepleri üçin okuw gollanmasy. – A.: Türkmen döwlet neşirýat gullugy, 2012.
12. *Архангельский А.Я.* Delphi 7. Справочное пособие. – М.: Издательство БИНОМ, 2006.
13. *Архангельский А.Я.* Программирование в Delphi. – М.: Издательство БИНОМ, 2008.
14. *Фаронов В.В.* Turbo Pascal 7.0. Начальный курс. Учебное пособие. – М.: Издательство «Нолидж», 2010.



MAZMUNY

Giriş.....	7
------------	---

I bap

Algoritm hakynda düşünje

Algoritm düşünjesi. Algoritmleriň esasy häsiýetleri.....	8
Algoritmiň beýan ediliş usullary.....	10
Algoritmleriň görnüşleri	12

II bap

Programma düzmegiň esaslary

Programmirlenmegiň dillerine umumy syn	15
Paskal diliniň esasy simwollary. Kömekçi sözler	16
Hemişelikler	17
Üýtgeýän ululyklar.....	18
Standart funksiýalar	20
Aňlatmalar.....	23
PASKAL dilinde programmanyň gurluşy	24
Paskal dilinde tipler.....	28
Skalýar tipler	28
Strukturaly tipler barada düşünje	32
Paskal algoritmik diliniň esasy operatorlary.....	33
Baha bermek operatory	33
Giriş-çykyş operatorlary	35
Geçiş operatory	39
Şertli operator.....	40
Saýlaw operatory	43
Gaýtalanmak operatorlary. FOR operatory.....	46
WHILE we REPEAT operatorlary.....	48
Düşündirişler (kommentariler).....	52
Massiwler.....	53
Funksiýalar we proseduralar	57



III bap

Simwollar we setirler bilen işlemek

Simwollar bilen işlemek	66
Setirler. Setirler bilen işlemek üçin proseduralar we funksiýalar	68
Köplükler	71
Paskal algoritmik dilinde ýazgylyar (kombinirlenen tip)	77

IV bap

Faýllar bilen işlemek

Programmirleme dilinde faýl düşünjesi	83
Faýlyň tipi (giňeldilmesi) barada	84
Fiziki we logiki faýllar. Faýl üýtgeýän ululygy bilen diskdäki faýly bellemek (baglanyşdyrmak)	85
Faýldan okamak ýa-da ýazmak. Faýly ýapmak	86
Tekst faýllary	87
Ikilik tipleşdirilen faýllar	88
Faýllar bilen işlemeklige degişli dürli mysallar	91

V bap

Modullar

Paskal dilinde modulyň gurluşy	96
Modullary döretmeklige degişli mysallar	99

VI bap

Dürli amaly programmalary döretmek

Talyplaryň bilimine baha berýän testirleýji programmany döretmek	101
Telefon gözleg ulgamyny döretmek	104
Peýdalanylan edebiýatlar	109

Baýramgeldi Atajanow, Orazberdi Nurgeldiýew

PROGRAMMIRLEMENIŇ ESASLARY

Ýokary okuw mekdepleri üçin okuw kitaby

Redaktor	<i>A. Kiçiyew</i>
Surat redaktory	<i>G. Orazmyradow</i>
Teh. redaktor	<i>O. Nurýagdyýewa</i>
Kompýuter bezegi	<i>M. Çaryýew</i>
Neşir üçin jogapkär	<i>A. Myratlyýew</i>

Çap etmäge rugsat edildi 05.05.2017. Ölçeği 60x90^{1/16}.
Edebi garniturasy. Çap listi 7,0. Şertli-reňkli ottiski 22,25.
Hasap-neşir listi 5,38. Şertli çap listi 7,0.
Sargyt № 27. Sany 1100.

Türkmen döwlet neşirýat gullugy.
744000. Aşgabat. Garaşsyzlyk şaýoly, 100.

Türkmen döwlet neşirýat gullugynyň Metbugat merkezi.
744015. Aşgabat. 2127-nji (G. Gulyýew) köçe, 51/1.