

**B.BOLTAYEV, A.AZAMATOV, SH.XIDIROV, B.XURRAMOV
VA K.ISKANDAROV**

**ALGORITMLASH
VA PASKAL
DASTURLASH TILI
BO'YICHA BERILGAN
MISOL VA
MASALALARNI YECHISH
USULLARI**



TOSHKENT 2012

УДК: 004.421

КБК 22.12

A45

МАТЕМАТИК МАНИИ

Algoritmlash va Paskal dasturlash tili bo'yicha Berilgan misol va masalalarni yechish usullari / B. Boltayev [va boshq.]. – Toshkent: Nihol, 2012. – 164 b.

ISBN 978-9943-23-057-6

УДК: 004.421

КБК 22.12

Respublika Ta'lim Markazi huzuridagi Informatika ilmiy metodika kengashining 2012-yil 11-sentyabrdagi yig'ilishida o'qituvchilar uchun metodik qo'llanma sifatida chop etishga tavsiya etildi

Taqrizchilar:

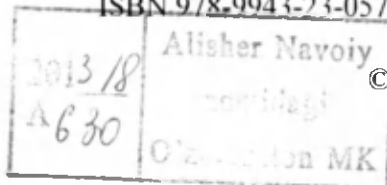
A.Usmonov - O'zbekiston davlat jahon tillari universiteti dotsenti texnika fanlari nomzodi;

Z.Xudayberdiyeva – Toshkent shahar 157-umumta'lim maktabining informatika o'qituvchisi.

R.Muxtorova – Toshkent viloyati Piskent tumani 6- umumta'lim maktabining I-toifali informatika o'qituvchisi.

Umumiy o'rta ta'limda «Informatika va hisoblash texnikasi asoslari» ta'limining asosiy vazifalaridan biri o'quvchilarga kompyuterda masalalar yechish texnologiyalari va uning asosiy bosqichlari haqida bilim berishni ifodalaydi. Ushbu uslubiy qo'llanmada 9-sinf «Informatika» darsligidagi algoritmlash va Paskal dasturlash tili bo'yicha berilgan misol va masalalar yechimlari to'liq yoritilgan.

ISBN 978-9943-23-057-6



© "Ofset Print" MChJ, "Nihol" nashriyoti, 2012

SO'ZBOSHI

Zamonaviy kompyuterlarda turli dasturlash tillari keng qo'llaniladi. Bu dasturlar iqtisodiyot, boshqarish, xizmat ko'rsatish va ayniqsa sanoat va ishlab chiqarishning turli sohalarida muhim ahamiyatga molik masalalarni hal etishda ba'zan yagona omilga aylanmoqda. Bu esa o'z navbatida muxandislik va boshqarish sohasi xodimlari uchun kompyuterlardan unumli va oqilona foydalanishni taqozo etadi.

Kompyuterda biror muammoni hal qilish bir necha bosqichlarga bo'linadi. Avvalambor, tahlil etilayotgan jarayon yoki inshootning zarur jixatlarini o'zida mumkin qadar to'liq aks ettirgan matematik in'ikosi (modeli) tuzib olinadi.

Odatda, matematik model formula va tenglamalar sistemasi ko'rinishida ifodalanadi. Keyingi qadamda hosil bo'lgan matematik masalani yechish uchun eng maqbul hisoblash algoritmi tuziladi.

Va nihoyat, biror algoritmik til asosida tuzilgan dastur bo'yicha olingan natijalar tahlil qilinadi va ular asosida o'rganilayotgan jarayon uchun xos bo'lgan umumiy qonuniyatlar aniqlanadi.

Algoritm – biror masalaning yechilishi uchun zarur bo'lgan buyruqlarning tartiblangan ketma-ketligi bo'lib, odatda so'zlar yordamida yoki blok-sxema ko'rinishida tasvirlanadi. Algoritm blok-sxema ko'rinishi o'quvchilar uchun ko'rinishli va ko'proq tushunarlidir. Bloklardan foydalanib ixtiyoriy masalani yechish uchun algoritmlar tuzish mumkin, bunda algoritmning asos tuzilmalaridan foydalanish maqsadga muvofiq bo'ladi.

Ushbu uslubiy qo'llanma 9-sinf «Informatika» darsligiga to'la mos keladi. Darslikdagi algoritmlash va Paskal dasturlash tili bo'yicha berilgan misol va masalalar yechimlari to'liq yoritilgan.

Qo'llanmadan maktab o'qituvchilari va o'quvchilaridan tashqari, akademik litsey, kasb- hunar kollejlari hamda oliy o'quv yurtlari talabalari ham foydalanishlari mumkin.

Mualliflar uslubiy qo'llanmaning sifatini yaxshilashga qaratilgan barcha fikr-mulohaza va takliflarni minnatdorchiilik bilan qabul qildi.

1 – dars. Masalalarni kompyuterda yechish bosqichlari mavzusiga

M-1. Turg'un suvdagi tezligi 15 km/soat bo'lgan qayiqling daryo oqimi bo'ylab 2 soatdagi bosib o'tgan masofasi oqimga qarshi 3 soatda bosib o'tgan masofasiga teng bo'lsa, daryo oqimining tezligini toping (yo'llanma: tezlik=yo'l / vaqt).

Bu mavzudagi masalalarni hal etishda dasturlash bilan bog'liq bosqichdan tashqari kompyuterda masala yechishning barcha bosqichlarini ko'rib chiqamiz.

Masalaning qo'yilishi.

Berilgan:

$V_T = 15$ km/soat {qayiqling turg'un suvdagi tezligi}

$T_{ob} = 2$ soat {ob - oqim bo'ylab}

$T_{oq} = 3$ soat {oq - oqimga qarshi}

Topish kerak: V_{do} {daryo oqimining tezligi}

Masalaning modelini tuzish.

$$S_{ob} = (V_T + V_{do}) \cdot T_{ob}; S_{oq} = (V_T - V_{do}) \cdot T_{oq}.$$

Shartga ko'ra: $S_{ob} = S_{oq}$ yoki $(V_T + V_{do}) \cdot T_{ob} = (V_T - V_{do}) \cdot T_{oq}.$

Bundan quyidagi model hosil bo'ladi:

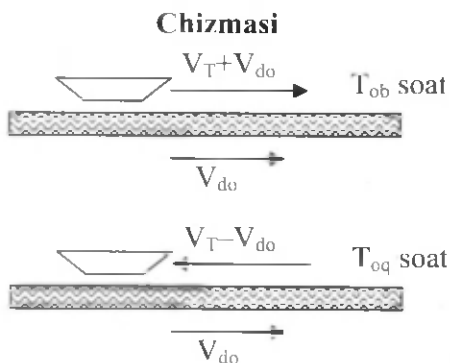
$$V_{do} = (V_T \cdot T_{oq} - V_T \cdot T_{ob}) / (T_{ob} + T_{oq}).$$

Algoritm tuzish. {Bajarilayotgan amallar aniq va tushunarli bo'lishi uchun A, B, D, E kabi belgilashlar kiritib olingan}

- 1) $A = V_T \cdot T_{oq};$
- 2) $B = V_T \cdot T_{ob};$
- 3) $D = T_{ob} + T_{oq};$
- 4) $E = A - B;$
- 5) $V_{do} = E / D.$

Natija olish va tahlil etish.

- 1) $A = 15 \cdot 3 = 45;$
- 2) $B = 15 \cdot 2 = 30;$
- 3) $D = 2 + 3 = 5;$



$$4) E=45-30=15;$$

$$5) V_{do}=15/5=3.$$

Natijaning tahlili: $S_{ob}=(15+3) \cdot 2=36$, $S_{oq}=(15-3) \cdot 3=36$. Demak, natija to'g'ri.

Javob: $V_{do}=3$ km/soat.

M-2. To'g'ri burchakli to'rtburchakning tomonlari, mos ravishda, 4 sm va 3 sm bo'lsa, uning diagonalini uzunligini toping (yo'llanma: to'g'ri to'rtburchakning diagonalini to'rtburchakni ikkita to'g'ri burchakli uchburchakka ajratadi, demak, diagonal gipotenuza bo'ladi).

Masalaning qo'yilishi.

Berilgan:

to'rtburchak to'g'ri burchakli, demak diagonalni to'rtburchakni ikkita to'g'ri burchakli uchburchakka ajratadi

$$a=4 \text{ sm}$$

$$b=3 \text{ sm}$$

Topish kerak: c diagonal

Masalaning modelini tuzish.

Pifagor teoremasi: $c^2 = a^2 + b^2$;

Algoritm tuzish. {Bajarilayotgan amallar aniq va tushunarli bo'lishi uchun AA, BB, CC kabi belgilashlar kiritib olingan}

$$1) AA = a \cdot a;$$

$$2) BB = b \cdot b;$$

$$3) CC = AA + BB;$$

$$4) c = \sqrt{CC}.$$

Natija olish va tahlil etish.

$$1) AA = 4 \text{ sm} \cdot 4 \text{ sm} = 16 \text{ sm}^2;$$

$$2) BB = 3 \text{ sm} \cdot 3 \text{ sm} = 9 \text{ sm}^2;$$

$$3) CC = 16 \text{ sm}^2 + 9 \text{ sm}^2 = 25 \text{ sm}^2;$$

$$4) c = \sqrt{CC} = \sqrt{25 \text{ sm}^2} = 5 \text{ sm}.$$

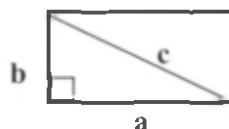
Natijani tahlili:

$$(5 \text{ sm})^2 = (4 \text{ sm})^2 + (3 \text{ sm})^2 \text{ yoki } 25 \text{ sm}^2 = 16 \text{ sm}^2 + 9 \text{ sm}^2.$$

Demak, natija to'g'ri.

Javob: $c=5$ sm.

Chizmasi



2 – dars. Model va uning turlari mavzusiga

M-1. Obyekt: viloyatlar (yo'llanma: nomi, maydoni, aholisi soni, asosiy iqtisodiy mahsuloti, ...).

Viloyatlar tavsifi	Nomi	Maydoni	Aholi soni	Asosiy iqtisodiy mahsuloti
1	Andijon	4,2 ming km ²	2451,2 mingta	paxta, avtomobil
2	Surhondaryo	20,1 ming km ²	1992,1 ming ta	paxta, bug'doy

M-2. Obyekt: sinfdoshlar (yo'llanma: jinsi, bo'yi, soch rangi, og'irligi, ko'z rangi, ...).

Sinfdoshlar tavsifi	Ismi	Jinsi	Bo'yi	Og'irligi	Ko'z rangi
1	Bahodirjon	Erkak	172 sm	65 kg	Qora
2	Hilolaxon	ayol	165 sm	52 kg	Moviy

M-3. Obyekt: kitoblar (yo'llanma: nomi, sahifa soni, rangliligi, og'irligi, narxi, ...).

Kitoblar tavsifi	Nomi	Muallifi	Mavzusi	Sahifa soni	Og'irligi
1	Informatika	B.Boltayev va b.	Darslik	144	250 gr
2	O'tikan kunlar	Abdulla Qodiriy	Roman	250	480 gr

3 – dars. Masalalarni kompyuterda yechish bosqichlari va modelning turlari mavzularini takrorlash mavzusiga

M-1. Quyidagi masalalar shartini tahlil qiling va bosqichlarga bo'lib hal eting.

A. Katetlari a va b bo'lgan to'g'ri burchakli uchburchakning gipotenuzasini hisoblang.

B. Katetlari a va b bo'lgan to'g'ri burchakli uchburchakning yuzasini hisoblang.

A va B masalalar yechimi avvalgi mavzudagi masala kabi ifodanadi.

D. Tomoni a bo'lgan teng tomonli uchburchakning balandligini toping.

Masalaning qo'yilishi.

Berilgan:

uchburchak teng tomonli, demak barcha tomonlari uzunligi a ga teng.

Topish kerak: **h** balandlik

Masalaning modelini tuzish.

Masalani modelini tuzish uchun uchburchak yuzasini hisoblashning quyidagi ikki formulasidan foydalanamiz va balandlikni aniqlaymiz.

1) Geron formulasi:

$$p = \frac{a + a + a}{2} = \frac{3 \cdot a}{2};$$

$$S = \sqrt{p \cdot (p - a) \cdot (p - a) \cdot (p - a)} =$$

$$= \sqrt{\frac{3 \cdot a}{2} \cdot \left(\frac{3 \cdot a}{2} - a\right) \cdot \left(\frac{3 \cdot a}{2} - a\right) \cdot \left(\frac{3 \cdot a}{2} - a\right)} =$$

$$= \sqrt{\frac{3 \cdot a}{2} \cdot \frac{a}{2} \cdot \frac{a}{2} \cdot \frac{a}{2}} = \left(\frac{a}{2}\right)^2 \cdot \sqrt{3};$$

$$2) S = \frac{a \cdot h}{2};$$

$$3) h = \frac{2 \cdot S}{a}.$$

Algoritm tuzish.

$$1) S = \left(\frac{a}{2}\right)^2 \cdot \sqrt{3};$$

$$2) h = \frac{2 \cdot S}{a}.$$

Natija olish va tahlil etish.

Bu bosqichda uchburchak tomoni bo'lgan **a** ni qiymatini turlicha olib natijani tahlil etish mumkin.

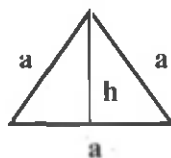
M-2. Quyidagi obyektlarni tavsifi va tavsif qiymatini yozing.

a) Obyekt: o'zingiz yashaydigan viloyat (shahar) kollejlari (yo'llanma: nomi, qurilgan yili, yo'nalishlari, qabul qilinadigan o'quvchilar soni, ...).

b) Obyekt: Asaka avtomobil zavodi ishlab chiqaradigan avtomobillari (markasi, ishlab chiqarila boshlangan yili, soni, ranglari, ...).

a) va **b)** mashqlar yechimi avvalgi mavzudagi jadvaldagi kabi ifodanadi.

Chizmasi



M-3. Quyidagi masalalarga mos model tuzing va hal eting.

A. Bankka yiliga A foizli daromad olish uchun qo'yilgan B so'm pulning M yildan keyingi holatini ifodalovchi model tuzing.

Yechim. 1-yil oxirida olinadigan daromad $\frac{B}{100} \cdot A$ so'm bo'ladi.

Shuning uchun yil oxirida bankdagi pul $\frac{B}{100} \cdot A + B = B \cdot \left(\frac{A}{100} + 1\right)$

so'm bo'ladi. Ikkinchi yil oxirida olinadigan daromad $B \cdot \left(\frac{A}{100} + 1\right) \cdot \frac{A}{100}$ so'm bo'ladi. Ikkinchi yil oxirida bankdagi pul

$B \cdot \left(\frac{A}{100} + 1\right) \cdot \frac{A}{100} + B \cdot \left(\frac{A}{100} + 1\right) = B \cdot \left(\frac{A}{100} + 1\right) \cdot \left(\frac{A}{100} + 1\right) = B \cdot \left(\frac{A}{100} + 1\right)^2$

so'm bo'ladi.

Quyidagicha jadval tuzib olamiz:

Yillar	Foiz daromadi	Bankdagi pul miqdori
1-yil oxirida	$\frac{B}{100} \cdot A$	$B \cdot \left(\frac{A}{100} + 1\right)$
2-yil oxirida	$B \cdot \left(\frac{A}{100} + 1\right) \cdot \frac{A}{100}$	$B \cdot \left(\frac{A}{100} + 1\right)^2$
3-yil oxirida	$B \cdot \left(\frac{A}{100} + 1\right)^2 \cdot \frac{A}{100}$	$B \cdot \left(\frac{A}{100} + 1\right)^3$
4-yil oxirida	$B \cdot \left(\frac{A}{100} + 1\right)^3 \cdot \frac{A}{100}$	$B \cdot \left(\frac{A}{100} + 1\right)^4$
...
M -yil oxirida	$B \cdot \left(\frac{A}{100} + 1\right)^{M-1} \cdot \frac{A}{100}$	$B \cdot \left(\frac{A}{100} + 1\right)^M$

Demak, masalaga mos matematik model: M yildan keyin bankdagi

pul $B \cdot \left(\frac{A}{100} + 1\right)^M$ so'm.

B. Samolyot orasidagi masofa 2400 km bo'lgan A shahardan B shahargacha 3 soat, orasidagi masofa 4800 km bo'lgan B shahardan M shahargacha 6 soat uchdi. Samolyot qanday o'rtacha tezlikda uchgan? (yo'llanma: o'rtacha tezlik = (1-yo'l + 2-yo'l) / (1-vaqt + 2-vaqt)).

Yechim berilgan yo'llanmaga asoslanib yoziladi, ya'ni masalani

modeli quyidagicha:

$$V_{O'R} = \frac{S_{AB} + S_{BM}}{T_{AB} + T_{BM}}, \{V_{O'R} - A \text{ shahardan } M \text{ shahargacha uchishda-}$$

gi o'rtacha tezlik. S_{AB} - A shahardan B shahargacha, S_{BM} - B shahardan M shahargacha bo'lgan masofalar, T_{AB} - A shahardan B shahargacha, T_{BM} - B shahardan M shahargacha uchish vaqti}

4 – dars. Algoritm tushunchasi mavzusiga

T-4. Darslikdan berilgan mavzuni topish algoritmini tuzing.

T-5. "Oshpalov" pishirish algoritmini tuzing.

T-6. Kompyuterni ishga tushirish algoritmini tuzing.

O'quvchilar bu topshiriqlarga mos algoritm tuza oladilar, lekin keyingi darsda izohlanadigan aniqlik yoki tushunarlilik xossalariiga e'tibor berishmaydi.

Masalan, berilgan mavzuni topish algoritmidagi "mundarijaga qaral-sin" (mundarija kitobning qayerida joylashgan), "kerakli tartib raqamli sahifa ochilsin" (qanday qilib, bir chekkadan varaqlabmi?, u holda mundarijani nima keragi bor).

"Oshpalov" pishirish algoritmidagi esa yog', suv yoki tuz qancha so-linishida adashishadi. Ko'pincha, "ozgina" yoki "keragicha" so'zlari ishlatishadi. Lekin bu kabi ko'rsatmalar ijrochiga tushunarli bo'lmaydi.

O'quvchilarga nomini aytmagan holda shu xossalarni buzilishi qanday oqibatlariga olib kelishini ko'rsatib o'tish bilan keyingi dars-ga zamin tayyorlanadi. Yana, o'quvchilar ijrochi nimalarni "tushuna-di" (ijrochining ko'rsatmalar sistemasi) degan savolga javob berishi ham ularni algoritm haqidagi tasavvurlarini kengaytiradi.

5 – dars. Algoritmning asosiy xossalari mavzusiga

T-5. Algoritmning diskretlilik xossasini misollar asosida tushuntiring.

Masalan, bir joydan boshqa joyga o'tish uchun oyoqlarimiz bir ne-cha yuz marta takrorlanadigan "o'ng oyoq oldinga qo'yilsin", "chap oyoq oldinga qo'yilsin" buyruqlari ketma-ket oyoqlarga ketma-ket be-iriladi. Bu buyruqlar baravar berilsa nima hodisa ro'y berishini o'quvchilarni o'zi izohlashi maqsadga muvofiq.

Kompyuterda matn yozishda ham klaviaturadan har bir belgi ketma-ket berilgan alohida buyruqlar asosida yoziladi. Birdaniga bir necha o'n klavishni bosish mumkinmikan?

T-6. Algoritmning natijaviylik xossasini misollar asosida tushuntiring. Natijaviylik xossasi algoritm qadamining chekliligini anglatadi.

Ichida ovqati yo'q bo'sh qoshiqni og'izga solish ham natija beradi: ishtahani ochadi yoki, qoshiqni og'izga oborishni mashq qildiradi. Umuman, biror maqsadga harakat qilinsa, shu maqsadga erishish ijobiy, maqsadga erishmaslik salbiy natija bo'ladi.

T-7. Natijaviylik xossasi bajarilmaydigan ko'rsatmalar ketma-ketligiga misol keltiring.

Masalan:

- 1) daryodan bir chelak suv olinsin;
- 2) chelakdagi suv daryoga solinsin;
- 3) 1-bandga o'tilsin.

Bu algoritm chekli qadamda tugamaydi, demak natijasi yo'q.

T-8. Algoritmning ommaviylik xossasini misollar asosida tushuntiring.

Al-Xorazmiyning ustunli qo'shish algoritmi barcha o'nli kasr ko'rinishidagi sonlar uchun o'rinli, Evklid algoritmi barcha natural sonlar uchun o'rinli, kvadrat tenglama yechish algoritmi barcha haqiqiy sonlar uchun o'rinli, oyoq kiyimi kiyishda avval paypoq keyin tuffi kiyish (teskarisi emas) barcha aqli raso inson uchun o'rinli.

T-9. Evklid algoritmi yordamida bir nechta natija oling.

$N=15$ va $M=25$ bo'lsa:

$(15, 25) \rightarrow (15, 25-15) \rightarrow (15, 10) \rightarrow (15-10, 10) \rightarrow (5, 10) \rightarrow (5, 10-5) \rightarrow (5, 5)$, javob 5.

$N=18$ va $M=39$ bo'lsa:

$(18, 39) \rightarrow (18, 39-18) \rightarrow (18, 21) \rightarrow (18, 21-18) \rightarrow (18, 3) \rightarrow (18-3, 3) \rightarrow (15, 3) \rightarrow (15-3, 3) \rightarrow (12, 3) \rightarrow (12-3, 3) \rightarrow (9, 3) \rightarrow (6, 3) \rightarrow (6-3, 3) \rightarrow (3, 3)$, javob 3.

$N=91$ va $M=65$ bo'lsa:

$(91, 65) \rightarrow (91-65, 65) \rightarrow (26, 65) \rightarrow (26, 65-26) \rightarrow (26, 39) \rightarrow (26, 39-26) \rightarrow (26, 13) \rightarrow (26-13, 13) \rightarrow (13, 13)$, javob 13.

6 – dars. Algoritm tushunchasi va algoritmning asosiy xossalari mavzularini takrorlash darsi mavzusiga

M-1. Ijrochi sifatida quyidagi ko'rsatmalardan qaysilarini bajara olmaysiz va nima uchun?

A. 200 kg lik tosh ko'tarilsin. B. 7 ga 2 ko'paytirilsin.

D. 1 dan 31622400000 gacha sanalsin.

A ko'rsatmani bajara olmaydi, chunki o'quvchining fizik imkoniyati cheklangan.

D ko'rsatmani bajara olmaydi, chunki o'quvchi to'xtamasdan sekundiga beshta sonni aytsa 6324480000 sekund= 105408000 minut= 1756800 soat= 73200 sutka= 200 yil kerak bo'ladi.

M-2. Algoritm Ijrochisi qo'yilgan maqsadga erishishi uchun qanday sodda ko'rsatmalarni bajara olishi lozimligini, ya'ni ijrochining ko'rsatmalar sistemasini aniqlang.

A. Ochiq eshik ijrochining chap yonida 5 qadam narida bo'lsa, maqsad "eshikdan chiqish".

IKS={chapga buril; bir qadam oldinga yur}.

B. Ijrochi jo'mrak va silindrik stakan oldida turgan bo'lsa, maqsad "yarim stakan suv olish".

IKS={jo'mrak ochilsin; jo'mrak yopilsin; stakaning ochiq tomoni qirrasini va unga yopiq tomondagi qarama-qarshi qirra gorizontal holatda ushlansin}.

D. Berilgan $44 \cdot 15 + 12 \cdot 15 : 20 - 43$ sonli ifoda qiymati aniqlansin.




IKS={bir son ikkinchisiga qo'shilsin; bir son ikkinchisidan ayrilsin; bir son ikkinchisiga ko'paytirilsin; bir son ikkinchisiga bo'linsin}.




M-3. Berilgan ko'rsatmalar yordamida masala yechimiga olib keluvchi algoritm yozing.

A. "Bo'ri, echki va karam" nomli qadimiy masala. Dehqon daryoning chap qirg'og'ida bo'ri, echki va karam bilan turibdi. U bularni hammasini o'ng qirg'oqqa o'tkazishi kerak. Uning qayig'i juda kichik bo'lgani uchun faqat bitta yo'lovchini olishi mumkin – yoki bo'rini, yoki echkini, yoki karamni. Yana, agar bo'ri va echki bir qirg'oqda qoldirilsa bo'ri echkini yeb qo'yadi, agar echki va karamni bir qirg'oqda qoldirilsa echki karamni yeb qo'yadi. Hayvonlar faqat dehqon borligidagina tinch turishadi. Dehqonning ko'rsatmalar sistemasini quyidagicha:

{echkini o'tkaz; bo'rini o'tkaz; karamni o'tkaz; suzib o't}.

echkini o'tkaz suzib o't bo'rini o'tkaz echkini o'tkaz karamni o'tkaz suzib o't echkini o'tkaz	yoki	echkini o'tkaz suzib o't karamni o'tkaz echkini o'tkaz bo'rini o'tkaz suzib o't echkini o'tkaz
--	------	--


B. Bo'g'irsoq uchun "oldindagi" katak qalpoqchasi ko'rsatayotgan katakdir. U o'ngga burilganda  ko'rinishda bo'ladi. Bo'g'irsoq 1 ta oldindagi katakka yura oladi yoki turgan katagida o'ngga burila oladi, ya'ni {oldinga; o'ngga} ko'rsatmalarini bajara oladi. Bo'g'irsoq bir katakdan bir necha marta o'tishi mumkin, lekin  shaklidagi to'siqli katakdan o'ta olmaydi. Bo'g'irsoq o'zi turgan katakdan  bilan belgilangan katakka biror yo'l bilan bora oladigan bo'lsa, zaruriy ko'rsatmalar ketma-ketligini yozing.

Qadamlar soni	1-algoritm	2-algoritm
1	1) o'ngga;	1) o'ngga;
2	2) o'ngga;	2) oldinga;
3	3) o'ngga;	3) o'ngga;
4	4) oldinga;	4) o'ngga;
5	5) oldinga.	5) oldinga;
6		6) oldinga;
7		7) oldinga.

7 – dars. Algoritmni tasvirlash usullari mavzusiga

M-1. Paint grafik muharririda "O'zbekiston kelajagi buyuk davlat!" matnidagi so'zlarni tagma-tag yozdirish uchun so'zlar yordamida algoritm tuzing.

- 1) boshlansin;
- 2) Paint grafik muharriri ishga tushirilsin;
- 3) Paint grafik muharriri Uskunalar panelidan  ("Yozuv") uskunasi tanlansin;
- 4) Ishchi maydonida o'lchami 300x30 pikselga teng yozuv maydoni hosil qilinsin;

- 5) Klaviaturada “**O‘zbekiston**” matni terilsin;
- 6) Klaviaturada Enter klavishi bosilsin;
- 7) Klaviaturada “**kelajagi**” matni terilsin;
- 8) Klaviaturada Enter klavishi bosilsin;
- 9) Klaviaturada “**buyuk**” matni terilsin;
- 10) Klaviaturada Enter klavishi bosilsin;
- 11) Klaviaturada “**davlat!**” matni terilsin;
- 12) Klaviaturada Enter klavishi bosilsin;
- 13) tugallansin.

M-2. MS Word dasturining WordArt obykti yordamida “**O‘zbekiston vatanim manim!**” iborasini yozish algoritmini qulay usulda tasvirlang.

- 1) boshlansin;
- 2) MS Word dasturi ishga tushirilsin;
- 3) “Chizish” uskunalar panelidan “WordArt” uskunasi tanlansin;
- 4) “WordArt” kolleksiasidan yozish usuli tanlansin;
- 5) Klaviaturada Enter klavishi bosilsin;
- 6) “WordArt” uskunasi matn maydoniga “**O‘zbekiston vatanim manim!**” matni yozilsin;
- 7) “OK” tugmasi bosilsin;
- 8) tugallansin.

M-3. Berilgan ikki natural sonning eng kichik umumiy karralisini topish algoritmini tuzing.

- 1) boshlansin;
- 2) M va N sonlari qiymati aniqlansin;
- 3) M va N sonlar ko‘paytmasi EKUK deb olinsin;
- 4) agar $N = M$ bo‘lsa, N soni EKUB deb olinsin va 7)-bandga o‘tilsin;
- 5) N va M sonlarning kattasi aniqlansin va o‘zi bilan kichik sonning ayirmasiga teng deb olinsin;
- 6) 4)-bandga o‘tilsin;
- 7) EKUK ni EKUB ga nisbati EKUK deb olinsin;
- 8) javob sifatida EKUK yozilsin;
- 9) tugallansin.

8 – dars. Algoritmni tasvirlash usullari mavzusiga oid amaliy mashg'ulot mavzusiga

M-1. Quyidagi masalalar algoritmlarini so'zlar yordamida tuzing.

A. Berilgan x da $y = 23 \cdot x - 1963$ funksiyasining qiymatini hisoblash algoritmini tuzing.

- 1) boshlansin;
- 2) x ning qiymati aniqlansin;
- 3) x ni 23 ga ko'paytirib z deb olinsin;
- 4) z dan 1963 ni ayirib y deb olinsin;
- 5) javob sifatida y yozilsin;
- 6) tugallansin.

B. Ijrochining ko'rsatmalar sistemasi faqat {5 ni qo'sh; 3 ni ayir} ko'rsatmalaridan iborat. Bu ijrochi 0 sonidan 11 sonini hosil qilishi uchun algoritm tuzing.

Qadam	Ko'rsatma	Natija
0	–	0
1	5 ni qo'sh	5
2	5 ni qo'sh	10
3	5 ni qo'sh	15
4	5 ni qo'sh	20
5	3 ni ayir	17
6	3 ni ayir	14
7	3 ni ayir	11

D. Ijrochi ko'rsatmalar sistemasi faqat {1 ni qo'sh; 2 ga ko'paytir} ko'rsatmalaridan iborat. Bu ijrochi 0 sonidan 17 sonini hosil qilishi uchun 3 xil usulda algoritm tuzing.

Qadam	Ko'rsatma	Nati-ja		Ko'rsatma	Nati-ja		Ko'rsatma	Natija
0	–	0		–	0		–	0
1	1 ni qo'sh	1		1 ni qo'sh	1		1 ni qo'sh	1
2	2 ga ko'paytir	2		1 ni qo'sh	2		2 ga ko'paytir	2
3	2 ga ko'paytir	4		2 ga ko'paytir	4		2 ga ko'paytir	4
4	2 ga ko'paytir	8		2 ga ko'paytir	8		1 ni qo'sh	5
5	2 ga ko'paytir	16		2 ga ko'paytir	16		2 ga ko'paytir	10

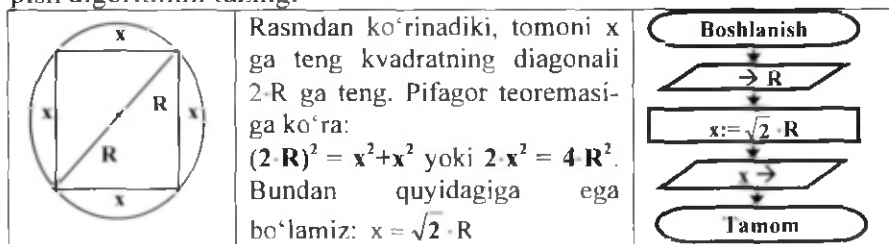
6	1 ni qo'sh	17	1 ni qo'sh	17	2 ga ko'paytir	15
7	-	-	-	-	1 ni qo'sh	16
8	-	-	-	-	1 ni qo'sh	17

E. $A=5$ va $B=8$ bo'lganda Ijrochi 4 litr suv o'lchab olishi uchun algoritm tuzing.

Ko'rsatma	A [5 litr]	B [8 litr]
-	0	0
A ni to'ldir	5	0
A dan B ga quy	0	5
A ni to'ldir	5	5
A dan B ga quy	2	8
B ni bo'shat	2	0
A dan B ga quy	0	2
A ni to'ldir	5	2
A dan B ga quy	0	7
A ni to'ldir	5	7
A dan B ga quy	4	8

M-2. Quyidagi masalalar algoritmlarini blok-sxema yordamida tuzing.

A. Radiusi R ga teng aylana ichiga chizilgan kvadratning tomonini topish algoritmini tuzing.



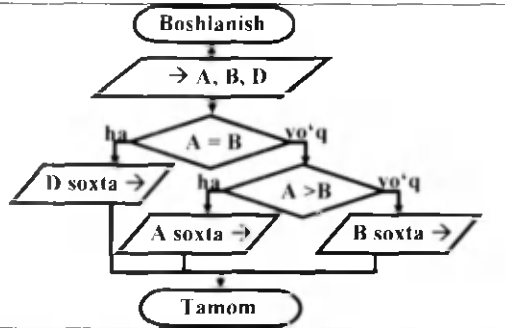
B. Uchta tanga berilgan. Ulardan biri soxta va og'ir. Tortish uchun ikki pallali tarozi o'lchov toshlarisiz berilgan. Soxta tangani aniqlash algoritmini tuzing.

Izoh: Joyni iqtisod etish uchun mazkur qollanmada barcha blok sxemadagi kiritilsin so'zi o'rniga (\rightarrow biror belgi); chop etilsin so'zi o'rniga (biror belgi \rightarrow) kabi belgilashni qo'llashni lozim topildi.

Avval tangalarni A, B, D kabi nomlaymiz.

1) Agar A va B tangalar taroziga qo'yilganda og'irligi teng bo'lsa, u holda D tanga soxta.

2) Aks holda A va B tangalardan og'ir tomondagi tanga soxta bo'ladi.



D. Uchta tanga berilgan. Ulardan biri soxta va faqat og'irligi bilan farqlanadi (aniq og'ir yoki yengilligi ham ma'lum emas). Tortish uchun ikki pallali tarozi o'lchov toshlarisiz berilgan. Eng kam tortish yordamida soxta tangani aniqlash algoritmini tuzing.

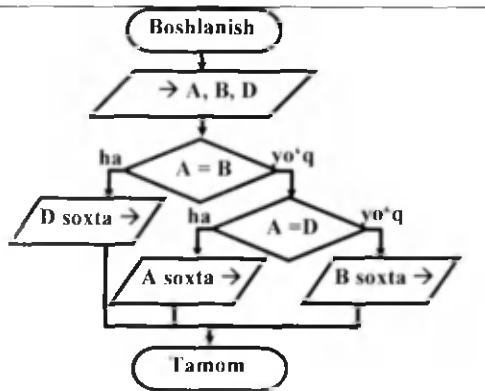
Avval tangalarni A, B, D kabi nomlaymiz.

1) Agar A va B tangalar taroziga qo'yilganda og'irligi teng bo'lsa, u holda D tanga soxta.

2) Agar A va D tangalar taroziga qo'yilganda og'irligi teng bo'lsa, u holda B tanga soxta.

3) Aks holda A tanga soxta.

Demak, soxta tangani ikki marta tortishda aniqlash mumkin.

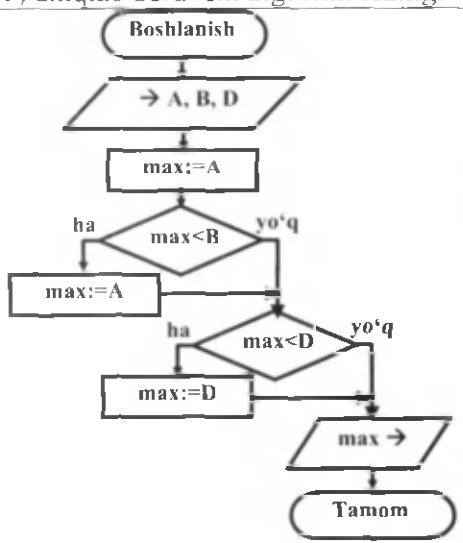


9 – dars. Algoritmning asosiy turlari mavzusiga

M-1. Uchta sondan kattasini (UKT) aniqlab beruvchi algoritm tuzing.

A sonni kattasi deb olamiz. Katta sonni keyingi sonlar bilan birin-ketin taqqoslaymiz va agar keyingi son katta bo'lsa, keyingi sonni kattasi deb olamiz:

- 1) boshlansin;
- 2) A, B, D sonlar qiymati aniqlansin;
- 3) A soni max deb olinsin;
- 4) Agar max soni B sondaan kichik bo'lsa B son max deb olinsin;
- 5) Agar max soni D sondaan kichik bo'lsa D son max deb olinsin;
- 6) javob sifatida max yozilsin;
- 7) tugallansin.



M-2. Quyidagi algoritmlar qanday algoritm turiga misol bo'lishini va natijasini aniqlang:

a) $a:=3$; $x:=2*a+a*a$. $a=?$, $x=?$	Chiziqli algoritm; $a=3$, $x=2*3+3*3=6+9=15$
b) $x:=1$; $x:=x+1$, $x:=x*x-4$. $x=?$	Chiziqli algoritm; $x=1$; $x=1+1=2$; $x=2*2-4=4-4=0$
d) $a:=15$; $b:=a$; $a:=a-b$. $a=?$, $b=?$	Chiziqli algoritm; $b=15$; $a=15-15=0$

e) 1) $a:=3$; 2) agar $a>2$ bo'lsa, u holda $x:=2*a+a*a$ va 4-bandga o'tilsin, aks holda 3-bandga o'tilsin; 3) $x:=9-a*x$; 4) natija x yozilsin; 5) tugallansin.	Tarmoqlanuvchi algoritm. $a=3$; $a(=3)>2$ shart bajariladi, shuning uchun $x=2*a+a*a=2*3+3*3=6+9=15$ Javob: $x=15$
--	---

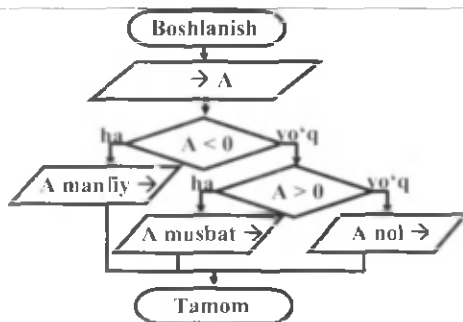
f) 1) $x:=1$; 2) agar $x > 2$ bo'lsa, u holda $x:=x+1$ va 4-bandga o'tilsin, aks holda 3-bandga o'tilsin;	Tarmoqlanuvchi algoritm. $x=1$; $x(=1)>2$ shart bajarilmaydi
---	---

3) $x := x * x - 4$; 4) natija x yozilsin; 5) tugallansin.	di, shuning uchun $x = x * x - 4 = 1 * 1 - 4 = 1 - 4 = -3$ Javob: $x = -3$
---	---

g) 1) $a := 15$; 2) $b := a$; 3) agar $a > b$ bo'lsa, u holda $a := a - b$ va 5-bandga o'tilsin, aks holda 4-bandga o'tilsin; 4) $a := a + b$; 5) natija a, b yozilsin; 6) tugallansin.	Tarmoqlanuvchi algoritm. $a := 15$; $b := a (=15)$; $a (=15) > b (=15)$ shart bajarilmaydi, shuning uchun $a := a + b = 15 + 15 = 30$ Javob: $a = 30$; $b = 15$
---	---

M-3. Berilgan sonni ishorasini aniqlovchi algoritmni blok-sxema yordamida tuzing.

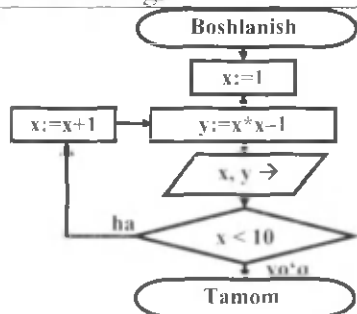
Ma'lumki, A soni manfiy, musbat yoki nol bo'ladi. Shuning uchun avval A sonini 0 dan kichiklikka tekshiramiz, agar son 0 dan kichik bo'lsa "A manfiy" deb javob chiqarib algoritmni tugatamiz. Aks holda A sonni 0 dan kattalikka tekshiramiz, agar son 0 dan kichik bo'lsa "A musbat" deb javob chiqarib algoritmni tugatamiz.



Aks holda A soni manfiy ham, musbat ham bo'lmaydi, shuning uchun "A nol" deb javob chiqarib algoritmni tugatamiz.

M-4. $y = x^2 - 1$ funksiyasini x ning $[1; 10]$ oraliqdagi butun qiymatlarida hisoblash algoritmini blok-sxema shaklida tuzing.

$[1; 10]$ oraliqda x ning butun qiymatlari 1; 2; 3; ...; 10 bo'ladi. Demak, algoritm takrorlanuvchi bo'ladi. Avval $x=1$ da funktsiyani hisoblaymiz, javob sifatida x ni qiymatini va unga mos y ning qiymatini chiqaramiz. Agar x ning qiymati 10 dan kichik bo'lsa, uni qiymatini bittaga oshiramiz, ya'ni $x=x+1$, so'ng yana funktsiya qiymatini hisoblab, x va unga mos y qiymatni chiqaramiz.



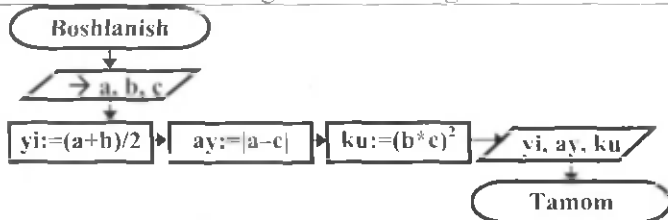
Agar $x=9$ bo'lsa, funktsiya qiymatini hisoblab x va unga mos y qiymatni chiqaramiz. Bu holda ham $x (=9) < 10$ bo'ladi, shuning uchun x ni qiymatini bit-

taga oshiramiz, ya'ni $x=x+1$ ($=10$), funksiya qiymatini hisoblab x va unga mos y qiymatni chiqaramiz. Endi $x=10$ (10 dan kichik emas) bo'lgani uchun algoritmni tamomlaymiz.

10 –dars. Algoritmning asos tuzilmalari mavzusiga

1-masala. Uchta a, b, c sonlar berilgan. a va b sonlar yig'indisi yarminini, a va c sonlar ayirmasini modulini, b va c sonlar ko'paytmasini kvadratini hisoblash algoritmini tuzing.

Bu masala algoritmi chiziqli bo'ladi, chunki faqat qiymat berish va oddiy hisoblashlardan iborat.

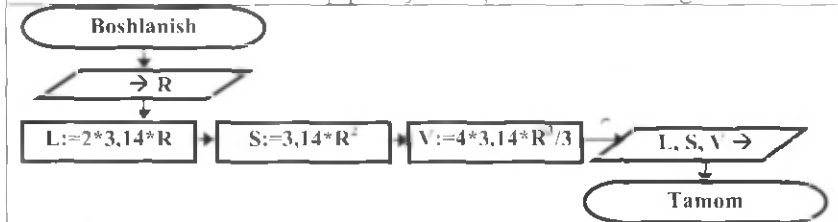


Algoritm blok-sxemasidagi bloklarni tagidan yoki yonidan yozish mumkin.

2-masala. Radiusi R ga teng aylana uzunligi, doira yuzi va shar hajmini hisoblash algoritmini tuzing (yo'llanma: $L=2\pi R$; $S=\pi R^2$;

$$V=\frac{4}{3}\pi R^3).$$

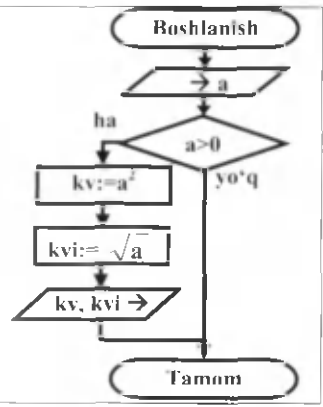
Avvalgi masala kabi bu masala algoritmi ham chiziqli bo'ladi. Algoritm blok-sxemasi ham farq qilmaydi, faqat formulalar o'zgaradi.



3-masala. Berilgan a son musbat bo'lsa, u holda uning kvadratini va kvadrat ildizini hisoblash algoritmini tuzing.

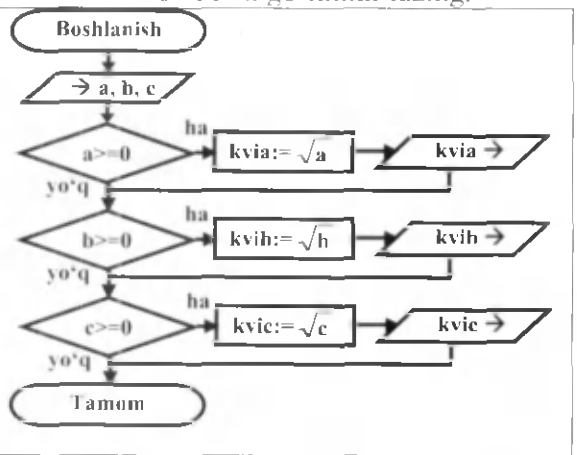
Bu masala algoritmi tarmoqlanuvchi bo'lib, faqat bir holatda, ya'ni $a > 0$ shart bajarilganda amallar bajarilishi talab etilmoqda. Shuning uchun tarmoqlanish tuzilmasini qisqa ko'rinishidan foydalanish qulay.

boshlansin
 a kiritilsin
agar $a > 0$ u holda
 $kv := a^2$
 $kvi := \sqrt{a}$
 kv, kvi chiqarilsin
oxiri
tamonlansin



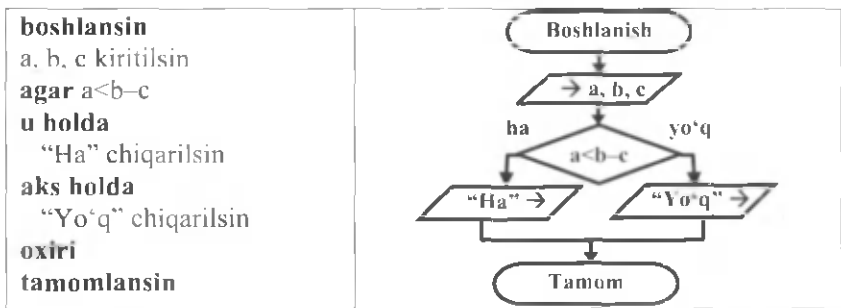
4-masala. Uchta a, b, c sonlar berilgan. Ular ichida manfiy bo'lmagan sonlar kvadrat ildizini hisoblash algoritmini tuzing.

Bu masala algoritmi ham tarmoqlanuvchi bo'lib, faqat bir holatda, ya'ni berilgan sonlar manfiy bo'lmaganda, ya'ni berilgan sonlar musbat yoki 0 bo'lganda, amallar bajarilishi talab etilmoqda. Shuning uchun tarmoqlanish tuzilmasini qisqa ko'rinishidan foydalanish qulay.



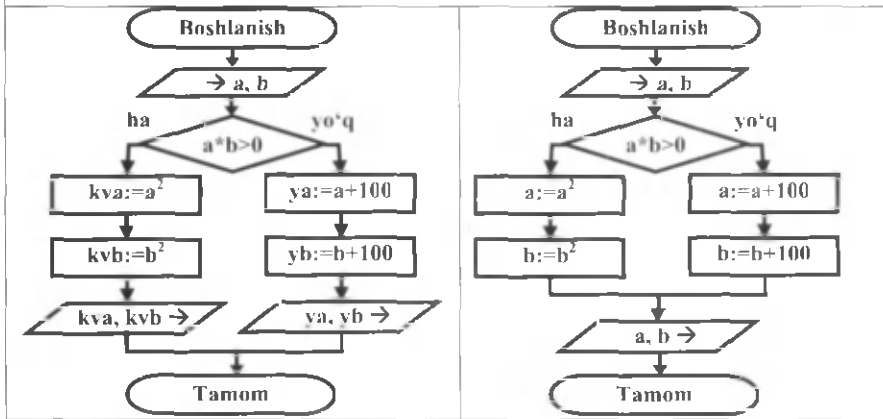
5-masala. Uchta a, b, c sonlar berilgan. $a < b - c$ shart bajarilsa "Ha", aks holda "Yo'q" deb javob chiqaruvchi algoritm tuzing.

Bu masala algoritmi ham tarmoqlanuvchi, lekin shart bajarilsa ham bajarilmasa ham amallar bajarilishi talab etilmoqda. Shuning uchun bu holda tarmoqlanish tuzilmasini to'liq shaklidan foydalanish qulay.



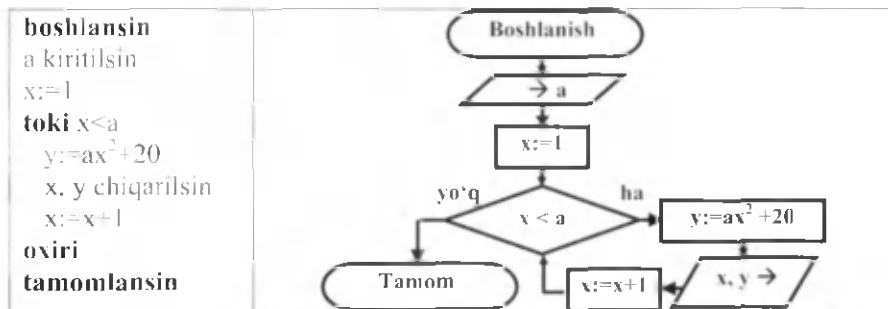
6-masala. a va b sonlar berilgan. Agar ularning ko‘paytmasi musbat bo‘lsa, ularning har birini kvadratini, aks holda ularning har biriga 100 ni qo‘shib chiqaruvchi algoritim tuzing.

Bu masala algoritimida ham tarmoqlanuvchi tuzilmani to‘liq shaklidan foydalanish qulay. Hisoblash jarayonida bajarilayotgan amallar qiymatini saqlab turish uchun yordamchi belgilar kiritish yoki bor belgilardan foydalanish mumkin.



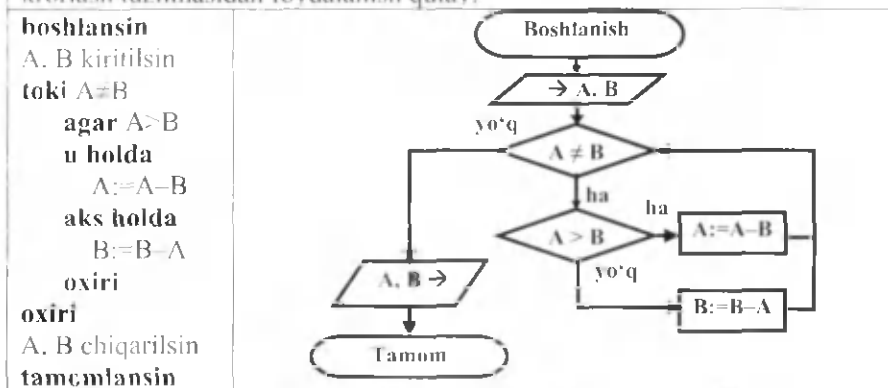
7-masala. Natural x ning qiymati berilgan a sonidan kichik bo‘lganda $y = ax^2 + 20$ funksiyaning qiymatlarini hisoblash algoritimini tuzing.

Demak, x ning a dan kichik har bir qiymatida y funksiyani hisoblash kerak. Lekin, takrorlanishlar soni avvaldan ma‘lum emas, shuning uchun shart bo‘yicha takrorlash tuzilmasidan foydalanish qulay.



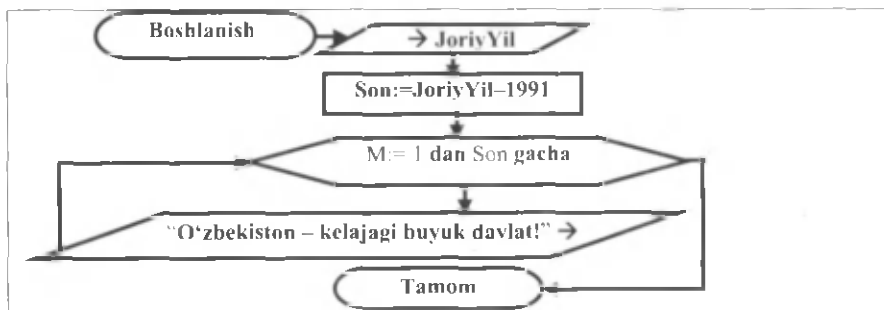
8–masala. Berilgan A va B musbat sonlarning qiymatlari teng bo'lguncha bu sonlarni kattasini kattasidan kichigini ayirmasi bilan almashtirib boruvchi algoritmi tuzing.

Bu masalada A va B sonlarni kattasidan kichigini ayirish orqali tenglashtirish talab qilingan. Kattasidan kichigini ayirish jarayoni necha marta takrorlanishi avvaldan noma'lum. Shuning uchun algoritmda shart bo'yicha takrorlash tuzilmasidan foydalanish qulay.



9–masala. “O‘zbekiston – kelajagi buyuk davlat!” iborasini shu o‘quv yilida Vatanimiz mustaqilligining nishonlangan sonicha yozdirish algoritmini tuzing.

Masaladan takrorlanishlar sonini aniqlash mumkin. Shuning uchun parametrlilik takrorlash tuzilmasidan foydalanish qulay.



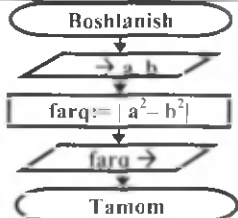
M-1. Tezligi V km/soat bo'lgan mashinaning T soatda bosib o'tgan yo'lini hisoblash algoritmini tuzing.

<p>Masala algoritmi chiziqli bo'ladi, chunki faqat qiymat berish, chiqarish va oddiy hisoblashdan iborat. Bosib o'tgan yo'lni formulasi: $S=V \cdot T$. Algoritmni blok-sxemasidagi bloklarni tagma-tag yoki yonma-yon yozish mumkin.</p>	<p>So'z orqali:</p> <p>hoshlansin V, T kiritilsin $S:=V \cdot T$ S chiqarilsin tamomlansin</p>	<pre> graph TD Start([Boshlanish]) --> Input[/→ V, T/] Input --> Process["S:=V *T"] Process --> Output[/S →/] Output --> End([Tamom]) </pre>
--	--	--

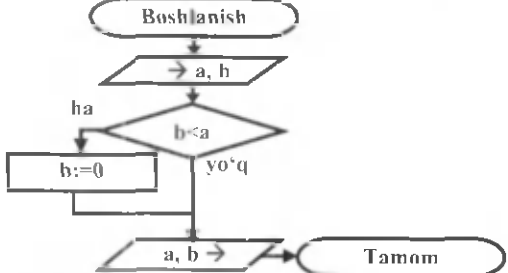
M-2. Radiuslari, mos ravishda, R_1, R_2, R_3 ga teng doiralarning umumiy yuzini kvadratini hisoblash algoritmini tuzing.

<p>Masala algoritmi chiziqli bo'ladi, chunki faqat qiymat berish, chiqarish va oddiy hisoblashdan iborat. Kerakli formulalar: $S_1=\pi R_1^2$; $S_2=\pi R_2^2$; $S_3=\pi R_3^2$; $S=S_1+S_2+S_3$ yoki yuzani bitta formula bilan ham ifodalash mumkin $S=\pi(R_1^2+R_2^2+R_3^2)$.</p>	<p>So'z orqali:</p> <p>hoshlansin R_1, R_2, R_3 kiritilsin $S:=\pi(R_1^2+R_2^2+R_3^2)$ S chiqarilsin tamomlansin</p>	<pre> graph TD Start([Boshlanish]) --> Input[/→ R1, R2, R3/] Input --> Process["S:= π *(R1^2+R2^2+R3^2)"] Process --> Output[/S →/] Output --> End([Tamom]) </pre>
---	--	--

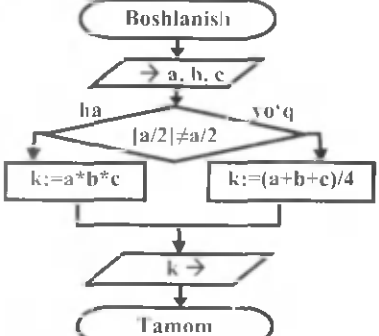
M-3. Tomonlari, mos ravishda, a va b bo'lgan kvadratlarning yuzalari farqini modulini topuvchi algoritm tuzing.

<p>Masala algoritmi chiziqli bo'ladi.</p> <p>Kerakli formulalar: $S1=a^2$; $S2=b^2$; $farq= S1-S2$. Farqni bitta formula bilan ham ifodalash mumkin: $farq= a^2-b^2$.</p>	<p>So'z orqali:</p> <p>boshlansin a, b kiritilsin $farq= a^2-b^2$ $farq$ chiqarilsin tamomlansin</p>	
--	---	--

M-4. Ikkita a va b sonlar berilgan. Agar b son a dan kichik bo'lsa, u holda b ni nol bilan almashtiruvchi, aks holda b o'zgarishsiz qoldiruvchi algoritmi tuzing.

<p>Bu masala algoritmi tarmoqlanuvchi bo'lib, faqat bir holatda, ya'ni $b < a$ shart bajarilganda amallar bajarilishi talab etilmoqda. Shuning uchun tarmoqlanish tuzilmasini qisqa ko'rinishidan foydalanish qulay.</p>	
<p>So'z orqali:</p> <p>boshlansin a, b kiritilsin agar $b < a$ u holda $b:=0$ oxiri a, b chiqarilsin tamomlansin</p>	

M-5. a, b va c sonlar berilgan. Agar a son toq bo'lsa uchchala sonni ko'paytmasini, aks holda uchchala sonni yig'indisining chorak qismini hisoblash algoritmini tuzing.

<p>Bu masala algoritmi ham tarmoqlanuvchi va shart bajarilsa ham bajarilmasa ham amallar bajarilishi talab etilmoqda. Shuning uchun bu holda tarmoqlanish tuzilmasini to'liq shakildan foydalanish qulay.</p>	<p>So'z orqali:</p> <p>boshlansin a, b, c kiritilsin agar $[a/2] \neq a/2$ u holda $k:=a*b*c$ aks holda $k:=(a+b+c)/4$ oxiri k chiqarilsin tamomlansin</p>	
---	---	--

M-6. a, b, c sonlar berilgan. $a+b+c < 0$ shart bajarilsa $y=a^2-b^2$ ni, aks holda $y=a^2+c^2$ ni hisoblash algoritmini tuzing.

<p>Bu masala algoritmi ham tarmoqlanuvchi va shart bajarilsa ham bajarilmasa ham amallar bajarilishi talab etilmoqda. Shuning uchun bu holda tarmoqlanish tuzilmasini to'liq shaklidan foydalanish qulay.</p>	<p>So'z orqali: boshlansin a, b, c kiritilsin agar $a+b+c < 0$ u holda $y := a^2 - b^2$ aks holda $y := a^2 + c^2$ oxiri y chiqarilsin tamomlansin</p>	
---	---	--

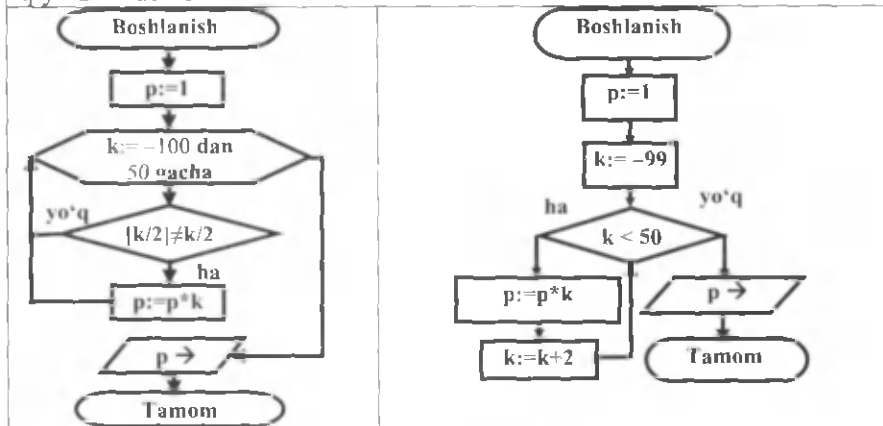
M-7. -100 dan 50 gacha bo'lgan sonlar oralig'idagi toq sonlar ko'paytmasini hisoblash algoritmini tuzing.

Masalani ikki xil usulda hal etamiz:

1-usul. Masalada takrorlanishlar soni avvaldan ma'lum, shuning uchun parametrli takrorlash tuzilmasidan foydalanish mumkin.

2-usul. Shart bo'yicha takrorlash tuzilmasidan foydalanish mumkin.

Ko'paytma hisoblanayotgani uchun ikkala usulda ham p ning boshlang'ich qiymati 1 deb olinadi.



11 – dars. Takrorlashga doir topshiriqlar

T-1. Kompyuterda masalalar yechishning birinchi uch bosqichini tomoni a ga teng bo'lgan kvadratga ichki chizilgan doiraning yuzini topish masalasi asosida ko'rsatib bering.

Masalaning qo'yilishi.

Berilgan:

kvadrat {ya'ni, barcha tomoni teng}

kvadrat tomoni a

kvadratga doira ichki chizilgan

Topish kerak: S_d {doira yuzi}

Masalaning modelini tuzish.

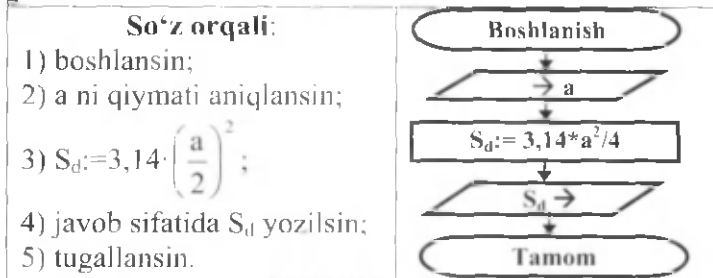
Chizмага asosan: $a=2 \cdot R$ yoki $R=\frac{a}{2}$

Doira yuzi formulasi: $S_d=\pi \cdot R^2$

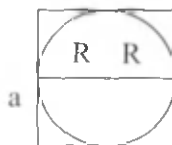
Bundan quyidagi model hosil bo'ladi:

$$S_d = \pi \cdot \left(\frac{a}{2}\right)^2$$

Algoritm tuzish.



Chizmasi



T-2. Kompyuterda masalalar yechishning birinchi uch bosqichini tomoni a ga teng bo'lgan kvadratga tashqi chizilgan aylananing uzunligini topish masalasi asosida ko'rsatib bering.

Masalaning qo'yilishi.

Berilgan:

kvadrat {ya'ni, barcha tomoni teng}

kvadrat tomoni a

kvadratga aylana tashqi chizilgan

Topish kerak: L {aylana uzunligi}

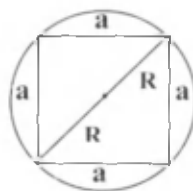
Masalaning modelini tuzish.

Chizмага va Pifagor teoremasiga asosan: $a^2+a^2=(2 \cdot R)^2$ yoki

$$R = \frac{a}{\sqrt{2}}$$

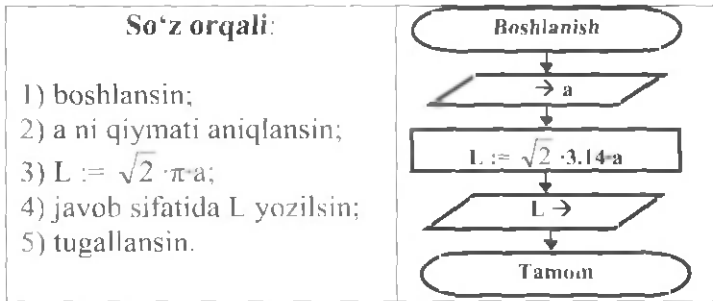
Aylana uzunligi formulasi: $L=2 \cdot \pi \cdot R$

Chizmasi



Bundan shunday model hosil bo'ladi: $L = \sqrt{2} \cdot \pi \cdot a$.

Algoritm tuzish.



T-3. Kompyuterda masalalar yechishning birinchi uch bosqichini quyidagi masalani hal etish asosida ko'rsatib bering: 50 litrlik idishda 5 kilogramm osh tuzi qo'shilgan 20 litr aralashma bor. Agar idishga yana 10 litr suv qo'shilsa aralashmadagi tuz miqdorini foiz hisobida toping.

Masalaning qo'yilishi.

Berilgan:

20 litr aralashmada 5 kg tuz bor
 aralashmaga 10 litr suv qo'shildi

Topish kerak: tuz miqdori {foiz hisobida}

Chizmasi



Masalaning modelini tuzish.

1 litr suv 1 kg bo'lgani uchun, 20 litr aralashmada 20 kg – 5 kg =
 = 15 kg = 15 litr suv bor

10 litr suv qo'shilganda idishda 25 litr suv va 5 kg tuz bo'ladi

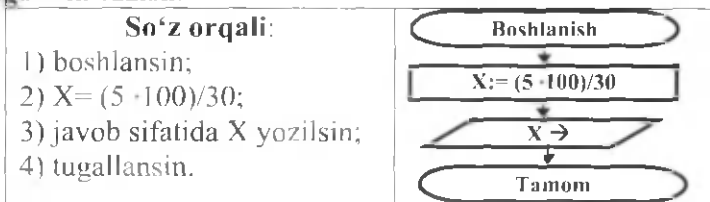
30 kg aralashma – 100 %

5 kg tuz – X %

Proporsiya yordamida quyidagiga ega bo'lamiz:

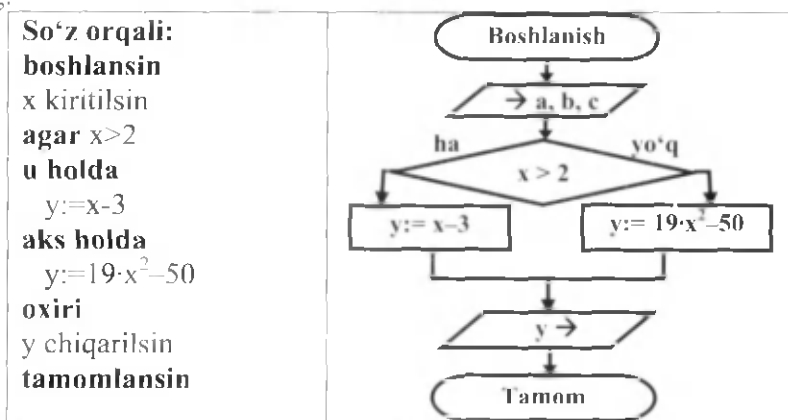
$X = (5 \text{ kg} \cdot 100\%) : 30 \text{ kg}$

Algoritm tuzish.



T-4. Berilgan x da $y = \begin{cases} 19 \cdot x^2 - 50, & \text{agar } x \leq 2 \\ x - 3, & \text{agar } x > 2 \end{cases}$ funksiyaning

qiymatini hisoblash algoritmini so'zlar va blok-sxema yordamida tuzing.

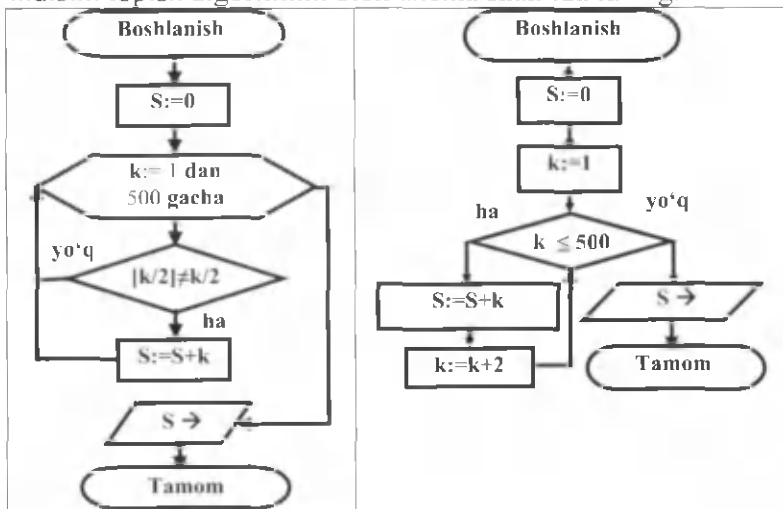


T-5. 1 dan 100 gacha bo'lgan sonlar ichidan 11 ga karrali sonlarning P ko'paytmasini topish algoritmini so'zlar yordamida tuzing.

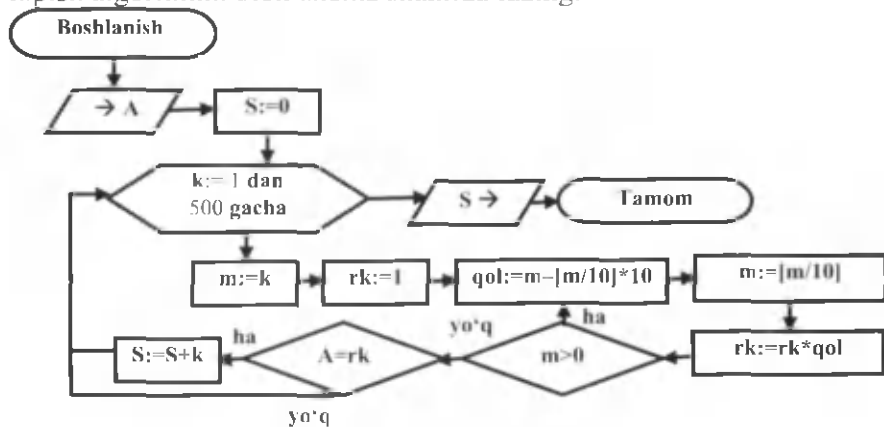
1-usul:	2-usul	3-usul
boshlansin	boshlansin	boshlansin
$P := 1$	$P := 1$	$P := 1$
$K := 1$	$K := 1$ dan 100 gacha	$K := 1$
toki $K \leq 100$	agar $[K/11] = K/11$	toki $K \leq 100$
u holda	u holda	$P := P * K$
$P := P * K$	$P := P * K$	$K := K + 11$
oxiri	oxiri	oxiri
$K := K + 1$	P chiqarilsin	P chiqarilsin
oxiri	tamomlansin	tamomlansin

P chiqarilsin
tamomlansin

T-6. 1 dan 500 gacha oraliqda bo'lgan toq sonlarning S yig'indisini topish algoritmini blok-sxema shaklida tuzing.



T-7. A natural son berilgan. 1 dan 1000 gacha bo'lgan sonlar ichida raqamlari ko'paytmasi A songa teng bo'lgan sonlar S yig'indisini topish algoritmini blok-sxema shaklida tuzing.



12 – dars. Dastur va dasturlash tillari mavzusiga

T-4. Dasturlash tillari elektron hisoblash mashinalarining turlariga bogʻliq boʻladimi? Javobingizni asoslang.

Albatta bogʻliq boʻladi. Har bir dasturlash tili yoki uning biror naqli (versiyasi) kompyuter imkoniyatlariga talab qoʻyadi, masalan: razryadiga, tezkor xotira hajmiga, adreslar sohasiga va hokazo. Hozirgi kunda ishlab chiqarilayotgan kompyuterlar uchun ularga mos dasturlash tillarining naqlari ishlab chiqarilmoqda.

13 – dars. Turbo Pascal 7.0 integrallashgan muhiti mavzusiga

M-a) Oʻzbekiston Respublikasi madhiyasining birinchi toʻrtligini kiriting;

Mashqning bajarilishi: Turbo Paskal integrallashgan muhiti ishga tushiriladi. **File** menyusidan **New** buyrugʻi tanlanadi. Hosil boʻlgan dastur matn muharriri oynasining matn maydoniga Bloknot matn muharriridagi kabi Oʻzbekiston Respublikasi madhiyasining birinchi toʻrtligi kiritiladi.

M-b) Yangi oyna oching va unga Respublikamiz madhiyasining ikkinchi toʻrtligini kiriting;

Mashqning bajarilishi: **File** menyusidan **New** buyrugʻi tanlanadi. Hosil boʻlgan ikkinchi dastur matn muharriri oynasining matn maydoniga Oʻzbekiston Respublikasi madhiyasining ikkinchi toʻrtligini kiritamiz.

M-d) 2-oynadagi matnning (madhiyaning 2-toʻrtligi) nusxasini olib, 1-oynadagi madhiyaning 1-toʻrtligini davomiga joylashtiring.



Mashqning bajarilishi: Faol oynadagi Oʻzbekiston Respublikasi madhiyasining ikkinchi toʻrtligi matnini bosh qismiga yurgichni joylashtiramiz. Soʻng, **Shift** klavishini bosib turgan holda zaruratga qarab ↓ yoki → yoʻnalish klavishini yurgich matn oxiriga kelgunga qadar bosib turamiz. Natijada matn (rangli fonda aks etadi) belgilab olinadi. Endi belgilangan qismni xotira buferiga olish uchun **Edit** menyusining **Copy** buyrugʻi tanlanadi yoki **Ctrl+Ins** tezkor klavishlar juftligi bosiladi. **F6** klavishini bosish orqali birinchi oyna faollashtiriladi. Yurgichni faol oynadagi matn maydonini kerakli joyiga joylashtiriladi.

Xotiradagi matn nusxasini joylashtirish uchun **Edit** menyusining **Paste** buyrug'i tanlanadi yoki **Shift+Ins** tezkor klavishlar juftligi bosiladi. Natijada nusxa madhiyaning birinchi to'rtligi davomiga joylashadi.

M-e) 1-oynadagi matnni "Madhiya.txt" nomi bilan saqlang;

Mashqning bajarilishi: O'zbekiston Respublikasi madhiyasining birinchi va ikkinchi to'rtligi yozilgan birinchi oynadagi matnni saqlash uchun **File** menyusining **Save** yoki **Save as** buyruqlaridan biri tanlanadi yoki **F2** tezkor klavish bosiladi. Natijada saqlash muloqot oynasi hosil bo'ladi. Ushbu oynaning **Save file as** darchasining yurgich turgan joyiga Madhiya.txt yozilgach **OK** tugmasi tanlanadi yoki **Enter** klavishi bosiladi.

M-f) 2-oynani saqlamasdan yoping.

Mashqning bajarilishi: Yuqoridagi vazifalarni bajarib bo'lgandan so'ng **F6** klavishini bosish orqali ikkinchi oyna faollashtiriladi. Faol oynani yopish uchun yopish  to'rtburchagini tanlash yoki **Alt+F3** tezkor klavishlarini bosish mumkin. Natijada so'rov oynasi hosil bo'ladi. Ushbu oynadan  tugmasi tanlansa ma'lumotlar saqlanmasdan yopiladi.

14 – dars. Paskal dasturlash tili alifbosi va tuzilishi mavzusiga

M-1. O'ng ustundagi belgilarni chap ustundagi to'plamiga mosini aniqlang.

Yechim:

Mantiqiy amallar
Munosabat belgilari
Maxsus belgilar

NOT, OR
<, <=, >, >=
%, \$, @, &, (,), {, }, [,]

M-2. Chap ustundagi iboralarga o'ng ustundagi tavsiflarni mosini aniqlang.

Yechim:

Modularni qo'llash
Nishonlarni tavsiflash
Dasturni nomlash
O'zgaruvchini tavsiflash

Uses Crt;
Label 19;
Program 9 sinf;
Var a21: integer;

M-3. Identifikator nomlarini “To‘g‘ri yozilgan” va “Noto‘g‘ri yozilgan” guruhlariga ajrating va sababini izohlang.

a	1kun	MeningBirinchiDasturim	BMA
Chegara#4	Keyingi yil	Kun_21_iyul_1963	and

Yechim:

To‘g‘ri yozilgan			
a	MeningBirinchiDasturim	BMA	Kun_21_iyul_1963

Noto‘g‘ri yozilgan	
1kun	Identifikator nomi lotin harfi yoki tagchiziq belgisidan boshlanadi, raqamdan boshlanishi mumkin emas
Chegara#4	Identifikator nomida lotin harfi, tagchiziq belgisi va raqam ishtirok etadi, # belgisi ishtirok etishi mumkin emas
Keyingi yil	Identifikator nomida probel belgisi ishtirok etishi mumkin emas
and	Identifikator nomida Paskal tilining zaxira so‘zi ishtirok etishi mumkin emas

15 – dars. O‘zgarmas va o‘zgaruvchi miqdorlar mavzusiga

M-1. Quyidagi o‘zgamaslarni turlarini aytib bering.

- a) ‘7!‘; ‘informatika‘; ‘-987378‘; ‘BMA‘; – bu o‘zgarmaslar satrli o‘zgarmaslar turiga kiradi, chunki belgilar apostrof ichida yozilgan va bittadan ortiq.
- b) ‘,‘; ‘u‘; ‘0‘; ‘ ‘; – bu o‘zgarmaslar belgili o‘zgarmaslar turiga kiradi, chunki o‘zgarmaslar apostrof ichiga olingan va bitta belgidan iborat.
- d) 99; -200; 101; 87; – bu o‘zgarmaslar sonli o‘zgarmaslar turining butun turiga kiradi.
- e) 0.01; 8.909; 132.001; 878887.1; – bu o‘zgarmaslar qo‘zg‘almas nuqtali sonli o‘zgarmaslar turiga kiradi, chunki ular o‘nli kasr ko‘rinishida berilgan.
- f) 0.07 E-3; -9.8 E6; – bu o‘zgarmaslar qo‘zg‘aluvchi nuqtali sonli o‘zgarmaslar turiga kiradi, chunki ular eksponensial ko‘rinishida berilgan (E yoki e yordamida ifodalangan).
- g) True; False; – bu o‘zgarmaslar mantiqiy o‘zgarmaslar turiga kiradi. Bu yerda, mantiqiy o‘zgarmaslar True (rost), False (Yolg‘on) qiymatli.

M-2. Quyidagi o'zgaruvchilarni turlarini aniqlang va izohlang.

a) men : Boolean; – **men** nomli o'zgaruvchini mantiqiy o'zgaruvchi turiga mansubligi **Boolean** maxsus so'z orqali ta'kidlanmoqda, u True yoki False qiymatlarni qabul qilishi mumkin.

b) bahodir : String[7]; – **bahodir** nomli o'zgaruvchini 7 ta belgili satrli o'zgaruvchi turiga mansubligi **String[7]** maxsus so'z orqali ta'kidlanmoqda, demak, bahodir nomli o'zgaruvchiga xotiradan 7 bayt joy ajratiladi va u 7 ta belgidan oshmaydigan belgili o'zgaruvchilar qiymatini qabul qilishi mumkin.

d) hayot : Real; – **hayot** nomli o'zgaruvchini haqiqiy sonli o'zgaruvchi turiga mansubligi **Real** maxsus so'z orqali ta'kidlanmoqda, demak, hayot nomli o'zgaruvchi $-2,9 \cdot 10^{30} \dots 1,7 \cdot 10^{38}$ oraliqdagi sonli o'zgaruvchilar qiymatini qabul qilishi mumkin.

e) son : char; – **son** nomli o'zgaruvchini qiymati belgili o'zgaruvchi turiga **Char** maxsus so'z orqali ta'kidlanmoqda, demak u belgili o'zgaruvchilar qiymatini qabul qilishi mumkin.

f) baxt : Integer; – **baxt** nomli o'zgaruvchini butun sonli o'zgaruvchi turiga mansubligi **Integer** maxsus so'z orqali ta'kidlanmoqda, demak, baxt nomli o'zgaruvchi $-32\ 768 \dots 32\ 767$ oraliqdagi butun sonli o'zgaruvchilar qiymatini qabul qilishi mumkin.

g) ser : Single; – **ser** nomli o'zgaruvchini haqiqiy sonli o'zgaruvchi turiga mansubligi **Single** maxsus so'z orqali ta'kidlanmoqda, demak, ser nomli o'zgaruvchi $-1,5 \cdot 10^{45} \dots 3,4 \cdot 10^{38}$ oraliqdagi haqiqiy sonli o'zgaruvchilar qiymatini qabul qilishi mumkin.

16 – dars. O'zgaruvchi va o'zgaruvchi miqdorlar mavzusini takrorlash mavzusiga

M-1. Quyidagi o'zgaruvchilarni turlarini aytib bering.

a) -9.22 E-2; 0.01 E+5; 1.11 E-4;	–	qo'zg'aluvchan nuqtali
b) 21; 21; 7; 7; 19; 19; 63; 63; -	–	butun sonli o'zgaruvchilar
d) true; true; true; false; false; false;	–	mantiqiy o'zgaruvchilar
e) '555'; 'aar'; 'mmr'; 'bbj'; 'aga';	–	satrli o'zgaruvchilar
f) 'Muqaddas'; 'Vatan'; 'Mustaqil';	–	satrli o'zgaruvchilar
g) 'i'; 'n'; 's'; 'o'; 'n'; 'i'; 'y'; 'a'; 't';	–	belgili o'zgaruvchilar

M-2. O'zgaruvchilarga nom berib turiga mos tavsiflang.

a) belgili b:char; aa21:char;	b) haqiqiy h:real; ha:real;	d) mantiqiy m:boolean; g20: boolean;	e) satrli s:string; b19:string;
f) belgilari 7 tadan oshmaydigan satrli satr:string[7]; ss: string[7];			

M-3. Har bir bandda bitta o'zgaruvchining barcha qiymati berilgan yoki xususiyati ifodalangan. Shu o'zgaruvchilarga nom bering va tavsiflang.

a) -5; 0; 7; 58; -15; 9. b: ShortInt; m:integer;	b) 'Xalq'; 'Vatan'; 'Ona'. a:string[5]; g: string;
d) 7.21; 4.2; 50.1902; -1.23. h:Single; haq:string;	e) birinchi 7 ta tub son. t:byte; tub:word;
f) true; true; false; true; false. man:boolean; s: boolean;	g) '000'; '001'; '002'; '003'. aa:string[3]; ag: string;
h) alifbo harflari. al:char; bel:string;	i) 'Yuksak'; 'ma'naviyat', 'yengilmas'; 'kuch'. ss:string[10]; kia:string;

M-4. Barcha qiymati berilgan butun o'zgaruvchilarga nom berib tavsiflang. O'zgaruvchini turini tanlashda xotiradan kam joy olishiga erishing.

a) -4; 0; -4; 8; 12; sho: ShortInt; bu holda sho xotiradan 1 bayt joy egallaydi, qiymati chegarasi -127..128; 1 bayt joy egallaydigan byte turini olib bo'lmaydi, chunki bu turda o'zgaruvchi qiymati manfiy bo'la olmaydi	b) 1; 16; 256; 4096; 65536; w2: word; bu holda w2 xotiradan 2 bayt joy egallaydi, qiymati chegarasi 0 .. 65536; xotiradan 2 bayt joy egallaydigan integer turini olib bo'lmaydi, chunki bu turda qiymat chegarasi -32768 .. 32767
d) 0; 2; 4; 6; 8; 10; juft: byte; bu holda juft xotiradan 1 bayt joy egallaydi, qiymati chegarasi 0..255; ikki: ShortInt; bu holda ikki xotiradan 1 bayt joy egallaydi, qiymati chegarasi -127..128; boshqa butun turlarda xotiradan kamida 2 bayt joy oli-	e) 29350; -2; 8000; 250; son1: integer; bu holda son1 xotiradan 2 bayt joy egallaydi, qiymati chegarasi -32768 .. 32767; 1 bayt joy egallaydigan byte yoki shortint turlarni olib bo'lmaydi, chunki bu sonlar turlarni qiymat chegarasidan chiqib

nadi	ketadi
f) 5; -32767; 46; 0; 32767; son2: integer; bu holda son2 xotiradan 2 bayt joy egallaydi, qiymati chegarasi -32768..32767; 1 bayt joy egallaydigan byte yoki shortint turlarni olib bo'lmaydi, chunki bu sonlar turlarni qiymat chegarasidan chiqib ketadi	g) 200000; 2000; -20; 99999; son3: longint; bu holda son3 xotiradan 4 bayt joy egallaydi, qiymati chegarasi -2147483648..2147483647; 2 bayt joy egallaydigan word yoki integer turlarni olib bo'lmaydi, chunki bu sonlar turlarni qiymat chegarasidan chiqib ketadi

M-5. Berilgan va amallar natijasida hosil bo'ladigan o'zgaruvchilarni tavsiflang.

a) f: butun; g: haqiqiy; $d=f*g+f+g$; Arifmetikadan ma'lumki, butun songa haqiqiy son qo'shilsa haqiqiy son hosil bo'ladi, masalan $5+0,3=5,3$. Shuning uchun d o'zgaruvchi turi haqiqiy bo'ladi. U holda quyidagi o'rinli: Var f: integer; g, d: real;	b) d: butun; n: butun; $k=d+2*n$; Arifmetikadan ma'lumki, butun songa butun son qo'shilsa yoki butun songa butun son ko'paytirilsa butun son hosil bo'ladi. Shuning uchun k o'zgaruvchi turi butun bo'ladi. U holda quyidagi o'rinli: Var d, b, k: integer;
d) s: mantiqiy; e: mantiqiy; $q:=not(s \text{ or } e)$; Har qanday mantiqiy amal yana mantiqiy natijaga olib keladi. Shuning uchun q o'zgaruvchi turi mantiqiy bo'ladi. U holda quyidagi o'rinli: Var s, e, q: boolean;	e) k: toq; m: juft; $vvv:=k+m/2$; Paskalda bo'lish amali natijasi qanday bo'lishidan qat'iy nazar haqiqiy turga mansub hisoblanadi. Shuning uchun vvv o'zgaruvchi turi haqiqiy bo'ladi. Toq va juft sonlar butundir. U holda quyidagi o'rinli: Var k, m: integer; vvv: real;

17 – dars. Jadval ko‘rinishidagi miqdorlar mavzusiga

M-1. Quyida keltirilgan ketma-ketliklar qanday o‘lchovli massivlarni ifodalashini va nechta elementdan iboratligini aniqlang.

<p>a) A[0], A[1], A[2], A[3], ... , A[99]; [va] qavslari ichida faqat bitta son yozilgan, demak massiv bir o‘lchovli. Elementlari sonini darslikdagi formula yordamida hisoblaymiz: $99-0+1=100$ ta.</p>	<p>b) B[0.0], B[0.1], B[0.2], ... , B[3.5]; [va] qavslari ichida vergul bilan ajratilgan ikkita son yozilgan, demak massiv ikki o‘lchovli. Elementlari sonini darslikdagi formula yordamida hisoblaymiz: $(3-0+1) \cdot (5-0+1)=4 \cdot 6=24$ ta.</p>
<p>d) M[0.0.0], M[0.0.1], ... , M[1.1.1]. [va] qavslari ichida vergul bilan ajratilgan uchta son yozilgan, demak massiv uch o‘lchovli. Elementlari sonini hisoblaymiz: $(1-0+1) \cdot (1-0+1) \cdot (1-0+1)=2 \cdot 2 \cdot 2=8$ ta.</p>	<p>e) G[-22.3], G[-22.4], G[-22.5], ..., G[-20.5]. [va] qavslari ichida vergul bilan ajratilgan ikkita son yozilgan, demak massiv ikki o‘lchovli. Elementlari sonini darslikdagi formula yordamida hisoblaymiz: $(-20-(-22)+1) \cdot (5-3+1)=3 \cdot 3=9$ ta.</p>

M-2. Butun turdagi, chiziqli 100 ta elementli jadval qaysi javobda to‘g‘ri tavsiflangan?

<p>a) var B: array [1..100] of real; Tavsif bo‘yicha B massiv bir o‘lchovli, elementlari soni $100-1+1=100$ ta, elementlari real, ya‘ni haqiqiy turda. Xato javob.</p>	<p>b) var M: array [1..100] of char; Tavsif bo‘yicha M massiv bir o‘lchovli, elementlari soni $100-1+1=100$ ta, elementlari char, ya‘ni belgili turda. Xato javob.</p>
<p>d) var A: array [0..99] of string; Tavsif bo‘yicha A massiv bir o‘lchovli, elementlari soni $99-0+1=100$ ta, elementlari string, ya‘ni satrli turda. Xato javob.</p>	<p>e) var G: array [5..104] of integer; Tavsif bo‘yicha G massiv bir o‘lchovli, elementlari soni $104-5+1=100$ ta, elementlari integer, ya‘ni butun turda. To‘g‘ri javob.</p>
<p>f) var M: array [1..10,1..10] of integer; Tavsif bo‘yicha M massiv ikki</p>	<p>g) var A: array [5..104] of string; Tavsif bo‘yicha A massiv bir o‘lchovli, elementlari soni 104-</p>

o'lovli, elementlari soni $(10-1+1)\cdot(10-1+1)=100$ ta, elementlari integer, ya'ni butun turda. Xato javob.	$5+1=100$ ta, elementlari string, ya'ni satrli turda. Xato javob.
---	---

M-3. 2-mashqdagi massivlarni turini, o'lovini va elementlari sonini aniqlang.

Avvalgi mashqda bajarildi.

M-4. Bittadan bir o'lovli butun va belgili jadvallarni hamda ikki o'lovli haqiqiy turdagi jadvalni tavsiflang.

Bir o'lovli butun turdagi jadval tavsifi: var B: array [10..100] of integer;
Bir o'lovli belgili turdagi jadval tavsifi: var A: array [7..21] of string;
Ikki o'lovli haqiqiy turdagi jadval tavsifi: var M: array [1..23,10..10] of real;

18 – dars. Jadval ko'inishidagi miqdorlar mavzusini takrorlash mavzusiga

M-1. To'g'ri to'rtburchakli haqiqiy turdagi sakkiz satrli va o'n bir ustunli F jadval qanday tavsiflanishini aniqlang.

a) var A: array [8..11] of real; Tavsif bo'yicha A massiv bir o'lovli, elementlari soni $11-8+1=4$ ta, elementlari real, ya'ni haqiqiy turda. Xato javob.	b) var B: array [1..8,1..11] of integer; Tavsif bo'yicha B massiv ikki o'lovli, elementlari soni $(8-1+1)\cdot(11-1+1)=88$ ta, elementlari butun, ya'ni haqiqiy turda. Xato javob.
d) var D: array [8..11,8..11] of real; Tavsif bo'yicha D massiv ikki o'lovli, elementlari soni $(11-8+1)\cdot(11-8+1)=16$ ta, elementlari real, ya'ni haqiqiy turda. Xato javob.	e) var M: array [0..8,0..10] of integer; Tavsif bo'yicha M massiv ikki o'lovli, elementlari soni $(8-0+1)\cdot(10-0+1)=99$ ta, elementlari integer, ya'ni butun turda. Xato javob.

f) var F: array [0..7,0..10] of real; Tavsif bo'yicha F massiv bir o'lchovli, elementlari soni $(7-0+1) \cdot (10-0+1) = 88$ ta, elementlari real, ya'ni haqiqiy turda. To'g'ri javob.	g) var F: array [0..7,0..10] of char; Tavsif bo'yicha F massiv bir o'lchovli, elementlari soni $(7-0+1) \cdot (10-0+1) = 88$ ta, elementlari char, ya'ni belgili turda. Xato javob.
---	--

M-2. 1-mashqda tavsiflangan massivlarni turini, o'lchovini va elementlari sonini aniqlang.

Avvalgi mashqda bajarildi.

M-3. Pifagor jadvalini tuzing. Jadval elementlarini tahlil qiling. Massivga nom berib tavsiflang.

Pifagor jadvali quyidagi ko'rinishdagi karra jadvalidir:

Pif	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15	18	21	24	27
4	4	8	12	16	20	24	28	32	36
5	5	10	15	20	25	30	35	40	45
6	6	12	18	24	30	36	42	48	54
7	7	14	21	28	35	42	49	56	63
8	8	16	24	32	40	48	56	64	72
9	9	18	27	36	45	54	63	72	81

Jadvalga **Pif** deb nom beramiz. Jadval ikki o'lchovli. Jadval elementlari satr bo'yicha ham 1 dan 9 gacha, ustun bo'yicha 1 dan 9 gacha tartiblangan. Jadval elementlari soni $(9-1+1) \cdot (9-1+1) = 81$ taga teng. Jadval elementlari 255 dan oshmaydigan butun musbat sonlar. Shuning uchun quyidagicha tavsiflaymiz:

Var Pif: array [1..9, 1..9] of byte;

M-4. Oilangiz a'zolarining ismi, tug'ilgan yili va ma'lumoti haqida jadval tuzing. Massivni tavsiflang va elementlarini tahlil qiling.

Quyidagi misoldagi kabi jadval tuziladi. Ism, tug'ilgan yil va ma'lumot ixtiyoriy tanlanishi mumkin:

Oila	1	2	3
1	Abdulla	1947	Oliy ma'lumotli
2	Bahodir	1950	Oliy ma'lumotli
3	Sayfulla	1950	Oliy ma'lumotli
4	Abdug'affor	1957	Oliy ma'lumotli

Jadvalga **Oila** deb nom beramiz. Jadval ikki o'lchovli. Jadval elementlari satr bo'yicha 1 dan 4 gacha, ustun bo'yicha 1 dan 3 gacha tartiblangan. Jadval elementlari soni $(4-1+1) \cdot (3-1+1) = 12$ ta. Jadval elementlari ham sonli ham satrli bo'lgani uchun **string** turni tanlaymiz. Agar xotiradan egallaydigan joy hajmini qisqartirmoqchi bo'lsak, u holda belgilari eng ko'p elementni aniqlaymiz. Bunday element 3-ustundagi 15 ta belgili "Oliy ma'lumotli" bo'lgani uchun massivni quyidagicha tavsiflaymiz:

Var Oila: array [1..4, 1..3] of byte;

M-5. Quyidagi **integer** turdagi M chiziqli massiv elementlariga bo'sh katakda turiga mos qiymat bering. Massivni tavsiflang.

M[-7]	M[-6]	M[-5]	M[-4]	M[-3]	M[-2]	M[-1]

Yechim: Topshiriq shartiga ko'ra, M massiv elementlariga faqatgina **integer** turdagi butun son kiritiladi. Bu sonlarni qiymati chegarasi -32758...32757 oraliqda. Demak massiv elementlariga shu oraliqdagi ixtiyoriy butun sonni yozish mumkin. Tavsifi:

M: array [-7..-1] of integer;

M[-7]	M[-6]	M[-5]	M[-4]	M[-3]	M[-2]	M[-1]
-32758	32757	-1963	1963	1950	-23	21

M-6. Quyidagi **char** turdagi B chiziqli massiv elementlariga bo'sh katakda turiga mos qiymat bering. Massivni tavsiflang va elementlarini ustun shaklida yozing. Massivni ikki o'lchovli massiv ko'rinishiga o'tkazib qayta tavsiflang.

B[9]	B[10]	B[11]	B[12]	B[13]	B[14]	B[15]	B[16]

Yechim: Topshiriq shartiga ko'ra, B massiv elementlariga faqatgina **char** turdagi, ya'ni bitta belgi kiritiladi. Tavsifi:

B: array [9..16] of char;

B[9]	B[10]	B[11]	B[12]	B[13]	B[14]	B[15]	B[16]
A	B	a	b	M	!	+	{

M-7. Quyidagi **string** turdagi A massiv elementlariga bo'sh katakda turiga mos qiymat bering. Jadvalni tavsiflang. Massivga boshqa tartib raqamlari berib G nom bilan qayta tavsiflang.

A[7,1]	A[7,2]	A[7,3]	A[7,4]
A[8,1]	A[8,2]	A[8,3]	A[8,4]

Yechim: 1) Topshiriq shartiga ko'ra, ikki o'lehovli A massiv elementlari **string** turda, ya'ni har bir elementdagi belgilar soni 255 tadan oshmaydi, element bo'sh bo'lishi ham mumkin (quyidagi A[7,4] kabi). Jadval elementlari satr bo'yicha 7 dan 8 gacha, ustun bo'yicha 1 dan 4 gacha tartiblangan, ya'ni 2 ta satr va 4 ta ustundan iborat. Tavsifi:

A: array [7..8, 1..4] of string;

A[7,1]	A[7,2]	A[7,3]	A[7,4]
Informatika	Hayot go'zal	Kelajak yoshlar qo'lida	AAR
A[8,1]	A[8,2]	A[8,3]	A[8,4]
o	??????	---999.25.00;5+++	KjkjhkJ

2) Jadval nomini G deb olamiz. Elementlar tartib raqamlarini satr bo'yicha 1, 2 (chunki 2 ta satr bor), ustun bo'yicha 3, 4, 5, 6 (chunki 4 ta ustun bor) deb o'zgartiramiz. Tavsifi:

var G: array[1..2,3..6] of string;

19 – dars. Standart funksiyalar va protseduralar, algebraik ifodalar mavzusiga

T-6. $\text{Trunc}(4.7) = \text{Round}(4.7)$ o'rinlimi? Javobingizni izohlang.

Topshiriqni bajarilishi. O'rinli emas. Chunki Trunc funksiyasi o'zgaruvchi turini butun turga o'zgaruvchi qiymatini butunlashtirib (kasr qismini tashlab yuborib) o'tkazadi: $\text{Trunc}(4.7) = 4$. Round funksiyasi esa o'zgaruvchi turini butun turga o'zgaruvchi qiymatini yaxlitlab (kasr qismidagi 5 dan kichik raqamni butunlay tashlab yuborib, 5 va undan katta raqamlarni oldingi xonadagi raqamni oshirish sharti bilan tashlab yuborib) o'tkazadi: $\text{Round}(4.7) = 5$.

T-7. $\sin x - c$ ko'rinishidagi yozuv Paskalda nima uchun xato hisoblanadi?

Topshiriqni bajarilishi. Paskal dasturlash tilida identifikator nomi-da probel ishlatib bo'lmaydi, shuning uchun **sin x** xato yozilgan identifikator nomi hisoblanadi. Agar $\sin x$ funksiyasi Paskal tilida yozilishi kerak bo'lsa, u holda argument qavsga olinishi shart: $\sin(x)$ kabi.

T-8. 2^*_{-v} ko'rinishidagi yozuv Paskalda to'g'ri yozilganmi? Javohingizni izohlang.

Topshiriqni bajarilishi. To'g'ri yozilgan. Chunki, Paskal tilida ifodadagi ******* belgisidan keyin yozilgan ****** belgisi v o'zgaruvchini ishorasi deb olinadi. Masalan: agar $v:=2$ bo'lsa, $2^*_{-v} = -2^*2 = -4$; agar $v:=-7$ bo'lsa, u holda $2^*_{-v} = 2^*_{-(-7)} = 2^*7 = 14$.

T-9. $\text{sqr}(\text{abs}(x + \sin(x)) - \pi)$ ifodada amallar bajarilish tartibini izohlang.

Topshiriqni bajarilishi. Qavslarga e'tibor berib, birinchi navbatda $\sin(x)$ funksiyasi, ikkinchi navbatda $x + \sin(x)$ yig'indi, uchinchi navbatda $\text{abs}(x + \sin(x))$ funksiyasi, to'rtinchi navbatda $\text{abs}(x + \sin(x)) - \pi$ ayirma, va nihoyat, beshinchi navbatda $\text{sqr}(\text{abs}(x + \sin(x)) - \pi)$ hisoblanishini aniqlaymiz.

M-1. Mavzuning 1-misolidagi qo'zg'aluvchi nuqtali sonlarni qo'zg'almas nuqtali sonlarga o'tkazing.

Yechim. Mavzuning birinchi misolida qo'zg'aluvchi nuqtali sonlarni mos funksiyalari bilan ko'chirib olib qo'zg'almas nuqtali sonlarga o'tkazamiz:

$\text{abs}(-4.9)$	4.9000000000e+00	4.9
$\text{abs}(4.9)$	4.9000000000e+00	4.9
$\text{sqr}(2.5)$	6.2500000000e+00	6.25
$\text{Sqrt}(16)$	4.0000000000e+00	4
$\text{Sqr}(0.0)$	0.0000000000e+00	0
$\text{Sqrt}(0.16)$	4.0000000000e-01	0.4
$\text{Sin}(0)$	0.0000000000e+00	0
$\text{Sin}(1)$	8.4147098481e-01	0.84147098481
$\text{Int}(5.3)$	5.0000000000e+00	5
$\text{Int}(5)$	5.0000000000e+00	5
$\text{Int}(-5.3)$	-5.0000000000e+00	-5
$\text{frac}(5.3)$	3.0000000000e-01	0.3
$\text{frac}(-5.3)$	-3.0000000000e-01	-0.3
$\text{frac}(5)$	0.0000000000e+00	0

M-2. Quyidagi algebraik ifodalarni Paskal dasturlash tilida yozing.

a) $ax+b$ Javob: $a*x+b$	b) $xyz-23$ Javob: $x*y*z-23$
d) ax^2+bx+c Javob: $a*x*x+b*x+c$ yoki $a*\text{sqr}(x)+b*x+c$	
f) $\frac{a+5}{2b}$ Javob: $(a+b)/(2*b)$	g) $8(a+b^2c)$ Javob: $8*(a+b*b*c)$ yoki $8*(a+\text{sqr}(b)*c)$
c) $a^4x^3-(1-y^2)^2$ Javob: $a*a*a*a*x*x*x-(1-y*y)*(1-y*y)$ yoki $\text{sqr}(a*a)*\text{sqr}(x)*x-\text{sqr}(1-\text{sqr}(y))$ yoki $\text{sqr}(\text{sqr}(a))*\text{sqr}(x)*x-\text{sqr}(1-\text{sqr}(y))$	
a^b darajagani exp(b*Ln(a)) ko'rinishda yozishda ehtiyot bo'lish lozim. Chunki, har doim ham $a>0$ shart bajarilmasligi mumkin. Bu esa Paskal dasturlash tilida xatolikka olib keladi. Hoji bo'lsa boshqa usulda yozish maqsadga muvofiq.	

M-3. Quyidagi ifodalarni Paskal dasturlash tilida yozing.

a) $25^{20}+ 1-y^2 $ Javob: $\text{exp}(20*\ln(25))+\text{abs}(1-\text{sqr}(y))$	b) $[5m]+\{100b\}$ Javob: $\text{int}(5*m)+\text{frac}(100*b)$
d) $x\sin a+y\cos b-5^2$ Javob: $x*\sin(a)+y*\cos(b)-\text{sqr}(5)$	e) $\sin\sin x+\cos\sin y$ Javob: $\sin(\sin(x))+\cos(\sin(y))$
f) $21-\sqrt{2011-x^2}$ Javob: $2011-x^2 \geq 0$ da $21-\text{sqr}(2011-\text{sqr}(x))$	g) $\sqrt[5]{x^3+a}-\sqrt{2}$ Javob: $\text{sqr}(x)*x+a>0$ da $\text{exp}(1/5*\ln(\text{sqr}(x)*x+a))-\text{sqr}(2)$ $\text{sqr}(x)*x+a<0$ da $\text{exp}(-1/5*\ln(-(\text{sqr}(x)*x+a)))-\text{sqr}(2)$ $\text{sqr}(x)*x+a=0$ da $-\text{sqr}(2)$
h) $\sqrt[4]{x}+5\sqrt[3]{y}$ Javob: $x>0$ va $y>0$ da $\text{sqr}(\text{sqr}(x))+5*\text{exp}(1/3*\ln(y))$ $x>0$ va $y>0$ da $\text{sqr}(\text{sqr}(x))+5*\text{exp}(-1/3*\ln(-y))$ $x>0$ va $y=0$ da $\text{sqr}(\text{sqr}(x))$	i) $2\pi R+\pi R^2$ Javob: $2*pi*R+pi*R*R$

M-4. Paskalda yozilgan quyidagi ifodalar orasidan noto'g'ri yozilganini toping.

a) 2^*a+b ; to'g'ri yozilgan	b) $\text{sqr}(x^*b^2)$; noto'g'ri, Paskalda b^2 kabi yozilmaydi	d) $\sin(-3^*x)$; to'g'ri yozilgan
e) $\sin((a+b+\cos(x)))$; noto'g'ri, hitta ochilgan qavs ortiqcha	f) $2^*(-b)+a2$ to'g'ri yozilgan	

M-5. Paskalda yozilgan quyidagi ifodalarni oddiy yozuv ko'rinishiga o'tkazing.

a) $a^*(\text{Sqr}(x)+1)$; Javob: $a \cdot (x^2+1)$	b) $\sin(x^*x^*x-\text{sqr}(\text{sqr}(x))+5)$; Javob: $\sin(x^3-x^4+5)$
d) $\pi^*h^*(\text{sqr}(r) + \text{sqr}(r_1) + r_1^*r_2)/3$; Javob: $\frac{\pi \cdot h \cdot (r^2 + r_1^2 + r_1 \cdot r_2)}{3}$	

20 – dars. Standart funksiyalar va protseduralar, algebraik ifodalar mavzusini takrorlash mavzusiga

M-1. Quyidagi ifodalarni Paskal dasturlash tilida yozing.

Algebraik ifoda ko'rinishi	Paskal dasturlash tilida yozilishi
a) $\frac{x-y}{x^2-y^3}$;	$(x-y) / (\text{sqr}(x)-\text{sqr}(y)^*y)$
b) $\frac{x+y}{xyz} + \sin^2 x$;	$(x+y)/(x^*y^*z)+\text{sqr}(\sin(x))$
d) $(5a^2 + 2x) + \frac{3x}{a^3} + \text{tg}^5 a^3$;	$(5^*a^*a+2^*x)+(3^*x)/(\text{sqr}(a)^*a)+$ $(\text{sqr}(\text{sqr}(\sin(\text{sqr}(a)^*a))^*\sin(\text{sqr}(a)^*a)) /$ $(\text{sqr}(\text{sqr}(\cos(\text{sqr}(a)^*a))^*\cos(\text{sqr}(a)^*a))$
e) $\cos^3 \sin^2 x + \cos a^5$	$\text{sqr}(\cos(\text{sqr}(\sin(x))))^*$ $\cos(\text{sqr}(\sin(x)))+\cos(\text{sqr}(\text{sqr}(a)^*a))$
f) $\sqrt{5+x} - \sqrt{z} \frac{3x}{a^3} + \sqrt[3]{a}$;	$((5+x) \geq 0, z \geq 0, a > 0)$ da $\text{sqr}(5+x)-\text{sqr}(z)^*(3^*x)/(\text{sqr}(a)^*a)+$ $\exp((1/7)^*\ln(a))$

M-2. Quyida Paskalda yozilgan funksiyalarni qiymatini hisoblang.

a) $\text{sqr}(\text{trunc}(4.95)) = \text{sqr}(4) = 16$	
b) $\text{trunc}(\text{int}(4.95) + 0.7) = \text{trunc}(4 + 0.7) = \text{trunc}(4.7) = 4$	
d) $\text{round}(\text{trunc}(3.5) + 0.7) = \text{round}(3 + 0.7) = \text{round}(3.7) = 4$	
e) $3 + \text{frac}(12.5) = 3 + 0.5 = 3.5$	f) $\text{sqr}(\text{sqr}(16)) = \text{sqr}(256) = 16$
g) $\text{sqr}(\text{sqr}(256) + 9) = \text{sqr}(16 + 9) = \text{sqr}(25) = 5$	
h) $\text{sqr}(5 - \text{abs}(-5)) = \text{sqr}(5 - 5) = \text{sqr}(0) = 0$	
i) $\text{abs}(-\text{sqr}(16)) = \text{abs}(-4) = 4$	

M-3. $a=5$, $b=4$ bo'lsa, quyida Paskalda yozilgan ifodalarni qiymatini hisoblang.

a) $\text{abs}(a+b-a*b) = \text{abs}(5+4-5*4) = \text{abs}(9-20) = \text{abs}(-11) = 11$
b) $\text{sqr}(a+b-a*b) - 110 = \text{sqr}(5+4-5*4) - 110 = \text{sqr}(9-20) - 110 = \text{sqr}(-11) - 110 = 121 - 110 = 11$
d) $\text{round}(a/b+0.3) + 9 = \text{round}(5/4+0.3) + 9 = \text{round}(1.25+0.3) + 9 = \text{round}(1.55) + 9 = 2 + 9 = 11$
e) $3 + \text{frac}(b/a) = 3 + \text{frac}(5/4) = 3 + \text{frac}(1.25) = 3 + 0.25 = 3.25$
f) $\text{sqr}(\text{sqr}(a) - b*b) = \text{sqr}(\text{sqr}(5) - 4*4) = \text{sqr}(25 - 16) = \text{sqr}(9) = 3$
g) $\text{sqr}(\text{sqr}(a+b) + 6) = \text{sqr}(\text{sqr}(5+4) + 6) = \text{sqr}(\text{sqr}(9) + 6) = \text{sqr}(3+6) = \text{sqr}(9) = 3$
h) $\text{sqr}(a - \text{abs}(b-a)) = \text{sqr}(5 - \text{abs}(4-5)) = \text{sqr}(5 - \text{abs}(-1)) = \text{sqr}(5-1) = \text{sqr}(4) = 16$
i) $\text{abs}(9 - \text{sqr}(a*b+a)) = \text{abs}(9 - \text{sqr}(5*4+5)) = \text{abs}(9 - \text{sqr}(25)) = \text{abs}(9-5) = \text{abs}(4) = 4$

M-4. Quyida Paskalda yozilgan ifodalar qiymati qanday turdagi o'zgarimas bo'lishini aniqlang.

a) $\text{abs}(-\text{sqr}(2011))$ Javob: sqr ni natijasi doimo haqiqiy son bo'ladi, abs esa haqiqiy sonni haqiqiy songa o'tkazadi, demak, ifoda qiymati haqiqiy son	b) $\text{abs}(\text{sqr}(2)) + 19$ Javob: sqr butun sonni butun songa o'tkazadi, abs ham butun sonni butun songa o'tkazadi, butun songa 19 ni qo'shsak yana butun son hosil bo'ladi, demak, ifoda qiymati butun son
d) $\text{frac}(\text{abs}(-20))$ Javob: abs butun sonni butun songa o'tkazadi, lekin frac butun sonni haqiqiy songa o'tkazadi, demak, ifoda qiymati haqiqiy son	e) $20 + \text{sqr}(4)$ Javob: sqr ni natijasi doimo haqiqiy son bo'ladi, haqiqiy songa 20 ni qo'shsak yana haqiqiy son hosil bo'ladi, demak, ifoda qiymati haqiqiy son

f) random*20 Javob: random ni natijasi doimo haqiqiy son bo'ladi, haqiqiy sonni 20 ga ko'paytirsak yana haqiqiy son hosil bo'ladi, demak, ifoda qiymati haqiqiy son	g) random(4)*20 Javob: random(4) ni natijasi Word turdagi, ya'ni butun son bo'ladi, butun sonni 20 ga ko'paytirsak yana butun son hosil bo'ladi, demak, ifoda qiymati butun son
h) 23*sqrt(1) Javob: sqrt ni natijasi doimo haqiqiy son bo'ladi, haqiqiy sonni 23 ga ko'paytirsak yana haqiqiy son hosil bo'ladi, demak, ifoda qiymati haqiqiy son	i) int(1.9)*trunc(0.2) Javob: int ni natijasi doimo kasr qismi 0 ga teng haqiqiy son bo'ladi, trunc ixtiyoriy sonni butun songa o'tkazadi, haqiqiy sonni butun songa ko'paytirsak yana haqiqiy son hosil bo'ladi, demak, ifoda qiymati haqiqiy son

21 – dars. O'zlashtirish va ma'lumotlarni ekranga chiqarish operatorlari mavzusiga

M-1. Quyidagi ifodalarni o'zlashtirish operatori yordamida yozing.

a) a= 48; b= 51; Javob: a:=48; b:=51;	b) x= 0; a= 3,6x+sinx; Javob: x:=0; a:=3.6*x+sin(x);
d) g= 4; g= g+16; Javob: g:=4; g:=g+16;	e) a= 9,81; m= 50; F= m a; Javob: a:=9.81; m:=50; F:=m*a;
f) $x= 1; y = \frac{x-63}{21-7x};$ Javob: x:=1; y:=(x-63)/(21-7*x);	g) z= 25; z= $\sqrt{z};$ Javob: z:=25; z:=sqrt(z);
h) a= 6; b= 8; c= $\sqrt{a^2 + b^2};$ Javob: a:=6; b:=8; c:=sqrt(a*a+b*b);	i) $x_1= 9; x_2= x_1^2+5x_1.$ Javob: x1:=9; x2:=sqr(x1)+5*x1;

M-2. Chiqarish operatorlari bajarilgach natija ekranda qanday aks etishini yozing.

a) write('a='); write(2+3); write('='); write('2+3'); Ekranda: a=5=2+3	b) writeln('a='); write(2+3); write('='); writeln('2+3'); Ekranda: a=
---	--

	$5=2+3$
d) <code>write('a=');</code> <code>writeln(2+3);</code> <code>write('=');</code> <code>writeln('2+3');</code> Ekkranda: a=5 =2+3	e) <code>write('a=');</code> <code>write(5);</code> <code>writeln('=');</code> <code>write('2+3');</code> Ekkranda: a=5= 2+3
f) <code>writeln('a=');</code> <code>writeln(5);</code> <code>write('=');</code> <code>write('2+3');</code> Ekkranda: a= 5 =2+3	g) <code>write('a=');</code> <code>writeln(5);</code> <code>writeln('=');</code> <code>write('2+3');</code> Ekkranda: a=5 = 2+3

M-3. Paskalda yozilgan quyidagi dastur lavhalaridagi barcha o'zgaruvchilarning oraliq qiymatini va ekranga chiqadigan natijani aniqlang.

a) <code>a:= -cos(pi)-sin(pi/2); x:= x*x + a;</code> <code>writeln('a=',a, 'x=',x);</code> Javob: oraliq natijalar $a= -\cos(\pi) - \sin(\pi/2) = -(-1) - 1 = 1 - 1 = 0;$ $x=0*0+0=0+0=0;$ Ekkranda: $a=0.0000000000E+00x=0.0000000000E+00$	d) <code>a:= 9; b:=a+a;</code> <code>a:= a*a-b;</code> <code>write('a=', a);</code> <code>write(' b=', b);</code> Javob: oraliq natijalar $a=9; b=9+9=18;$ $a=9*9-18=81-18=63;$ Ekkranda: $a= 63 b=18$
b) <code>a:= 'Men ';</code> <code>g:= 'mustaqil ';</code> <code>b:= 'O`zbekiston';</code> <code>m:= ' farzandiman!'</code> <code>write(a, g, b, m);</code> Javob: oraliq natijalar $a= 'Men ';$ $g= 'mustaqil ';$ $b= 'O`zbekiston';$ $m= ' farzandiman!'$ Ekkranda: Men mustaqil O`zbekiston farzandiman!	

22 – dars. O‘zlashtirish va ma’lumotlarni ekranga chiqarish operatorlari mavzusini takrorlash mavzusiga

M-1. Quyidagi ifodalarni o‘zlashtirish operatori yordamida yozing.

<p>a) $y = \frac{x-21}{7-x^{63}}$;</p> <p>Javob: $y:=(x-21)/(7-\exp(63*\ln(x)))$;</p>	<p>b) $a = 3.6x + \sin x$;</p> <p>Javob: $a:=3.6*x+\sin(x)$;</p>
<p>d) $z = \sqrt{x-5y+xtgx}$;</p> <p>Javob: $z:=\text{sqrt}(x-5*y+x*\sin(x)/\cos(x))$;</p>	<p>e) $S = \pi r^2$;</p> <p>Javob: $S:=\text{pi}*sqr(r)$;</p>
<p>f) $F = ma$;</p> <p>Javob: $F:=m*a$;</p>	<p>g) $S = \frac{ah}{2}$.</p> <p>Javob: $S:=a*h/2$;</p>

M-2. Chiqarish operatori natijasi qanday bo‘lishini aniqlang.

<p>a) $a:=123.45$; write('a=', a:2:1); Ekranda: a=123.4 a soni xotiraga 123.45 kabi yoziladi. Chiqarish formati a:2:1 da butun qism uchun 2-1-1=0 xona (hammasi bo‘lib 2 xona kasr qismiga 1 xona, nuqta uchun 1 xona) ajratildi, lekin a sonning butun qismi 3 xonali, shu sababli butun qism uchun chiqarish formati inkor etilib butun qism to‘liq, kasr qismidan 1 xona chiqariladi.</p>
<p>b) $a:=123.45$; write('a=', a:5:1); Ekranda: a=123.4 Chiqarish formati a:5:1 butun qism uchun 5-1-1=3 xona (hammasi bo‘lib 5 xona kasr qismiga 1 xona, nuqta uchun 1 xona) ajratildi, a sonning butun qismi ham 3 xonali bo‘lgani uchun chiqarish formati ishlaydi, shunda kasr qismidan 1 xona chiqariladi</p>
<p>d) $a:= '2011'$; writeln(a:3, ' yil':3); Ekranda: 2011 yil Chiqarish formati a:3 ajratgan 3 ta belgi joyiga 4 ta belgisi '2011', ' yil':3 ajratgan 3 ta belgi joyiga 4 ta belgisi ' yil' sig‘maydi, shuning uchun ikkala chiqarish formati ham bekor qilinadi</p>
<p>e) $a:= '2011'$; writeln(a:4, ' yil':5); Ekranda: 2011 yil Chiqarish formati a:4 ajratgan 4 ta belgi joyiga 4 ta belgisi '2011', ' yil':5 ajratgan 5 ta belgi joyiga 4 ta belgisi ' yil' sig‘adi, shuning</p>

uchun ikkala chiqarish formati qabul qilinadi hamda 'yil':5 formatida ortib qolgan 1 ta joyga probel chiqariladi

M-3. Dasturdagi o'zgaruvchilarning qiymati turiga mos bo'lishi uchun so'roq belgisi o'rniga zaruriy standart funktsiyani yozing va ekranga chiqadigan natijani aniqlang.

```
a) var a, b, c: integer;  
   begin a:=25; b:=?(sqrt(a)); c:=?(a/b);  
         writeln(a, ' ', b, 'c= ', c);  
   end.
```

Javob:

b, c o'zgaruvchilar turi **integer** (butun), demak ifodalarning natijasi ham butun chiqishi kerak. Lekin **sqrt(a)** va a/b natijalari doimo haqiqiy, shuning uchun ifodalardagi ? belgilari o'rniga **trunc** yoki **round** yozilishi kerak. Hisoblaymiz:

$b = \text{trunc}(\text{sqrt}(25)) = \text{trunc}(5.0) = 5$; $c = \text{trunc}(25/5) = \text{trunc}(5.0) = 5$
(round ham shu natijalarni beradi).

Ekkranda:

25 5c= 5

```
b) var x, y, z: word;  
   begin x:=?(?(-7.21)); y:=?(sqrt(x*x));  
         z:=?(x+y-100); write(z-x, y);  
   end.
```

Javob:

x, y, z o'zgaruvchilar turi **word** (manfiymas butun), demak ifodalarning natijasi ham manfiymas butun chiqishi kerak. Demak, x ning ifodasidagi ? belgilaridan biri o'rniga natija manfiy bo'lmasligi uchun **abs** yoziladi, ikkinchi ? belgisi o'rniga natija butun bo'lishi uchun **trunc** yoki **round** yoziladi: $x = \text{trunc}(\text{abs}(-7.21)) = \text{trunc}(7.21) = 7$ (**trunc** yoki **round** yozilishi, **abs** va **trunc** o'zni almashtirilishi bir xil natija beradi). **sqrt(x*x)** natijasi doimo haqiqiy, shuning uchun y o'zgaruvchi ifodasidagi ? belgisi o'rniga **trunc** yoki **round** yozilishi kerak: $y = \text{trunc}(\text{sqrt}(7*7)) = \text{trunc}(\text{sqrt}(49)) = \text{trunc}(7.0) = 7$ (round ham shu natijani beradi). Oxirgi z o'zgaruvchi ifodasida $x+y-100 = 7+7-100 = 14-100 = -86$ bo'lgani uchun ? belgisi o'rniga **abs** yozilishi shart: $z = \text{abs}(-86) = 86$. Chiqarish ro'yxatidagi ifodani hisoblaymiz:

$z-x = 86-7 = 79$

Ekkranda:

797

23 – dars. Ma'lumotlarni xotiraga muloqot usulida kiritish operatori mavzusiga

M-1. Read operatori yordamida N sonini kvadratini N ning 10; 11; 12; 13; 14; 15 qiymatlarida hisoblash dasturini tuzing.

Yechim.

Izoh	Dasturi
1-usul. N soni kvadratini xotirada saqlash uchun M o'zgaruvchi kiritamiz. N sonini va N sonining kvadrati qiymatlari musbat va 255 dan kichik, shuning uchun ikkala o'zgaruvchi uchun butun sonning byte turini tanlaymiz.	Var N, M: byte; Begin Read(N); M:=sqr(N); Write(N, ' ni kvadrati: ', M); End.
2-usul. N sonini kvadratini xotirada saqlamasdan ham chiqarish mumkin. U holda N o'zgaruvchini qiymatlari musbat va 255 dan kichik bo'lgani uchun byte turda tanlaymiz.	Var N: byte; Begin Read(N); Write(N, ' ni kvadrati: ', N*N); End.

M-2. Quyidagi dasturda a o'zgaruvchining "O'ZBEKISTONIM"; "VATANIM"; "ONA DIYORIM" qiymatlarini xotiraga qulay usulda kiritib natija oling.

Var a, b, g: string;

Begin

b:= ' – SAJDAGOH KABI'; g:= ' MUQADDASDIR!'; write(a,b,g);

readln;

End.

Yechim.

Topshiriqda a o'zgaruvchining uchta qiymati berilgan. Ularni xotiraga muloqot usulda kiritish qulaydir. Buning uchun dasturga Readln(a); satrini qo'shish kerak. Dasturdagi readln; operatori dastur natijasini ko'rib olishni qulaylashtirish uchun yozilgan. Bu operator ishlashi uchun ham Read(a); emas Readln(a); yozilishi lozim. Ekran-dagi natijada so'zlar qo'shilib ketmaydi, chunki b va g o'zgaruvchilar qiymatida probellar hisobga olingan.

```

Var a, b, g: string;
Begin Readln(a);
  b:= ' - SAJDAGOH KABI'; g:= ' MUQADDASDIR!';
  write(a, b, g); readln;
End.

```

Dasturni satrli **a** o'zgaruvchining har bir qiymatida ishlatib natija quyidagicha olinadi.

Dastur ishga tushirilgach (**Ctrl+F9** tugmachalarini birgalikda bosib yoki **RUN** menyusidan Run buyrug'ini tanlab), ekranda satr boshidan yurgich chiqadi va dastur **a** satrli o'zgaruvchiga qiymati kiritilishini kutib turadi. Topshiriq shartiga ko'ra **a** satrli o'zgaruvchi qiymati sifatida "O'ZBEKISTONIM" so'zini kiritib, **ENTER** klavishi bosilsa ekranda quyidagilar aks etadi:

O'ZBEKISTONIM – SAJDAGOH KABI MUQADDASD!R!

Endi **ENTER** klavishi bosilsa, yana matn muharriri oynasiga qaytiladi. Dastur qayta ishlatilib **a** satrli o'zgaruvchi qiymati sifatida "VATANIM" so'zini kiritib, **ENTER** klavishi bosilsa ekranda quyidagilar aks etadi:

VATANIM – SAJDAGOH KABI MUQADDASDIR!

Huddi shunday 'ONA DIYORIM' qiymati kiritiladi.

M-3. "Matiz" avtomobili joyidan qo'zg'alib T sekundda S metr yo'l bosdi. Uning o'rtacha tezligini (m/s) larda quyidagi qiymatlarda hisoblash dasturini ma'lumotlarni muloqot usulda kiritish orqali tuzing (yo'llanma: $V=S/T$).

Yechim.

var T, S: integer; V: real;

Begin

write('Vaqtni kiriting: '); readln(T);

write('Masofani kiriting: '); readln(S);

V:=S/T;

write('Matizning o'rtacha tezligi V= ',V:0:2, ' m/s'); readln;

End.

24 – dars. Ma'lumotlarni xotiraga muloqot usulida kiritish operatori mavzusini takrorlash mavzusiga

M-1. Quyidagi dasturda a o'zgaruvchining "ozod"; "obod" qiymatlarini xotiraga qulay usulda kiritib natija oling.

Var a, b, g: string;

Begin

b:= 'Bizdan *; g:= ' vatan qolsin!'; write(b, a, g); readln;

End.

Yechim. **Begin** dan keyin Write(*a ni qiymatini kiriting: *); Readln(a); operatorlari yoziladi.

M-2. Quyidagi dasturdagi so'roq belgisi o'rniga joriy yilni m o'zgaruvchi yordamida kiriting va mustaqilligimiz bilan tabriklovchi natija oling.

Var a, b, g: string; m: word;

Begin write(*Joriy yilni kiriting: *); ?;

a:= 'Mustaqillikning *; g:= ' yilligi bilan *; b:= 'tabriklaymiz!';

writeln(a); writeln(m-1991, g); write(b); readln;

End.

Yechim. So'roq belgisi o'rniga Readln(m); operatori yoziladi. Dasturni ishlatish uchun **Ctrl+F9** tugmachalari birgalikda bosiladi yoki **RUN** menyusidan Run buyrug'ini tanlanadi. Ekranda "Joriy yilni kiriting: " matni aks etganda joriy yil, masalan 2012, yoziladi va **ENTER** klavishi bosiladi.

M-3. Agar jismga ta'sir etayotgan kuch F, olgan tezlanishi a bo'lsa, quyidagi qiymatlarda jismni massasini hisoblash dasturini kiritish operatoridan foydalanib tuzing (yo'llanma $m=F/a$) va natijalar oling.

a) $F=15, a=55$; b) $F=55, a=15$; d) $F=10, a=100$; e) $F=100, a=10$;

Yechim. Quyidagi dastur kompyuter xotirasiga kiritilgach F va a o'zgaruvchilar juftligi uchun dastur qayta ishlatiladi va natijalar olinadi.

Var F, a: integer; m: real;

Begin write(*F kuch qiymatini kiriting: *); readln(F);

```

write('Tezlanish a ni qiymatini kiriting: '); readln(a);
m:=F/a; write('Jismning massasi m= ', m:0:2.); readln;
End.

```

M-4. $a=19$, $b=2$, $d=1950$ qiymatlarni xotiraga qulay usulda kiritib quyidagi ifodalarning qiymatini hisoblash dasturini tuzing.

a) $y = a + b^2 + ad$; b) $t = \sqrt{a+b} - \sqrt[3]{d-a}$;

d) $s = b \cos a + \sin d$; e) $n = \pi d^2 + ab$.

Yechim. Ma'lumotlarni xotiraga kiritishni qulay usuli muloqot usulidir, ya'ni Read yoki Readln operatorlaridan foydalanishdir.

a) $y = a + b^2 + ad$;

Topshiriq shartidagi barcha o'zgaruvchilar va y o'zgaruvchi qiymati butun, shuning uchun **integer** turidan foydalanamiz.

var a, b, d, y: integer;

```

Begin Write('a= '); Readln(a); Write('b= '); ReadLn(b); Write('d= ');
ReadLn(d); y:=a+sqr(b)+a*d; Write('y= ', y);

```

End.

b) $t = \sqrt{a+b} - \sqrt[3]{d-a}$;

Topshiriq shartidagi barcha o'zgaruvchilar butun va t o'zgaruvchi qiymati ildiz qatnashganligi sababli haqiqiy, $a+b=19+2=21 \geq 0$ va $d-a=1950-19=1931 \geq 0$ o'rinli.

var a, b, d: integer; t: real;

```

Begin Write('a= '); Readln(a); Write('b= '); ReadLn(b);

```

```

Write('d= '); ReadLn(d);

```

```

t:=sqrt(a+b)-exp((1/3)*ln(d-a)); Write('t= ', t:0:2);

```

End.

d) $s = b \cos a + \sin d$;

Topshiriq shartidagi barcha o'zgaruvchilar butun va s o'zgaruvchi qiymati trigonometrik funksiyalar qatnashganligi sababli haqiqiy.

var a, b, d: integer; s: real;

```

Begin Write('a= '); Readln(a); Write('b= '); ReadLn(b);

```

```

Write('d= '); ReadLn(d); S:=b*cos(a)+sin(d); Write('s= ', s:0:2);

```

End.

e) $n = \pi d^2 + ab$.

Topshiriq shartidagi barcha o'zgaruvchilar butun va n o'zgaruvchi qiymati π soni qatnashganligi sababli haqiqiy.

```

var a, b, d: integer; n: real;
Begin Write('a= '); ReadLn(a); Write('b= '); ReadLn(b);
  Write('d= '); ReadLn(d); n:=pi*sqr(d)+a*b; Write('n= ', n:0:2);
End.

```

M-5. Tomonlari a , b , c bo'lgan uchburchakning yuzini hisoblash dasturini kiritish operatoridan foydalanib tuzing va natijalar oling.

- a) $a=5$, $b=7$, $c=4$; b) $a=8$, $b=6$, $c=10$;
 d) $a=3$, $b=4$, $c=5$; e) $a=10$, $b=8$, $c=10$;

Yechim. Kiritish operatori deganda ma'lumotlarni xotiraga muloqot usulida kiritish, ya'ni Read yoki ReadLn operatorlaridan foydalanish, tushuniladi. Topshiriq shartidagi **a**, **b**, **c** o'zgaruvchilar butun va **s** o'zgaruvchi qiymati ildiz qatnashganligi sababli haqiqiy.

```

Var a, b, c: integer; p, s: real;
Begin Write('a ni kiriting= '); ReadLn(a); Write('b ni kiriting= ');
  ReadLn(b); Write('c ni kiriting= '); ReadLn(c);
  p:=(a+b+c)/2; s:=sqr(p*(p-a)*(p-b)*(p-c));
  Write('Uchburchk yuzi s= ', s:0:3); readLn;
End.

```

M-6. $y=23 \cdot x+1$ funksiyaning qiymatini x ning -5 , -4 , -3 , -2 , -1 , 0 , 1 , 2 , 3 , 4 , 5 qiymatlarini xotiraga qulay usulda kiritib hisoblash dasturini tuzing va natijalar oling.

Yechim. Ma'lumotlarni xotiraga kiritishni qulay usuli muloqot usulidir, ya'ni Read yoki ReadLn operatorlaridan foydalanishdir. x o'zgaruvchining berilgan har bir qiymati uchun dastur qayta ishlatiladi va natijalar olinadi.

```

Var x, y: integer;
Begin Write('x ning qiymatini kiriting= '); ReadLn(x);
  y:=23*x+1; write('x= ', x, '-da ', 'y= ', y ); readLn;
end.

```

M-7. $y = 21 \cdot x^2 + 7 \cdot x + 1963$ funksiyaning qiymatini x ning -5 , -4 , -3 , -2 , -1 , 0 , 1 , 2 , 3 , 4 , 5 qiymatlarini xotiraga qulay usulda kiritib hisoblash dasturini tuzing va natijalar oling.

Yechim. Avvalgi topshiriq dasturida $y:=23 \cdot x+1$; o'rniga $y:=21 \cdot \text{sqr}(x)+7 \cdot x+1963$; yoziladi.

25 – dars. Matn holatida ekran bilan ishlash mavzusiga

M-1. Quyidagi dastur natijasida ekran rangi, matn foni rangi va matn ranglari qanday bo'lishini va matnlar joyini aniqlang.

```
Uses crt;  
Begin textbackground(yellow); writeln('O'zbekiston*);  
  clrscr; textcolor(4); write('kelajagi '); textbackground(blue);  
  writeln('buyuk*'); textcolor(2); write('DAVLAT!'); readln;  
end.
```

Yechim:

Uses crt;

Begin

```
  textbackground(yellow); writeln('O'zbekiston*);
```

{ekranda 'O'zbekiston' matni oq rangda sariq fonda chiqadi, lekin Paskal dasturlash tili buyruqlarni katta tezlikda bajargani sababli foydalanuvchi bu matnni ko'rishga ulgurmaydi. Bu matnni ko'rib olish uchun clrscr dan avval readln; yozish mumkin.}

```
  clrscr; {clrscr; dan avval textbackground(yellow); yozilgani uchun  
ekran sariq rang bilan "o'chiriladi", ya'ni ekran tozalanib sariq rang  
bilan bo'yaladi, hamda yurgich ekranning chap yuqori burchagiga olib  
kelinadi}
```

```
  textcolor(4); write('kelajagi '); {bu matn sariq ekranda qizil rangda  
ekranning chap yuqori burchagida aks etadi}
```

```
  textbackground(blue); writeln('buyuk*'); {bu matn sariq ekranda qizil  
rangda ko'k fonda avvalgi matnning davomida probeldan keyin aks  
etadi, chunki avvalgi chiqarish operatori Ln qo'shimchasisiz yozilgan}
```

```
  textcolor(2); write('DAVLAT!'); readln; {bu matn sariq ekranda ya-  
shil rangda ko'k fonda keyingi satr boshida aks etadi, chunki avvalgi  
chiqarish operatori Ln qo'shimchasi bilan yozilgan}
```

end.

Natijaning ekrandagi ko'rinishi:



M-2. "O'zbekiston konstitutsiyasi – erkinlik posboni" matniga qizil, matn foniga ko'k rangni tanlab ekranga chiqaring.

Yechim:

Uses crt;

Begin

```
clrscr; {ekran qora rang bilan "o'chiriladi", ya'ni ekran tozalanib qo-  
ra rang bilan bo'yaladi, yurgich ekran boshiga o'rnatiladi}  
textbackground(blue); {matnga ko'k fon tanlandi}  
textcolor(4); {matnga qizil rang tanlandi}  
writeln('O'zbekiston konstitutsiyasi -- erkinlik posboni'); readln;  
end.
```

Natijaning ekrandagi ko'rinishi:



M-3. Quyidagi dasturga shunday protseduralar qo'shingki, barcha matn ko'k rangda, matn fonlari sariq rangda ekranga chiqsin. Dastur ishlashini izohlang.

Var a,b: string; m, s : real;

Begin

```
a:= 'Kvadratning tomonini kiriting: '; b:= 'Kvadratning yuzi: ';  
Write(a); readln(m); s:=sqr(m); write(b, s:8:2, ' kvadrat birlik');  
readln;
```

End.

Yechim:

Dasturga matn rangi ko'k (blue) va matn foni rangi sariq (yellow) bo'lishi uchun Begin xizmatchi so'zidan keyin 2 ta protsedura qo'shamiz:

```
textcolor(blue); textbackground(yellow);
```

Dastur bajarilishining izohi:

Dastur ishga tushganda **a** va **b** satrli turdagi o'zgaruvchilarga xotiradan har biri uchun 255 baytdan, **m** va **s** haqiqiy turdagi o'zgaruvchilarga xotiradan har biri uchun 2 baytdan joy ajratadi.

Dasturning asosiy qismi ishlaganda matn uchun **textcolor(blue)**; protsedurasi ishi sababli ko'k, matn foni uchun **textbackground(yellow)**; protsedurasi ishi sababli sariq rang belgilanadi.

Birinchi va ikkinchi o'zlashtirish operatorlari bajarilganda satrli **a** o'zgaruvchi uchun ajratilgan xotiradagi joyga "Kvadratning tomonini

kiriting: “, satrli **b** oʻzgaruvchi uchun ajratilgan xotiradagi joyga “Kvadratning yuzi: “qiymatlar yoziladi.

Write(a); bajarilganda **a** oʻzgaruvchining xotiradagi “Kvadratning tomonini kiriting: “ qiymati chaqirilib, ekranda 1-satrdan koʻk rangda sariq fonda aks ettiriladi.

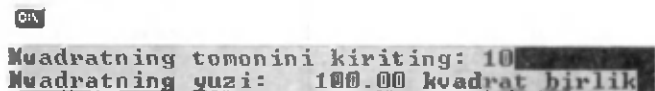
Keyingi **readln(m)**; operatori ishlaganda dastur **m** oʻzgaruvchiga sonli qiymat berilishini 1-satrdagi matn davomida kutib turadi. Sonli **m** oʻzgaruvchiga biror son qiymat, masalan, **10** sonini kiritsak (koʻk rangda aks etadi), **m** oʻzgaruvchi uchun ajratilgan xotiradagi joyga **10** qiymat yoziladi.

Keyingi oʻzlashtirish operatori **s:=sqr(m)** ishlaganda **m** oʻzgaruvchini qiymatini chaqiradi va kvadratga koʻtarib hosil boʻlgan **100** sonini sonli **s** haqiqiy turdagi oʻzgaruvchi uchun ajratilgan xotiradagi joyga **1.000000000E+2** kabi yozadi.

Chiqarish operatori **write(b, s:8:2, ‘ kvadrat birlik’)**; xotiradan **b** satrli oʻzgaruvchining “Kvadratning yuzi: “ qiymatini chaqirib sariq fonda koʻk rangda aks ettiradi, uning davomidan **s** sonli oʻzgaruvchining 100 qiymatini chaqirib berilgan format asosan koʻk rangda aks ettiradi va uning davomidan “ kvadrat birlik” matnini koʻk rangda aks ettiradi.

Endi natijani koʻrib olganimizcha kuttirish uchun yozilgan **readln**; operatori ishlaydi. Va nihoyat biror klavish bosilganda oxirgi **End**. xizmatchi soʻzi ishidan keyin dastur oʻz ishini toʻxtatadi.

Natijaning ekrandagi koʻrinishi:



```
□%  
Kvadratning tomonini kiriting: 10  
Kvadratning yuzi: 100.00 kvadrat birlik
```

M-4. A va B oʻzgaruvchilarning berilgan qiymatini kiritib quyidagi dastur ishlashini izohlang.

Uses Crt;

Begin

ClrScr; write('A= '); readln(a); write('B= '); readln(b);

GotoXY(A,B); Writeln('Kitob bilim manbai'); readln;

End.

a) A=1, B=1; b) A=8, B=1; d) A=1, B=8;

c) A=8, B=8; f) A=25, B=25; g) A=100, B=10;

Dastur bajarilishining izohi:

Dastur ishga tushirilganda xatolik xabarini ekranga **“Error 3: Unknown identifier”** (noma’lum identifikator, ya’ni bu holda noma’lum o’zgaruvchi) yozuvi orqali ifodalaydi, chunki **a** va **b** butun qiymatli o’zgaruvchilar tavsiflanmagan. Berilgan qiymatlarni tahlil etib, **a** va **b** butun o’zgaruvchilar **byte** turda tavsiflanishi kifoyaligini ko’rish mumkin. Shuning uchun dasturda **Uses Crt;** va **Begin** orasiga **var: a, b: byte;** tavsifini yozamiz.

Tuzatish kiritilgan dastur har safar ishga tushirilganda **CtrSer;** protsedurasi ta’sirida ekran (qora rangda) tozalanib yurguch I-ustun va I satr kesishgan joyda o’rnatiladi.

So’ngra **write(‘A= ‘); readln(a); write(‘B= ‘); readln(b);** operatorlari yordamida **a** va **b** butun sonli o’zgaruvchilar qiymati qora ekranda oq rangda muloqot usulda xotiraga kiritiladi.

GotoXY(A,B); protsedurasi ishlaganda yurgich A-ustun va B-satr kesishgan joyga o’rnatiladi va shu joydan **Writeln(‘Kitob bilim manbai’);** protsedurasi asosida “Kitob bilim manbai” matni qora ekranda oq rangda aks etadi.

Natijalarning ekrandagi ko’rinishi:

a) A=1, B=1;

```
Turbo Pascal
Kitob bilim manbai
B= 1
```

b) A=8, B=1;

```
Turbo Pascal
A= 8   Kitob bilim manbai
B= 1
```

d) A=1, B=8;

```
Turbo Pascal
A= 1
B= 8

Kitob bilim manbai
```

e) A=8, B=8;

```
Turbo Pascal
A= 8
B= 8

Kitob bilim manbai
```

f) A=25, B=25;

g) A=100, B=10;

```
Turbo Pascal
A= 100
B= 10
Kitob bilim manbai
```

Ma’lumki, **GotoXY(A,B)** protsedurasida quyidagi tengsizlik bajarilishi



shart:

$1 \leq A \leq 80$ va $1 \leq B \leq 25$
 Agar A yoki B ning qiymati shu tengsizlikni qanoatlantirmasa, u holda bu qiymatlar inkor etiladi va yurgich avvalgi natijalarning davomiga bajarilgan eng oxirgi operator ta'sirida o'atiladi.

M-5. Ekranga ismingiz, familiyangiz va sharifingizni 3 xil rangda matnni 3 xil fon rangida va ekranning turli joylarida chiqaring.

Yechim:

Uses crt;

Begin

```
  clrscr; textcolor(red); textbackground(3); GotoXY(2,2);
  writeln('BAHODIR'); {bu matn ekranda qizil rangda billur fonda 2-
  ustun va 2 satr kesishgan joyda aks etadi}
  textcolor(green); textbackground(blue); GotoXY(26,8);
  writeln('XURRAMOV'); {bu matn ekranda yashil rangda ko'k fonda
  26-ustun va 8 satr kesishgan joyda aks etadi}
  textcolor(13); textbackground(6); GotoXY(51,14);
  writeln('SAPPAROVICH'); {bu matn ekranda pushti rangda jigarrang
  fonda 51-ustun va 14-satr kesishgan joyda aks etadi}
end.
```

Natijaning ekrandagi ko'rinishi:



26 – dars. Matn holatida ekran bilan ishlash mavzusini takrorlash mavzusiga

M-1. “Ajdodlar merosini qadrlaylik” matniga ko‘k, matn foniga yashil rangni tanlab ekranga chiqaring.

Yechim:

Uses crt;

Begin

```
clrscr; {ekran tozalanib, yurgich ekran boshiga o‘rnatiladi}
textcolor(blue); textbackground(2);
writeln(‘Ajdodlar merosini qadrlaylik ‘); {matn ekranda ko‘k rangda
yashil fonda aks etadi}
end.
```

Natijaning ekrandagi ko‘rinishi:

M-2. “Vatanni sevmoq iymondandir!” matnini ekranning o‘ng tomonidan 12-satrdan yashil rangda qizil fonda chiqaring.

Yechim:

Matn holatida 80 ta ustun bor, shuning uchun matndagi belgilar soni 27 ga teng bo‘lgani uchun ustun tartib raqamini quyidagicha aniqlaymiz:

$$80 - \text{Son} + 1 = 80 - 27 + 1 = 53 + 1 = 54.$$

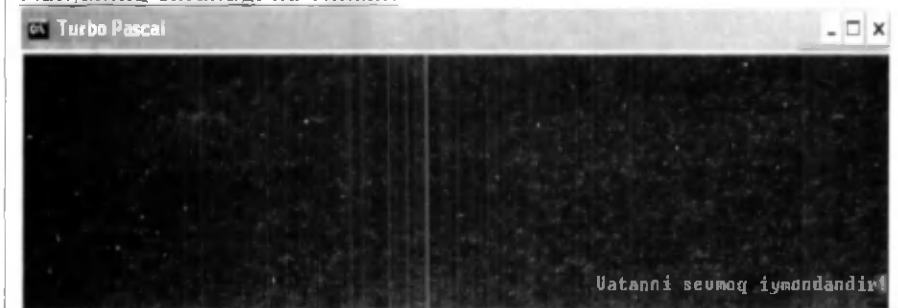
Uses crt;

Begin

```
clrscr; textcolor(green); textbackground(red);
GotoXY(54,12); writeln(‘Vatanni sevmoq iymondandir!’); readln;
```

end.

Natijaning ekrandagi ko'rinishi:



M-3. "Eng yuqoridagi satr, chapdan", "Eng yuqoridagi satr, o'ngdan", "Eng yuqoridagi satr, o'rtadan", "Eng quyidagi satr, chapdan", "Eng quyidagi satr, o'ngdan", "Eng quyidagi satr, o'rtadan", "Markazdagi satr, chapdan", "Markazdagi satr, o'ngdan", "Markazdagi satr, o'rtadan" matnlarini ekranning matn mazmuniga mos joylarida chiqishini ta'minlovchi dastur tuzing.

Yechim:

Ma'lumki, "Eng yuqoridagi satr" bu 1-satr, "chapdan" ga mos ustun 1-ustun; "Eng quyidagi satr" uchun 23-satr olish lozim (agar 24-yoki 25-satrlar olinsa, 1-satrdagi matn ko'rinmay qoladi); "Markazdagi satr" bu 12-satr. Matnlardagi "o'ngdan" talabini bajarish uchun matndagi belgilar soni Son ni aniqlab, kerakli ustun tartib raqami T-2 mashqdagi kabi aniqlanadi. Matnlardagi "o'rtadan" talabini bajarish uchun esa matndagi belgilar soni Son ni aniqlab, kerakli ustun tartib raqami quyidagicha aniqlanadi: $[(80 \text{ Son}):2]+1$. Ekran rangini oq, matn rangini qizil tanlaymiz. Dastur natijasidan ko'rish mumkinki, eng yuqori satrda aks ettirilayotgan ma'lumotlar bir satrga sig'maganligi uchun ba'zi so'zlar ustida boshqa so'z aks etgan.

Uses crt;

Begin

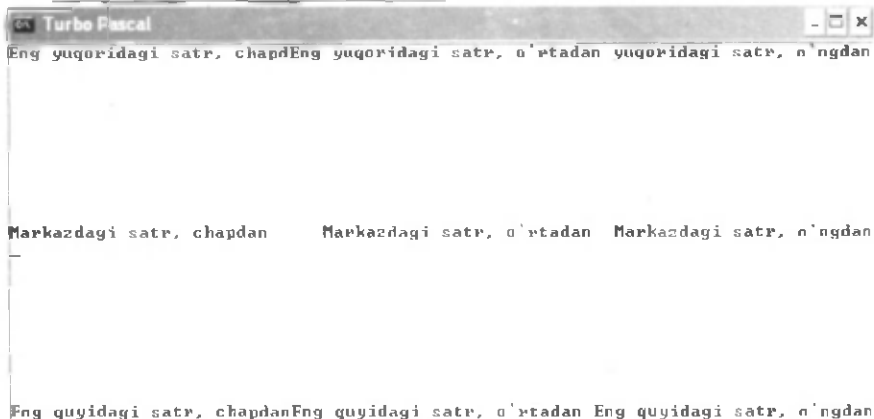
```
textbackground(white); clrscr; textcolor(red);  
gotoXY(1,1); writeln('Eng yuqoridagi satr, chapdan');  
gotoXY(53,1); writeln('Eng yuqoridagi satr, o'ngdan');  
gotoXY(27,1); writeln('Eng yuqoridagi satr, o'rtadan');  
gotoXY(1,23); writeln('Eng quyidagi satr, chapdan');  
gotoXY(55,23); writeln('Eng quyidagi satr, o'ngdan');
```

```

gotoXY(27,23); writeln('Eng quyidagi satr, o'rtadan');
gotoXY(1,12); writeln('Markazdagi satr, chapdan');
gotoXY(57,12); writeln('Markazdagi satr, o'ngdan');
gotoXY(30,12); writeln('Markazdagi satr, o'rtadan');
readln;
end.

```

Natijaning ekrandagi ko'rinishi:



M-4. “Suv – hayot manbai” degan iborani ekranda 5 xil rangda turli joylarda chiqaruvchi dastur tuzing.

Yechim:

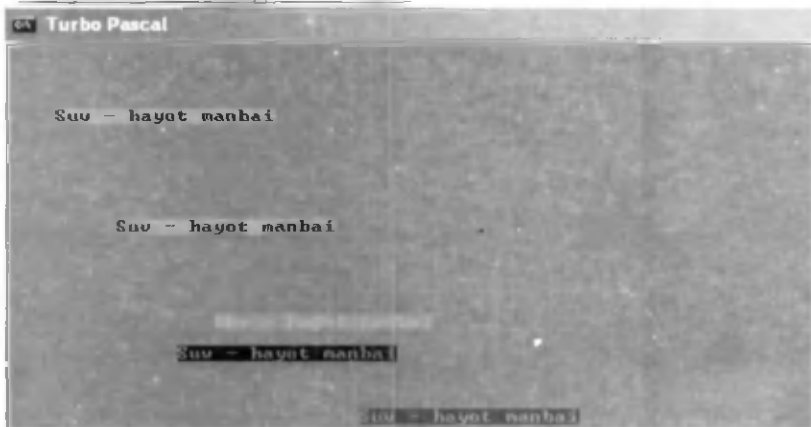
```

Uses crt;
Begin
  textbackground(green); clrscr; {ekran yashil rangda tozalandi}
  textcolor(blue); textbackground(3); GotoXY(5,5);
  writeln('Suv - hayot manbai'); {matn billur rangda ko'k fonda 5-
ustun satr kesishgan joydan chiqariladi}
  textcolor(red); textbackground(3); GotoXY(10,12);
  writeln('Suv - hayot manbai'); {matn qizil rangda billur fonda 10-
ustun va 12-satr kesishgan joydan chiqariladi}
  textcolor(green); textbackground(blue); GotoXY(15,20);
  writeln('Suv - hayot manbai'); {matn ko'k rangda yashil fonda 15-
ustun va 20-satr kesishgan joydan chiqariladi}
  textcolor(13); textbackground(6); GotoXY(18,18);
  writeln('Suv - hayot manbai'); {matn pushti rangda och jigarrang

```

```
fonda 18-ustun va satr kesishgan joydan chiqariladi}
textcolor(6); textbackground(13); GotoXY(30,24);
writeln('Suv - hayot manbai'); {matn och jirarrang rangda pushti
fonda 30-ustun va 24-satr kesishgan joydan chiqariladi}
Readln;
end.
```

Natijaning ekrandagi ko'rinishi:



M-5. Ekranga 5 ta sinfdoshingizi ismini turli ranglarda va ekraning sariq rangida chiqaring.

Yechim:

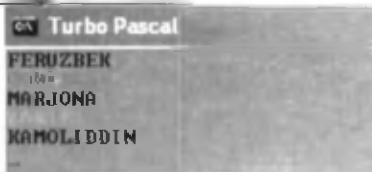
Uses crt;

Begin

```
textbackground(yellow); clrscr; {ekran sariq rangda tozalandi}
textcolor(blue); writeln('FERUZBEK');
textcolor(Green); writeln('QODIRALI ');
textcolor(red); writeln('MARJONA ');
textcolor(13); writeln('ZARIFA');
textcolor(blue); writeln('KAMOLIDDIN'); readln;
```

end.

Natijaning ekrandagi ko'rinishi:



27 – dars. Chiziqli dasturlar tuzish

Odatda, chiziqli algoritmlarning dastur shaklida yozilishi **chiziqli dastur** deb ataladi. Demak, chiziqli dasturdagi barcha operatorlar kelish tartibida ketma-ket bajariladi va xech qanday shart tekshirilmaydi.

M-1. Quyidagi chiziqli dasturlarni ishini izohlang va natijasini aniqlang.

<p>a) Var a, b : String; Begin a:= 'O'zbekiston *; b:= 'Davlat*; WriteLn(a,'Mustaqil ', b); End.</p>	<p><u>Izoh:</u> Bu dastur chiziqli bo'lib, quyidagi amallarni bajaradi: 1) a satrli o'zgaruvchiga "O'zbekiston" qiymatini o'zlashtiradi; 2) b satrli o'zgaruvchiga "Davlat" qiymatini o'zlashtiradi; 3) Xotiradagi va qo'shimcha ma'lumotlarni ekranga chiqaradi.</p>
<p><u>Natijaning ekrandagi ko'rinishi:</u> O'zbekiston Mustaqil Davlat</p>	

<p>b) var a,b:Integer; s: Real; Begin a:=4; a:=sqr(a); b:=b-a; s:=2*a+3*b; WriteLn('S= ',s); End.</p>	<p><u>Izoh:</u> Bu dastur chiziqli bo'lib, quyidagi amallarni bajaradi: 1) a butun o'zgaruvchiga 4 ni o'zlashtiradi; 2) a o'zgaruvchi qiymatini kvadratga oshirib yana a o'zgaruvchiga o'zlashtiradi; 3) b butun o'zgaruvchi qiymatidan (quymat berilmagani uchun b: =0) a o'zgaruvchi qiymatini ayirib b o'zgaruvchiga o'zlashtiradi; 4) a o'zgaruvchi qiymatini 2 ga, b o'zgaruvchi qiymatini 3 ga ko'paytiradi va hosil bo'lgan ko'paytmalarni yig'indisini s haqiqiy o'zgaruvchiga o'zlashtiradi; 5) Ekranda "S= " matn va uni davomidan s haqiqiy o'zgaruvchi qiymatini aks ettiradi.</p>
<p><u>Natijaning ekrandagi ko'rinishi:</u> S= -1.6000000000E+01</p>	

M-2. Quyida trapetsiya yuzini hisoblash dasturi tartibsiz yozilgan. Operatorlarni mantiqan to'g'ri ketma-ketlikda joylashtiring.

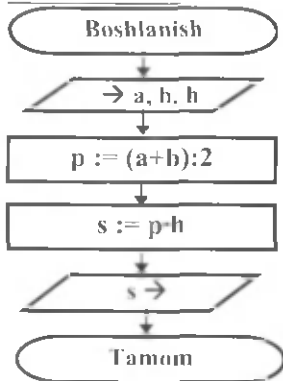
```

p := (a+b)/2; s := p * h; Program trapesiya_yuzi; End.
WriteLn('S =', s, 'kvadrat birlik'); ReadLn(a,b,h);
Begin Write('A,B,H qiymatlarini kiriting: ');
Var a,b,h:Integer; p,s:Real;

```

Javob: Dastur chiziqli bo'ladi. Dastur va unga mos blok sxema hosil qilamiz.

Blok sxemasi:



Dasturi:

```

Program trapesiya_yuzi;
Var
  a,b,h:Integer;
  p,s:Real;
Begin
  Write('a,b,h qiymatlarini kiriting: ');
  ReadLn(a,b,h);
  p := (a+b)/2;
  s := p * h;
  WriteLn('S =', s, 'kvadrat birlik');
End.

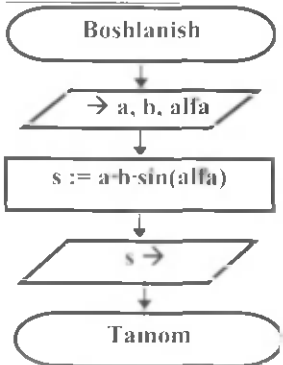
```

M-3. Uchburchakning a, b tomonlari va ular orasidagi α burchak berilgan. Uchburchakning yuzini hisoblash dasturini tuzing (yo'llanma: $s = \frac{1}{2} \cdot a \cdot b \cdot \sin\alpha$).

Yechim:

Berilgan ko'rsatmaga ko'ra dastur chiziqli bo'ladi. Dastur va unga mos blok sxema hosil qilamiz

Blok sxemasi:



Dasturi:

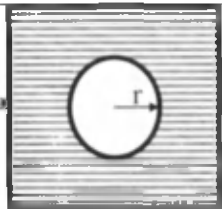
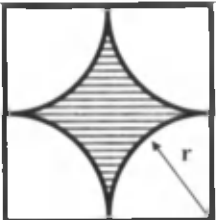
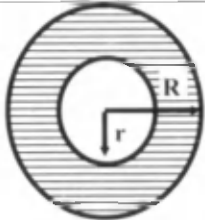
```

Program uchburchak_yuzi;
Var
  a,b,alfa,s:Real;
Begin
  Write('a= '); ReadLn(a);
  Write('b= '); ReadLn(b);
  Write('alfa burchak= '); ReadLn(alfa);
  s := a*b*sin(alfa)/2;
  WriteLn('Uchburchak yuzi S =',
  s, 'kvadrat birlik');
End.

```


28 – dars. Chiziqli dasturlar tuzish mavzusini takrorlash mavzusiga

M-1. Quyida berilgan shakllarning shtrixlangan qismlarini yuzalarini hisoblash dasturini tuzing (yo'llanma: qanday shakl yuzasidan qaysi shakl yuzasini ayirish kerak?).

 <p style="text-align: center;">b</p>		
<p><u>Yechim:</u> Rasmdan ko'rinib turibdiki, to'g'ri to'rtburchakni yuzidan doirani yuzini ayirish kerak. Ma'lumki, to'g'ri to'rtburchakni yuzi $S1=a \cdot b$, doirani yuzi $S2=\pi \cdot r^2$. U holda kerakli yuza: $S = a \cdot b - \pi \cdot r^2$.</p>	<p><u>Yechim:</u> Agar rasmdagi to'rtta chekkadagi bo'laklarni yig'sak, to'liq doira hosil bo'ladi. Demak, kvadratni yuzidan doirani yuzini ayirish kerak. Ma'lumki, kvadratni yuzi $S1=a^2$, doirani yuzi $S2=\pi \cdot r^2$. U holda kerakli yuza: $S = a^2 - \pi \cdot r^2$.</p>	<p><u>Yechim:</u> Rasmdan ko'rinib turibdiki, tashqi doirani yuzidan ichki doirani yuzini ayirish kerak. Ma'lumki, tashqi doirani yuzi $S1=\pi \cdot R^2$, ichki doirani yuzi $S2=\pi \cdot r^2$. Lekin, Paskalda R va r bitta identifikator hisoblanadi, shuning uchun kerakli yuzani quyidagicha yoza-miz: $S = \pi \cdot R1^2 - \pi \cdot R2^2$.</p>
<p><u>Dasturi:</u></p>	<p><u>Dasturi:</u></p>	<p><u>Dasturi:</u></p>
<p>Var a,b,r,s: Real; Begin Write('a= '); ReadLn(a);</p>	<p>Var a,r,s: Real; Begin Write('a= '); ReadLn(a);</p>	<p>Var r1,r2,s: Real; Begin Write('R= '); ReadLn(R1);</p>

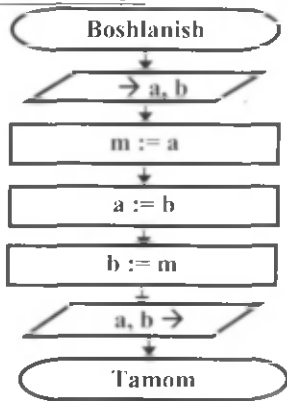
<pre>Write('b= '); ReadLn(b); Write('r= '); ReadLn(r); s:= a*b-pi*sqr(r); WriteLn('S =', s); End.</pre>	<pre>Write('r= '); ReadLn(r); s:=sqr(a)-pi*sqr(r); WriteLn('S =', s); End.</pre>	<pre>Write('r= '); ReadLn(R2); s:=pi*sqr(R1)- pi*sqr(R2); WriteLn('S =', s); End.</pre>
---	--	---

M-2. Berilgan **a** va **b** butun sonlarni qiymatini almashtiruvchi dastur tuzing, ya'ni $a=7$ va $b=2$ kiritilsa, $a=2$ va $b=7$ natija chiqsin (yo'llanma: o'rin almashtirish $m = a, a = b, b = m$).

Yechim:

Agar qo'shimcha o'zgaruvchi kiritmasdan faqat **a:=b; b:=a;** operatorlari bajarilsa, u holda **a:=b;** o'zlashtirish operatori bajarilganda **a** o'zgaruvchining qiymati xotiradan o'chadi va o'rniga **b** o'zgaruvchining qiymati yoziladi. Yo'llanmadan ko'rinib turibdiki, o'zgaruvchilarni o'rnini almashtirish uchun qo'shimcha o'zgaruvchidan foydalanish kerak. Dastur va unga mos blok sxema hosil qilamiz.

Blok sxemasi:



Dasturi:

```

Var
a,b,m: Real;
Begin
Write('a= '); ReadLn(a);
Write('b= '); ReadLn(b);
m := a;
a := b;
b := m;
WriteLn('a =', a, 'b= ', b);
ReadLn;
End.
```

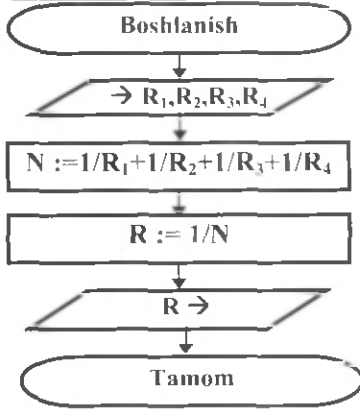
M-3. Mos ravishda **R₁**, **R₂**, **R₃**, **R₄** qarshilikka ega bo'lgan o'tkazgichlar parallel ulanganda hosil bo'ladigan **R** qarshilikni hisoblash dasturini tuzing (yo'llanma: $1/R=1/R_1+1/R_2+1/R_3+1/R_4$).

Yechim:

Fizika fanidan ma'lumki, o'tkazgichlar parallel ulanganda hosil bo'ladigan natijaviy qarshilik yo'llanmada berilgan formula bilan hi-

soblanadi. Albatta, ixtiyoriy o'ztkazgich noldan farqli qarshilikka ega-dir. Formulada natijaviy qarshilik R nisbat shaklida berilgan. Shuning uchun avval N nisbatni, so'ng natijaviy qarshilikni hisoblab olish qulay. Dastur va unga mos blok sxema hosil qilamiz.

Blok sxemasi:



Dasturi:

```

Var
  R1, R2, R3, R4, N, R: Real;
Begin
  Write('R1= '); ReadLn(R1);
  Write('R2= '); ReadLn(R2);
  Write('R3= '); ReadLn(R3);
  Write('R4= '); ReadLn(R4);
  N := 1/R1 + 1/R2 + 1/R3 + 1/R4;
  R := 1/N;
  WriteLn('Natijaviy qarshilik R
= ', R);
End.
  
```

29 – dars. O'tish va tarmoqlanish operatorlari mavzisiga

M-1. Quyidagi o'tish operatorlaridan xato yozilganini aniqlang.

a) Goto 10;	To'g'ri yozilgan
b) goto 30;	To'g'ri yozilgan
d) goto -5;	Xato, chunki nishonlarda ishlatiladigan son 0 dan 9999 gacha bo'la oladi
e) GoTo 5;	To'g'ri yozilgan
f) goto sin;	To'g'ri yozilgan, chunki sin zahira so'z emas
g) goto 2_5;	Xato, chunki nishonlarda lotin harflari va 0 dan 9999 gacha bo'lgan sonlar ishlatiladi, 2_5 bu holatlarga to'g'ri kelmaydi, chunki tagchziq belgisi ikkita sonni ajratib turibdi
h) GOTO a 5;	To'g'ri yozilgan

M-2. Tarmoqlanish operatori uchun quyidagi taqqoslash shartlaridan xato yozilganini toping.

a) a<>b;	To'g'ri yozilgan
b) a<-b;	To'g'ri yozilgan
d) a<<b;	Xato yozilgan, chunki "teng emas" sharti <> kabi yoziladi (Paskal alifbosi asosida)
e) -a>0;	To'g'ri yozilgan
f) -1>0;	To'g'ri yozilgan, natija False
g) a>>b;	Xato yozilgan, chunki "katta" sharti > kabi yoziladi (Paskal alifbosi asosida)
h) a:=b;	Xato yozilgan, chunki taqqoslashda o'zlashtirish operatori ishlatilmaydi

M-3. Quyidagilardan xato yozilganini toping.

a) IF a=b THEN a:=a+1; ELSE b:=a;	Xato yozilgan, chunki Else dan oldin ";" yozilmaydi
b) IF a:=1 THEN a:=a+1 ELSE b:=a;	Xato yozilgan, chunki taqqoslashda o'zlashtirish operatori ishlatilmaydi

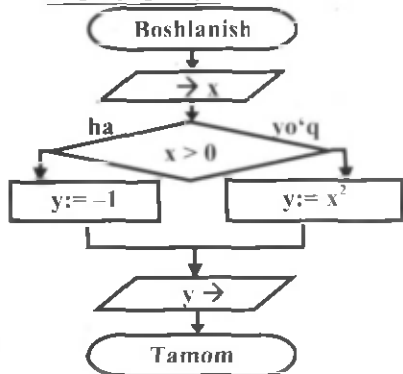
M-4. x ning berilgan qiymatida $y = \begin{cases} -1, & \text{agar } x > 0 \\ x^2, & \text{agar } x \leq 0 \end{cases}$ funksiyasi

qiymatini hisoblash dasturini tuzing.

Yechim:

Funksiyani aniqlanish formulasi haqiqiy x o'zgaruvchining qiymati aniqlangan sohaga bog'liq bo'lmoqda, ya'ni x ning qiymatiga qarab, y funksiyaga berilgan funsiyalardan birining qiymati o'zlashtiriladi: agar $x > 0$ shart bajarilsa $y = -1$, aks holda $y = x^2$. Demak, dasturda tarmoqlanish operatoridan foydalanib shart tekshirish talab etiladi. Ma'lumki, shart tekshiriladigan algoritm tarmoqlanuvchi sturkturali algoritmlar deb ataladi.

Blok sxemasi:



Dasturi:

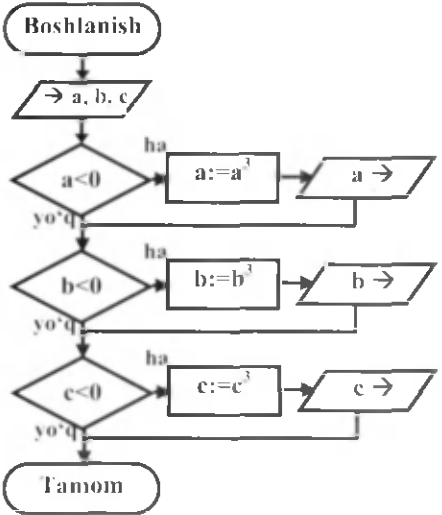
```

Program Taqqoslash1;
Var
  x,y: Real;
Begin
  Write('x ni qiymatini kiriting: ');
  ReadLn(x);
  If x>0 Then y:= - 1
    Else y:= sqr(x);
  WriteLn('y= ', y:0:3);
  ReadLn;
End.
  
```

M-5. Uchta son berilgan. Ular ichida manfiy sonlarning kubini hisoblovchi dastur tuzing.

Yechim:

Bu masalada sonni manfiyligini aniqlash uchun manfiy bo'lish sharti – sonni 0 dan kichikligini tekshirish, ya'ni tarmoqlanish operatoridan foydalanish zarur. Masala ichida ko'rinmaydigan kichik masalacha bor: agar birorta ham manfiy son kiritilmasa, dastur qanday javob chiqaradi? Bu masalachani hal etish uchun dasturda **bor** nomli mantiqiy o'zgaruvchi kiritamiz va **true** qiymat beramiz. Agar kiritilgan sonlardan birortasi manfiy bo'lsa, u holda **bor** mantiqiy o'zgaruvchiga **false** qiymat beramiz, bu holda dastur oxiridagi tarmoqlanish operatorida shart **yolg'on** bo'ladi va Write('Manfiy son kiritilmadi'); operatori bajarilmaydi. Blok sxemada bu holni aks ettirmaymiz.

Blok sxemasi:	Dasturi:
 <pre> graph TD Start([Boshlanish]) --> Input[/→ a, b, c/] Input --> DecA{a < 0} DecA -- ha --> ProcA[a := a^3] ProcA --> OutA[/a →/] DecA -- yo'q --> DecB{b < 0} DecB -- ha --> ProcB[b := b^3] ProcB --> OutB[/b →/] DecB -- yo'q --> DecC{c < 0} DecC -- ha --> ProcC[c := c^3] ProcC --> OutC[/c →/] DecC -- yo'q --> End([Tamom]) </pre>	<p>Dasturi:</p> <pre> Var a, b, c: Real; bor: boolean; BEGIN bor:=true; Write('a , b, c kiriting: '); ReadLn(a,b,c); IF a<0 Then begin bor:= false; a:=sqr(a)*a; Write('a=',a); end; IF b<0 Then begin bor:= false; b:=sqr(b)*b; Write('b=',b); end; IF c<0 Then begin bor:= false; c:=sqr(c)*c; Write('c=',c); end; If bor then Write('Manfiy son kiritilmadi'); END. </pre>

M-6. Berilgan butun son manfiy bo'lsa uning modulini hisoblash dasturi tuzilsin. Dasturni 2 xil usulda, modul hisoblash uchun $\text{abs}(x)$ funksiyasidan foydalanib va foydalanmasdan, tuzing.

Yechim:

Bu masalada ikki xil talab qo'yilgan.

Agar birinchi talab bo'yicha $\text{abs}(x)$ funksiyasidan foydalanilsa, dastur chiziqli bo'ladi, chunki bu funksiya manfiy sonni ham musbat sonni ham musbat songa, 0 ni 0 ga o'tkazadi. Shuning uchun shart tekshirishni keragi bo'lmaydi.

Agar ikkinchi talab bo'yicha $\text{abs}(x)$ funksiyasidan foydalanilmasa, u holda sonni manfiyligi tekshirish va agar son manfiy bo'lsa, u holda sonni teskari ishora bilan o'zlashtirish kerak bo'ladi. Demak, bu holda tarmoqlanish operatoridan foydalanish zarur.

Dasturi: 1-hol:

```

var x: Real;
BEGIN
Write('Sonni kiriting: ');
ReadLn(x);
  
```

Dasturi: 2-hol:

```

var x: Real;
BEGIN
Write('Sonni kiriting: ');
ReadLn(x);
  
```

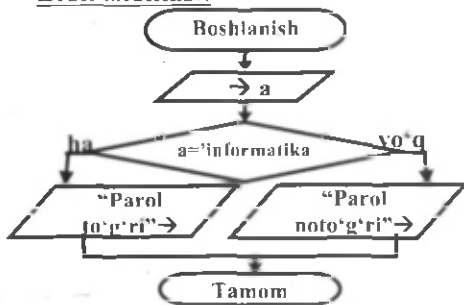
<code>x:=abs(x); Write('Javob: ',x); END.</code>	<code>If x<= then x:=-x; Write('Javob: ',x); END.</code>
--	---

M-7. Parol “informatika” bo’lsa, u holda parol to’g’ri kiritilganini tekshiruvchi dastur tuzing.

Yechim:

Masalada kiritilayotgan matn “informatika” matni bilan bir xilligini tekshirish zarur. Albatta, kiritilayotgan matnni xotirada saqlash uchun **a** satrli o’zgaruvchi aniqlanadi. Shart tekshirish natijasiga ko’ra ekranga “Parol to’g’ri” yoki “Parol noto’g’ri” javoblaridan biri chiqariladi.

Blok sxemasi:



Dasturi:

```

var a: String;
BEGIN
Write('Parolni kiriting: ');
ReadLn(a);
If a = 'informatika' then
write('Parol to'g'ri') else
write('Parol noto'g'ri');
readln;
END.
  
```

30 – dars. O‘tish va tarmoqlanish operatorlari mavzusini takrorlash mavzusiga

M-1. Berilgan qiymat asosida shart qiymatini aniqlang.

<p>a) $a:=10; b:=a*3$; shart: “$a<b/3$”;</p> <p><u>Yechim:</u> $a=10$ va $b=10*3=30$ bo’lgani uchun $a<b/3 \Leftrightarrow 10<30:3 \Leftrightarrow 10<10$, bu esa “Yolg’on”</p>	<p>b) $a:=10; b:=a*3$; shart: “$a<=b/3$”;</p> <p><u>Yechim:</u> $a=10$ va $b=10*3=30$ bo’lgani uchun $a<=b/3 \Leftrightarrow 10<=30:3 \Leftrightarrow 10<=10$, bu esa “Rost”</p>
<p>d) $a:=10; b:=a$; shart: “$a+b=2*b$”;</p> <p><u>Yechim:</u> $a=10$ va $b=10$ bo’lgani uchun $a+b=2*b \Leftrightarrow 10+10=2*10 \Leftrightarrow 20=20$, bu esa “Rost”</p>	<p>e) $a:=10; b:=a+3$; shart: “$a+3>=b-3$”;</p> <p><u>Yechim:</u> $a=10$ va $b=10+3=13$ bo’lgani uchun $a+3>=b-3 \Leftrightarrow 10+3>=13-3 \Leftrightarrow 13>=10$, bu esa “Rost”</p>

M-2. Tarmoqlanish natijasida hosil bo'ladigan qiymatlarni aniqlang.

a) aa:=7; bb:=6.6; if aa=round(bb) then mm:='Ha' else mm:='Yo'q';

Yechim:

aa=7, bb=6.6 bo'lgani va **round** funksiyasi sonni yaxlitlagani uchun aa=round(bb) \Leftrightarrow 7=round(6.6) \Leftrightarrow 7=7, bu esa "Rost". Shuning uchun **mm** satrli o'zgaruvchi '**Ha**' qiymatni o'zlashtiradi.

b) ag:=true; if ag then aa:=21 else aa:=7; a:=a+1963;

Yechim:

ag=true, ya'ni "Rost". Shuning uchun **aa** sonli o'zgaruvchi tarmoqlanish operatori ishi natijasida **21** qiymatni o'zlashtiradi. Keyin tarmoqlanish operatoridan tashqaridagi o'zlashtirish operatori ishi natijasida a=a+1963=21+1963=1984 bo'ladi.

d) ag:=true; if ag then aa:=21 else begin aa:=7; a:=a+1963; end;

Yechim:

ag=true, ya'ni "Rost". Shuning uchun **aa** sonli o'zgaruvchi tarmoqlanish operatori ishi natijasida **21** qiymatni o'zlashtiradi.

e) ms:=50; aa:=10; if ms div aa = aa*5 then ms:=trunc(ms/3) else aa:=ms mod aa;

Yechim:

ms=50, aa=10, **div** butun bo'lish amali bo'lgani uchun

ms div aa=aa*5 \Leftrightarrow 50 div 10=10*5 \Leftrightarrow 5=50, bu esa "Yolg'on". Endi **mod** qoldiqni aniqlagani uchun aa=ms mod aa=50 mod 10=0 bo'ladi.

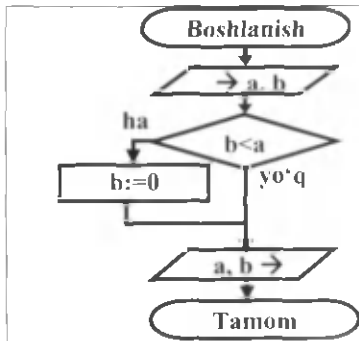
M-3. Ikkita a va b sonlar berilgan. Agar b son a dan kichik bo'lsa, u holda b ni nol bilan almashtiruvchi, aks holda b o'zgarishsiz qoldiruvchi dastur tuzing.

Yechim:

Bu masala algoritmi avvalgi bobning 10-darsida tuzilgan edi. Masalada faqat bir holatda, ya'ni b<a shart bajarilganda amallar bajarilishi talab etgani uchun tarmoqlanish tuzilmasini qisqa ko'rinishidan foydalanilgan edi. Shu sababli masalaga mos dasturda tarmoqlanish operatorining qisqa ko'rinishidan foydalanish maqsadga muvofiq. Odatda, masala shartida a va b o'zgaruvchilarni butun turdali aytilmaganligi uchun haqiqiy turdagi deb qaraladi.

Blok sxemasi:

Dasturi:



var

a, b: real;

BEGIN

Write('a ni kiriting: '); ReadLn(a);

Write('b ni kiriting: '); ReadLn(b);

If b < a then b := 0;

writeln('a= ', a:0:8, 'b= ', b:0:8);

readln;

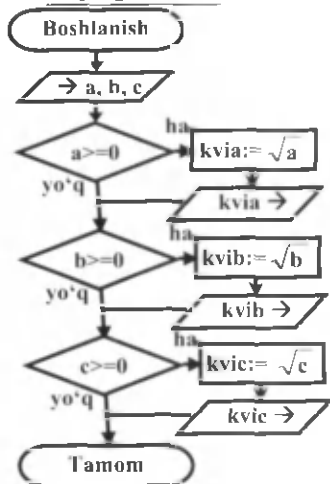
END.

M-4. Uchta a, b, c sonlar berilgan. Bu sonlardan faqat musbatlarini kvadrat ildizini hisoblab chiqaruvchi dastur tuzing.

Yechim:

Bu masala ham avvalgi bobning 10-darsida ko'rib o'tilgan. Bu masalaga mos dasturda ham tarmoqlanish operatorining qisqa ko'rinishidan foydalanish maqsadga muvofiq. Masala algoritmidan ko'rish mumkinki, shart o'rinli bo'lganda 2 ta amal bajariladi, shu sababli dasturda **then** xizmatchi so'zidan keyin **begin** va **end**; xizmatchi so'zlari yozilgan.

Blok sxemasi:



Dasturi:

var

a, b, c, kvia, kvib, kvic: real;

BEGIN

Write('a ni kiriting: '); ReadLn(a);

Write('b ni kiriting: '); ReadLn(b);

Write('c ni kiriting: '); ReadLn(c);

If a >= 0 then begin kvia := sqrt(a);

writeln('a uchun ', kvia:0:8); end;

If b >= 0 then begin kvib := sqrt(b);

writeln('b uchun ', kvib:0:8); end;

If c >= 0 then begin kvic := sqrt(c);

writeln('c uchun ', kvic:0:8); end;

readln;

END.

Masala yechimini boshqacha usullarda ham yozish mumkin, masalan, kvia, kvib, kvic kabi o'zgaruvchilar yangi kiritmasdan ham.

Dasturi:

var a, b, c: real;

BEGIN

Write('a ni kiriting: '); ReadLn(a);

Write('b ni kiriting: '); ReadLn(b);

Write('c ni kiriting: '); ReadLn(c);

If a>=0 then writeln('a ni kvadrat ildizi ', sqrt(a):0:8);

If b>=0 then writeln('b ni kvadrat ildizi ', sqrt(b):0:8);

If c>=0 then writeln('c ni kvadrat ildizi ', sqrt(c):0:8); readln;

END.

M-5. $ax + b = 0$ tenglamaning ildizini a, b ning quyidagi qiymatlarida hisoblash dasturini tuzing.

a) $a = -1, b = 1$;

b) $a = 0, b = 4$;

d) $a = 1, b = 0$;

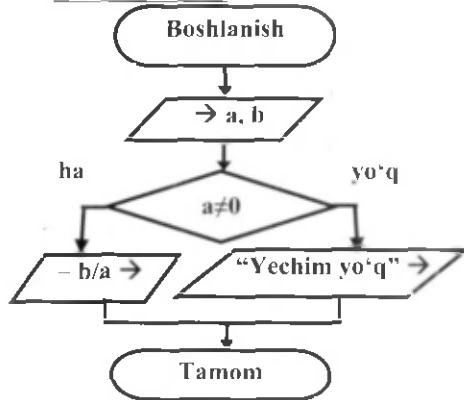
e)

$a = 1, b = -5$.

Yechim:

Matematika kursidan ma'lumki, bu tenglama yechimi mavjud bo'lishi uchun $a \neq 0$ shart bajarilishi kerak va bu holda yechim quyidagi ko'rinishda aniqlanadi: $x = -b/a$, aks holda tenglamaning yechimi mavjud emas. Demak, bu masalada tarmoqlanish operatorining to'liq ko'rinishidan foydalanish maqsadga muvofiq.

Blok sxemasi:



Dasturi:

var a, b: real;

BEGIN

Write('a ni kiriting: ');

ReadLn(a);

Write('b ni kiriting: ');

ReadLn(b);

If a<>0 then

write('x = ', -b/a:0:5)

else

write('Yechim yo'q');

readln;

END.

Natijaning ekrandagi ko'rinishi:

```

a ni kiriting: -1
b ni kiriting: 1
x = 1.000000
a ni kiriting: 0
b ni kiriting: 4
Yechim yo'q
a ni kiriting: 1
b ni kiriting: 0
x = 0.000000
a ni kiriting: 1
b ni kiriting: -5
x = 5.000000_

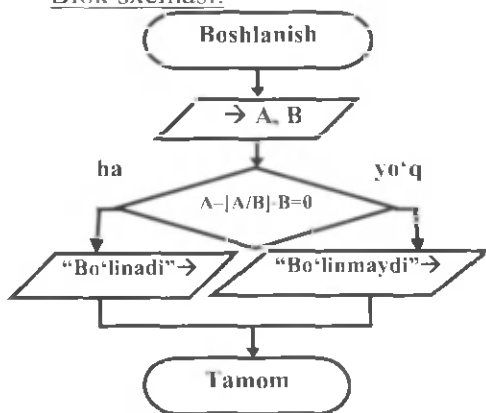
```

M-6. Berilgan A butun son berilgan noldan farqli B butun songa qoldiqsiz bo'linishi yoki bo'linmasligini aniqlovchi dastur tuzing.

Yechim:

Matematika kursidan ma'lumki, A sonni B songa bo'lgandagi qoldiqni quyidagi formula yordamida hisoblash mumkin: $qol = A - [A/B] \cdot B$. Lekin Pascal dasturlash tilida A sonni B songa bo'lgandagi qoldiqni **mod** amali yordamida hisoblash mumkin: $qol = A \bmod B$. Endi qoldiqni 0 ga teng yoki teng emasligini tarmoqlanish operatorining to'liq ko'rinishidan foydalanib aniqlash mumkin. Masalada B sonning 0 ga teng emasligi bo'lish amali yoki qoldiq uchun muhim.

Blok sxemasi:



Dasturi:

```

var A, B: real;
BEGIN
  Write('A ni kiriting: ');
  ReadLn(A);
  Write('B ni kiriting: ');
  ReadLn(B);
  If A mod B=0 then
    write('Bo‘linadi')
  else
    write('Bo‘linmaydi ');
  readln;
END.

```

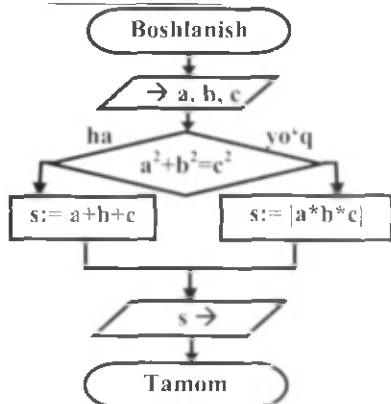
M-7. Uchta a, b, c sonlar berilgan. Agar $a^2 + b^2 = c^2$ shart bajarilsa, bu sonlarning yig'indisini, aks holda ularni modullari ko'paytmasini hisoblovchi dastur tuzing.

Yechim:

Masalada a, b, c sonlar haqida hech qanday chegaralash yo'q (ma-

salan, butun, natural kabi). Shuning uchun ularni haqiqiy turdagi sonli o'zgaruvchilar deb olish mumkin. Yechimda tarmoqlanish operatorining to'liq ko'rinishidan foydalanish qulay. Dastur tuzayotganda matematika kursidan ma'lum bo'lgan quyidagi tenglikdan foydalanish mumkin: $|a| \cdot |b| \cdot |c| = |a \cdot b \cdot c|$.

Blok sxemasi:



Dasturi:

```

Var a, b, c: Real;
Begin
  Write('a ni kiriting: ');
  ReadLn(a);
  Write('b ni kiriting: ');
  ReadLn(b);
  Write('c ni kiriting: ');
  ReadLn(c);
  If sqr(a)+sqr(b)=sqr(c) Then
    s:=a+h+c Else s:= abs(a*b*c);
  WriteLn('Javob: ', s:0:5);
End.
  
```

31 – dars. Tarmoqlanuvchi strukturali dasturlar tuzish mavzusiga

M-1. Quyidagi berilgan operatorlardagi xatolarni aniqlang va izohlang.

<p>a) IF d>0 THEN 63 ELSE s:=d+a;</p> <p><u>Yechim:</u> Tarmoqlanish operatorida shart to'g'ri yozilgan. Lekin THEN xizmatchi so'zidan keyin o'tish operatori xato yozilgan, chunki Goto yozilmagan. ELSE xizmatchi so'zidan keyin o'zlashtirish operatori to'g'ri yozilgan.</p>	<p>b) IF s1<>s2 THEN ELSE g1:=s1*s2;</p> <p><u>Yechim:</u> Tarmoqlanish operatorida shart to'g'ri yozilgan. THEN xizmatchi so'zidan keyin operator yozilmagan, bu esa xato emas – bu holda hech qanday amal bajarilishi talab qilinmagan. ELSE xizmatchi so'zidan keyin o'zlashtirish operatori to'g'ri yozilgan.</p>
<p>d) IF i*j THEN goto vo ELSE goto ne;</p>	<p>e) IF x<>0 AND x<=5 THEN y=4*sin(x);</p>

<p><u>Yechim:</u> Tarmoqlanish operatorida shart xato yozilgan, chunki mantiqiy qiymat hosil qiluvchi ifoda yo'q. THEN va ELSE xizmatchi so'zlaridan keyingi o'tish operatorlari to'g'ri yozilgan.</p>	<p><u>Yechim:</u> Tarmoqlanish operatorida murakkab shart xato yozilgan, chunki mantiqiy amallar yordamida yozilgan shartlarning har biri qavs ichiga olinishi shart. THEN xizmatchi so'zidan keyingi o'zlashtirish operatorida xato bor: tenglik belgisidan avval "=" belgisi yozilishi kerak.</p>
--	---

M-2. Quyidagi mantiqiy ifodalardagi amallarning bajarilish tartibini aniqlang.

<p>a) $a < -6 \text{ OR } a \geq 0 \text{ AND } a < 4$; <u>Yechim:</u> 1) $(a \geq 0) \text{ AND } (a < 4)$ 2) $(a < -6) \text{ OR } ((a \geq 0) \text{ AND } (a < 4))$</p>	<p>b) $x * x + y > 0 \text{ AND } a = 0.1 \text{ OR } (b > 3.7 \text{ AND } s < k4)$; <u>Yechim:</u> 1) $(b > 3.7) \text{ AND } (s < k4)$ 2) $(x * x + y > 0) \text{ AND } (a = 0.1)$ 3) $((x * x + y > 0) \text{ AND } (a = 0.1)) \text{ OR } ((b > 3.7) \text{ AND } (s < k4))$</p>
<p>d) $v = 'ha' \text{ AND } x1 > 0 \text{ AND } x2 > 0$; <u>Yechim:</u> 1) $(v = 'ha') \text{ AND } (x1 > 0)$ 2) $((v = 'ha') \text{ AND } (x1 > 0)) \text{ AND } (x2 > 0)$</p>	<p>e) $a > 0 \text{ OR } a < 1 \text{ OR NOT } x * x + x * x <= 1$; <u>Yechim:</u> 1) $\text{NOT } (x * x + x * x <= 1)$ 2) $(a > 0) \text{ OR } (a < 1)$ 3) $((a > 0) \text{ OR } (a < 1)) \text{ OR } (\text{NOT } (x * x + x * x <= 1))$</p>
<p>f) $\text{NOT } v <= b \text{ AND } (f <= f1 \text{ OR } t = '.')$; <u>Yechim:</u> 1) $(f <= f1) \text{ OR } (t = '.')$ 2) $\text{NOT}(v <= b)$ 3) $(\text{NOT}(v <= b)) \text{ AND } ((f <= f1) \text{ OR } (t = '.'))$</p>	<p>g) $\text{NOT}(\text{NOT}(\text{NOT}(a > b) \text{ OR TRUE}) \text{ AND FALSE})$; <u>Yechim:</u> 1) $\text{NOT}(a > b)$ 2) $\text{NOT}(a > b) \text{ OR TRUE}$ 3) $\text{NOT}(\text{NOT}(a > b) \text{ OR TRUE})$ 4) $\text{NOT}(\text{NOT}(a > b) \text{ OR TRUE}) \text{ AND FALSE}$ 5) $\text{NOT}(\text{NOT}(\text{NOT}(a > b) \text{ OR TRUE}) \text{ AND FALSE})$</p>

M-3. Uzunliklari orqali berilgan uchta kesmadan uchburchak hosil qilish mumkin yoki mumkin emasligini aniqlovchi dastur tuzing.

Yechim:

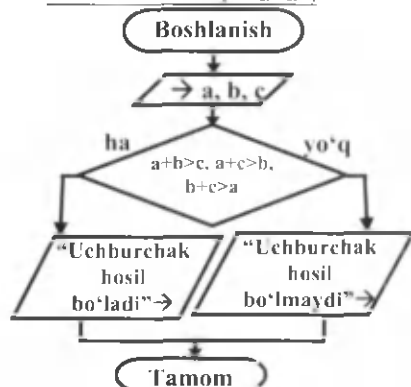
Masala shartidagi kesmalarni a, b va c haqiqiy sonli o'zgaruvchilar orqali ifodalash mumkin. Geometriya kursidan ma'lumki, uchbur-

chakning ixtiyoriy ikki tomoni yig'indisi uchinchil tomonidan katta bo'ladi, bu uchburchak hosil bo'lish shartidir: $a+b>c$ va $a+c>b$ va $b+c>a$.

1-usul: Bu uchala tengsizlik bir vaqtda bajarilishi shart, shuning uchun dasturda AND mantiqiy amalidan foydalaniladi.

2-usul. Har bir shartni alohida tarmoqlanish operatoridan foydalanib tekshirish mumkin.

Blok sxemasi (1-usul)

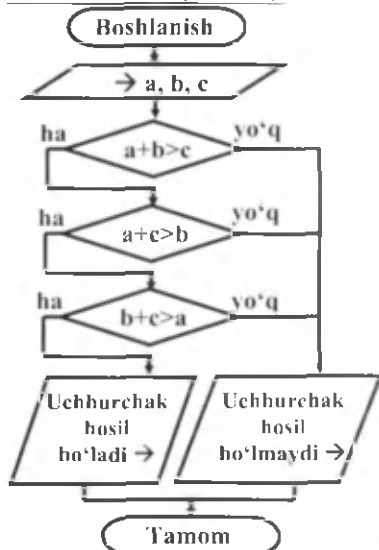


Dasturi (1-usul):

```

Var a, b, c: Real;
Begin
Write('a ni kiriting: '); ReadLn(a);
Write('b ni kiriting: '); ReadLn(b);
Write('c ni kiriting: '); ReadLn(c);
If (a+b>c)and(a+c>b)and(b+c>a)
Then WriteLn('Uchburchak hosil boladi')
Else WriteLn('Uchburchak hosil bo'lmaydi');
End.
    
```

Blok sxemasi (2-usul):



Dasturi (2-usul-A):

```

Label 1;
Var a, b, c: Real;
Begin
Write('a ni kiriting: '); ReadLn(a);
Write('b ni kiriting: '); ReadLn(b);
Write('c ni kiriting: '); ReadLn(c);
If a+b>c then
If a+c>b then
If b+c>a then Begin
WriteLn('Uchburchak hosil boladi'); goto 1; end;
WriteLn('Uchburchak hosil bo'lmaydi');
1: ReadLn;
End.
    
```

Dasturi (2-usul-B):

```

Label 1, 2, 3, 4;
Var a, b, c: Real;
Begin
  Write('a ni kiriting: '); ReadLn(a); Write('b ni kiriting: ');
  ReadLn(b); Write('c ni kiriting: '); ReadLn(c);
  If a+b>c then goto 2 else goto 1;
  2: If a+c>b then goto 3 else goto 1;
  3: If b+c>a then
  Begin WriteLn('Uchburchak hosil boladi'); goto 4; end;
  1: WriteLn('Uchburchak hosil bo'lmaydi');
  4: ReadLn;
End.

```

M-4. Quyí sinf o'quvchisini ko'paytirish jadvali bo'yicha sinovchi dastur tuzing. To'g'ri javob berilganda "Barakalla", aks holda "Qayta ishla" matnlari turli rangda chiqsin.

Yechim:

Ko'paytirish (karra) jadvalida, odatda, 1 dan 9 gacha bo'lgan sonlar ko'paymasini hisoblangan bo'ladi. Shuning uchun dasturda 1 dan 9 gacha sonlarni hosil qilish uchun **Random(x)** funksiyasidan foydalanish mumkin. Ma'lumki, **Random(9)** funksiyasi funksiyasi [0, 9) oraliqdan (9 kirmaydi) tasodifiy butun sonlarni olib beradi. Shuning uchun [1, 9] oraliqdagi butun sonlarni tasodifan olish uchun **Random(9)+1** ifodani yozish lozim. Dastur har ishlatilganda yangi sonlar hosil qilishi uchun **Randomize** funksiyasini yozish kerak bo'ladi. Dastur ishlaganda hosil bo'ladigan sonlar 0 dan katta va 127 dan kichik bo'lgani uchun o'zgaruvchilar turini **byte** kabi olish maqsadga muvofiq.

Dasturi:

```

Uses crt;
Label 1;
Var a, b, c: byte;
Begin
  1: textbackground(14); clrscr; Randomize; a:=Random(9)+1;
  b:=Random(9)+1; textcolor (0);GotoXY(35,12); Write(a,' * ',b, ' = ');
  textcolor(2); ReadLn(c);
  If a*b=c Then Begin textcolor(7); GotoXY(35,13);
  Writeln('Barakalla!'); readln; End
  Else Begin textcolor (4); GotoXY(34,13);

```

```
WriteLn('Qayta ishla'); readLn; goto I; End;
End.
```

M-5. Kiritilgan I dan 7 gacha oraliqda bo'lgan raqamga asosan hafta kunini ekranga chiqaruvchi dastur tuzing.

Yechim:

Masala shartida kiritilayotgan raqam **byte** turdagi butun o'zgaruvchiga mos. Dasturda tarmoqlanish operatorining qisqa ko'rinishidan foydalanish maqsadga muvofiq.

Dasturi:

```
Label I;
Var a: byte;
Begin Write('Hafta kunini tartib raqamini kiriting: '); ReadLn(a);
  If a=1 Then WriteLn('DUSHANBA');
  If a=2 Then WriteLn('SESHANBA');
  If a=3 Then WriteLn('CHORSHANBA');
  If a=4 Then WriteLn('PAYSHANBA');
  If a=5 Then WriteLn('JUMA');
  If a=6 Then WriteLn('SHANBA');
  If a=7 Then WriteLn('YAKSHANBA'); readLn;
End
```

32 – dars. Tarmoqlanuvchi strukturali dasturlar tuzish mavzusini takrorlash mavzusiga

M-1. Murakkab mantiqiy amallar natijasini aniqlang.

a) a:=true; b:=true; m:=false; bb:=NOT(a AND m) AND (a OR b) OR m;

Yechim: Bajarilayotgan amal kursiv ko'rinishda yozilgan.

bb:= NOT(true AND false) AND (true OR true) OR false = NOT(false) AND true OR false = true AND true OR false = true OR false = true.

b) a:=77; b:=11; m:=7; ms:= (a div b=m) AND (a mod m=0) AND NOT((a>b) OR (b<m));

Yechim: Bajarilayotgan amal kursiv ko'rinishda yozilgan.

ms:= (77 div 11 = 7) AND (77 mod 7 = 0) AND NOT ((77>11) OR (11<7)) =
= (7 = 7) AND (0 = 0) AND NOT (true OR false) =
= true AND true AND NOT (true) = true AND false = false.

M-2. Tarmoqlanish natijasida hosil bo'ladigan qiymatlarni aniqlang.

a) $x := -1; y := 0; a := 0.1; \text{IF } (x*x + y > 0) \text{ AND } (a = 1/10) \text{ THEN } mm := \text{true} \text{ else } mm := \text{false};$

Yechim:

Shart tekshirish natijasi $(x*x + y > 0) \text{ AND } (a = 1/10) \Leftrightarrow$

$\Leftrightarrow ((-1)*(-1) + 0 > 0) \text{ AND } (0.1 = 1/10) \Leftrightarrow (1 + 0 > 0) \text{ AND } (0.1 = 0.1) \Leftrightarrow$

$\Leftrightarrow (1 > 0) \text{ AND true} \Leftrightarrow \text{true AND true} \Leftrightarrow \text{true}.$

U holda **mm:=true.**

b) $x1 := \text{sqr}(-1); v := 'h a'; x2 := \text{sqr}(x1); \text{IF } (v = 'ha') \text{ AND } (x1 > 0) \text{ AND } (x2 > 0) \text{ THEN } x1 := 0;$

Yechim:

Avval $x1$ va $x2$ qiymatlari aniqlanadi:

$x1 = \text{sqr}(-1) = 1; x2 = \text{sqr}(x1) = \text{sqr}(1) = 1;$

Shart tekshirish natijasi $(v = 'ha') \text{ AND } (x1 > 0) \text{ AND } (x2 > 0) \Leftrightarrow$

$\Leftrightarrow ('ha' = 'ha') \text{ AND } (1 > 0) \text{ AND } (1 > 0) \Leftrightarrow \text{true AND true AND true} \Leftrightarrow$

$\Leftrightarrow \text{true AND true} \Leftrightarrow \text{true}$

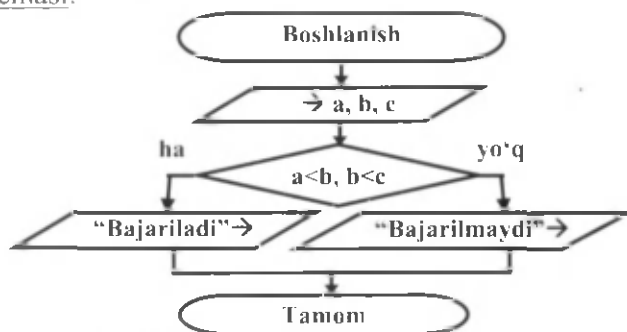
U holda **x1:=0.**

M-3. Uchta a, b, c son berilgan. $a < b < c$ tengsizlikni bajarilish yoki bajarilmasligini tekshiruvchi dastur tuzing.

Yechim:

Masala shartidagi a, b va c sanlar haqiqiy turdagi o'zgaruvchilar bo'ladi. Tekshirish talab etilayotgan $a < b < c$ tengsizlikni Paskal dasturlash tilida $(a < b) \text{ AND } (b < c)$ kabi yozish mumkin.

Blok sxemasi:



Dasturi:

Var a, b, c : Real;

Begin

Write('a ni kiriting: '); ReadLn(a); Write('b ni kiriting: ');

ReadLn(b); Write('c ni kiriting: '); ReadLn(c);

```

If (a<b) and (b<c) Then WriteLn('Bajariladi')
Else WriteLn('Bajarilmaydi ');

```

ReadLn;

End.

M-4. Berilgan A butun son berilgan B butun songa qoldiqsiz bo'lsa bu ikkala sonni yig'indisi kvadratini, aks holda ko'paytmasini chiqaruvchi dastur tuzing.

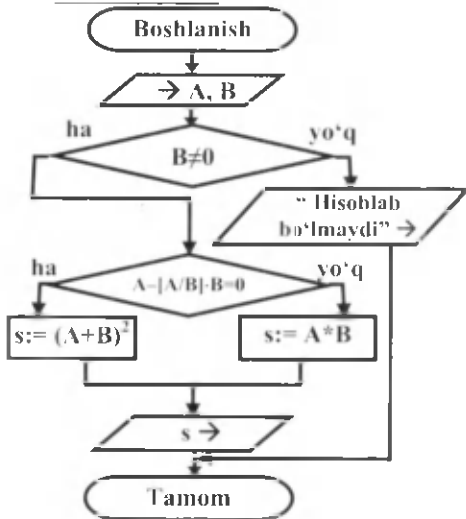
Yechim:

Bu masala sharti 30-dars 6-topshiriqdagi masala shartidan quyidagicha farq qiladi:

1) 30-dars 6-topshiriqda B noldan farqli butun son, demak, bu songa bo'lish mumkin edi.

2) 30-dars 6-topshiriqda faqatgina "Bo'linadi" yoki "Bo'linmaydi" javoblaridan biri chiqadi.

Blok sxemasi:



Dasturi:

```

Var
a, b, s: integer;
Begin
Write('A ni kiriting: ');
ReadLn(A);
Write('B ni kiriting: ');
ReadLn(B);
If B <> 0 Then begin
If A mod B = 0 then
s := sqr(A+B) else s := A * B;
WriteLn('Javob: ', s); end
Else WriteLn('Hisoblab bo'lmaydi');
ReadLn;
End.

```

M-5. Berilgan butun N son musbat va 5 ga karrali bo'lsa, shu sonni kvadrat ildizini, aks holda kvadratini hisoblash dasturini tuzing.

Yechim:

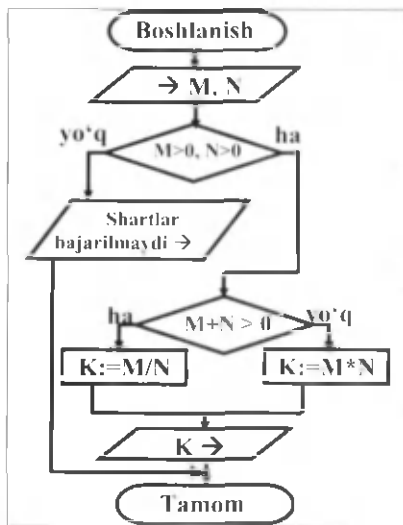
Oldingi bobda K sonini 11 ga karraliligini tekshirish uchun $[K/11]=K/11$ sharti tekshirilgan edi. Bu masalada ham shartni shu kabi yozish mumkin: $[N/5]=N/5$. Algoritm tuzishda bu yoki $N - [N/5] \cdot 5 = 0$

formuladan foydalanish mumkin, lekin Paskal dasturlash tilida bu kabi shartlarni, agar N butun son bo'lsa, $N \bmod 5 = 0$ shart bilan ham almashtirish qulay. Agar N sonining butunligi noma'lum bo'lsa, u holda Paskal dasturlash tilida $\text{INT}(N/5) = N/5$ shartdan foydalanish maqsadga muvofiq.

<p><u>Blok sxemasi:</u></p> <pre> graph TD Start([Boshlanish]) --> Input[/N/] Input --> Decision{N > 0, [N/5] = N/5} Decision -- ha --> Process1[K := sqrt(N)] Decision -- yo'q --> Process2[K := N^2] Process1 --> Output[/K/] Process2 --> Output Output --> End([Tamom]) </pre>	<p><u>Dasturi:</u></p> <pre> Var N: integer; K: Real; Begin Write('N ni kiriting: '); ReadLn(N); If (N > 0) AND (N mod 5 = 0) then K := sqrt(N) else K := N^2; WriteLn('Javob: ', K:0:3); ReadLn; End. </pre>
--	--

M-6. M va N sonlar berilgan. Agar ular musbat va yig'indisi 100 dan katta bo'lsa, M sonni N soniga nisbatini, ular musbat va yig'indisi 100 dan katta bo'lmasa M ni N ga ko'paytmasini hisoblash dasturini tuzing.

<p><u>Yechim:</u></p> <p>Berilgan sonlar M va N haqiqiy turdagi o'zgaruvchilarga o'zlashtiriladi. Masala shartidan ayonki, $M > 0$, $N > 0$ bo'lganda $M + N > 100$ bo'lsa M/N bo'linmani ($N > 0$ bo'lgani uchun mahraj 0 bo'la olmaydi), aks holda, ya'ni $M + N \leq 100$ bo'lsa, $M \cdot N$ ko'paytmani hisoblash kerak. Bu holda tarmoqlanish operatorlari ichma-ich joylashmoqda. Adashib ketmaslik uchun ichki tarmoqlanish operatorini begin va end xizmatchi so'zlari orasiga yozish qulay. Boshqa hollarda hech qanday amal bajarilmaydi.</p>	
<p><u>Blok sxemasi:</u></p>	<p><u>Dasturi:</u></p> <pre> Var M, N, K: Real; Begin Write('M ni kiriting: '); ReadLn(M); </pre>



```

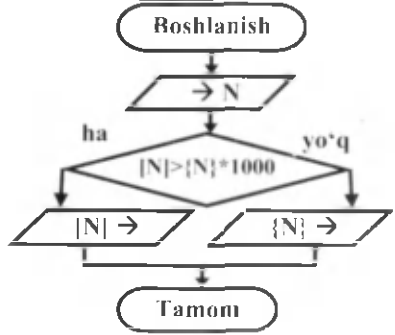
Write('N ni kiriting: '); ReadLn(N);
If (M>0) AND (N>0) then
begin
  If M+N>100 then K:=M/N
  else K:= M*N;
  WriteLn('Javob: ', K:0:3);
end
else
WriteLn('Shartlar bajarilmaydi');
ReadLn;
End.
  
```

M-7. Berilgan N sonni butun qismi kasr qismini 1000 ga ko'paytirilganidan katta bo'lsa, sonning butun qismini, aks holda kasr qismining birinchi 3 ta raqamini chiqaruvchi dastur tuzing.

Yechim:

Berilgan son N haqiqiy turdagi o'zgaruvchiga o'zlashtiriladi. Masala shartiga ko'ra N sonini butun qismini (**trunc** yordamida ajratib) kasr qismini (**frac** yordamida ajratib) 1000 ga ko'paytmasi bilan taqqoslash kerak. Sonni kasr qismini chiqarishda formatlash talab etilgan. Algoritmida formatlash aks ettirilmaydi. Matematikada $\{N\}$ – N sonini kasr qismi.

Blok sxemasi:



Dasturi:

```

Var
N: Real;
Begin
Write('N ni kiriting: '); ReadLn(N);
If trunc(N) > frac(N)*1000 then
  WriteLn('Butun: ', trunc(N))
else
  WriteLn('Kasr: ', frac(N):0:3);
ReadLn;
End.
  
```

33 – dars. Parametrlı takrorlash operatori mavzusiga

M-1. Quyidagi operatorlardagi takrorlanishlar sonini aniqlang.

Yechim:

Har bir **For I := N1 To N2 Do <takrorlanish tanasi>**; parametrlı takrorlash operatorida takrorlanishlar sonini aniqlash uchun takrorlash parametrining boshlang'ich N1 va oxirgi N2 qiymatlari orasidagi farqni hisoblab, 1 ni qo'shish kerak:

$$\text{Takrorlanishlar soni} = N2 - N1 + 1$$

a) for i:=1 to 88 do b:=1: takrorlash operatorida $88 - 1 + 1 = 88$ ta

b) for i:=73 to 161 do m:=2: takrorlash operatorida $161 - 73 + 1 = 89$ ta

d) for i:=-21 to 0 do a:=3: takrorlash operatorida $0 - (-21) + 1 = 21 + 1 = 22$ ta

e) a:=5; b:=34; for i:=a+7 to b-1 do s:=s+1; takrorlash operatorida $(b-1) - (a+7) + 1 = (34-1) - (5+7) + 1 = 33 - 12 + 1 = 19 + 1 = 20$ ta

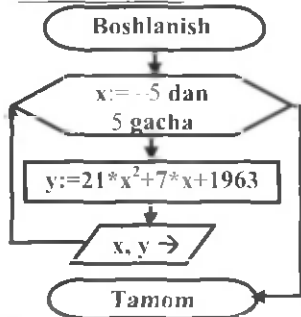
f) a:=5; b:=19; for i:=a*a to 2*b+8 do s:=s+1; takrorlash operatorida $(2*b+8) - (a*a) + 1 = (2*19+8) - (5*5) + 1 = 46 - 25 + 1 = 21 + 1 = 22$ ta

M-2. $y = 21x^2 + 7x + 1963$ funksiyaning qiymatini x ning $-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5$ qiymatlarida hisoblash dasturini tuzing.

Yechim:

Masalada butun turdagi takrorlash parametri x ning boshlang'ich qiymati $N1 = -5$ va oxirgi qiymati $N2 = 5$ ga teng, ya'ni takrorlanishlar soni $N2 - N1 + 1 = 5 - (-5) + 1 = 11$ ta. Takrorlanish tanasi $y = 21x^2 + 7x + 1963$ funksiyasini hisoblaydigan o'zlashtirish operatori, o'zgaruvchi x ning va unga mos y funksiyaning qiymatini ekranga chiqaruvchi operatoridan iborat. Demak, takrorlanish tanasida ikkita operator bo'lgani uchun **Begin** va **End** xizmatchi so'zlari orasida yoziladi.

Blok sxemasi:



Dasturi:

Var x, y: integer;

Begin

For x:=-5 to 5 do

Begin

y:=21*sqr(x)+7*x+1963;

WriteLn('x= ', x, ' da ', 'y= ', y);

End;

ReadLn;

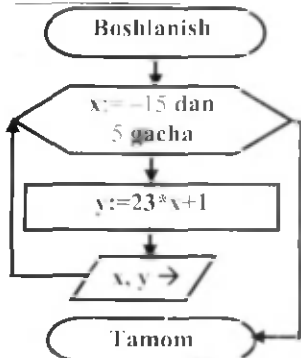
End.

M-3. $y = 23x + 1$ funksiyaning qiymatini x ning $[-15, 5]$ oraliqdagi butun qiymatlarida hisoblash dasturini tuzing.

Yechim:

Bu masalada butun turdagi takrorlash parametri x ning boshlang'ich qiymati $N1 = -15$ va oxirgi qiymati $N2 = 5$ ga teng, ya'ni takrorlanishlar soni $N2 - N1 + 1 = 5 - (-15) + 1 = 21$ ta. Takrorlanish tanasi $y = 23 \cdot x + 1$ funksiyasini hisoblaydigan o'zlashtirish operatori, o'zgaruvchi x ning va unga mos y funksiyaning qiymatini ekranga chiqaruvchi operatordan iborat. Demak, takrorlanish tanasida ikkita operator bo'lgani uchun **Begin** va **End** xizmatchi so'zlari orasida yoziladi.

Blok sxemasi:



Dasturi:

```

Var
  x, y: integer;
Begin
  For x:=-15 to 5 do
    Begin
      y:=23*x+1;
      WriteLn('x= ', x, ' da ', 'y= ', y);
    End;
  ReadLn;
End.
  
```

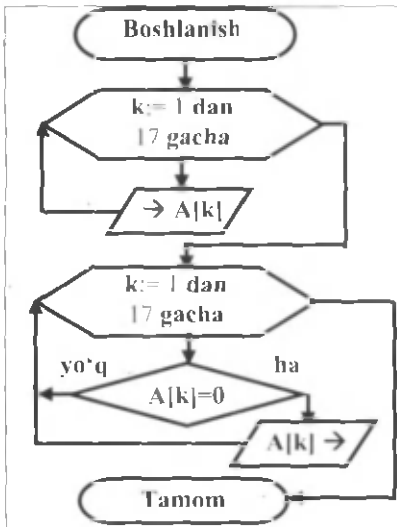
M-4. $A[1..17]$ massiv berilgan. Massivning nolga teng elementlarini indeksini chiqaruvchi dastur tuzing.

Yechim:

Bu masalada elementlari haqiqiy turda bo'lgan 17 ta elementli A massiv berilgan, demak, massiv kiritiladi. Shuning uchun takrorlash parametri k ning boshlang'ich qiymati $N1 = 1$ va oxirgi qiymati $N2 = 17$ ga teng, ya'ni takrorlanishlar soni $N2 - N1 + 1 = 17 - 1 + 1 = 17$ ta bo'lib, takrorlanish tanasi massiv elementini muloqot usulida kiritish operatoridan iborat bo'ladi. Masala yechimida parametrli takrorlash operatori ikkinchi marta qo'lanilganda takrorlanish tanasi massiv elementini 0 bilan taqqoslash uchun qisqa ko'rinishdagi tarmoqlanish operatoridan iborat bo'ladi va agar massiv elementi 0 ga teng bo'lsa ma'lumotlarni chiqarish operatori yordamida ekranga chiqariladi.

Blok sxemasi:

Dasturi:



```

Var
  k: integer;
  A: array[1..17] of real;
Begin
  For k:= 1 to 17 do
  begin
    write('A[', k, ']= ');
    readln(A[k]);
  end;
  For k:= 1 to 17 do
  If A[k]=0 then
    WriteLn('A[', k, ']= ', A[k]:0:0);
  Readln;
End.
  
```

M-5. $y = 2x + 19$ funksiyaning qiymatini x ning $[0, 10]$ oraliqda 0,25 qadam bilan hisoblash dasturini tuzing (yo'llanma: $i=0$ da $x=0$; $i=1$ da $x=0,25$; ...; $i=40$ da $x=10$).

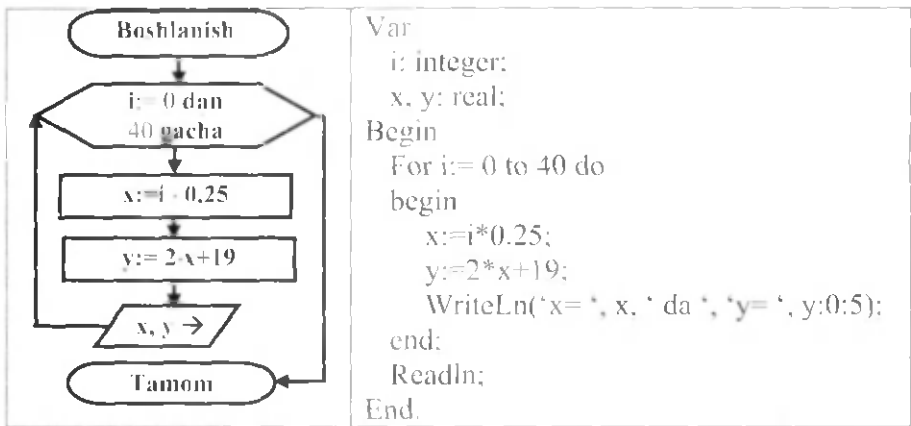
Yechim:

Parametrlil takrorlash operatorining parametri qiymati butun son bo'lishi hamda parametr qiymati 1 qadam bilan o'zgarishi kerak. Bu masalada esa x o'zgaruvchining qiymati 0,25 qadam bilan o'zgarishi sababli haqiqiy son qiymatlarni qabul qiladi. Bu muammoni hal etish uchun masalada berilgan yo'llanmadan foydalanish maqsadga muvofiq. Berilgan yo'llanmadagi qonuniyat quyidagicha:

- $i=0$ da $x=i-0.25=0-0.25=0$;
- $i=1$ da $x=i-0.25=1-0.25=0.25$;
- $i=2$ da $x=i-0.25=2-0.25=0.5$;
- $i=3$ da $x=i-0.25=3-0.25=0.75$;
- $i=4$ da $x=i-0.25=4-0.25=1$;
- ...
- $i=39$ da $x=i-0.25=39-0.25=9.75$;
- $i=40$ da $x=i-0.25=40-0.25=10$.

Demak, parametr sifatida butun turdagi i o'zgaruvchini olish mumkin. Bu holda takrorlanishlar soni $40 - 0 + 1 = 41$ ta.

Blok sxemasi:	Dasturi:
---------------	----------



34 – dars. Parametrlilik takrorlash operatori mavzusini takrorlash mavzusiga

M-1. Quyidagi operatorlardagi xatolarni aniqlang va izohlang.

a) for I=-15 to 5 do s:=s+I;

Yechim:

Parametrlilik takrorlash operatorida parametrga boshlang'ich qiymat o'zlashtiriladi. Shuning uchun "I=-15" ifoda xato yozilgan.

b) for kub:=100/10+11 to 1963 do begin a:=7;end;

Yechim:

Parametrlilik takrorlash operatorida parametr qiymati butun son bo'lishi shart. Yozilgan boshlang'ich qiymatda nisbat qatnashgani uchun Paskal dasturlash tilida haqiqiy son hisoblanadi. Shuning uchun "100/10" xato yozilgan.

d) for mag:=99 dawnto 1 do readln(aa);

Yechim:

Bu operatorida takrorlanish tanasidagi operator xato yozilgan, u "readln(aa)" kabi yozilishi lozim.

e) for bma:= 0.5 to 10 do writeln(k);

Yechim:

Parametrlilik takrorlash operatorida parametr qiymati butun son bo'lishi shart. Yozilgan boshlang'ich qiymatda haqiqiy son qatnashgan, ya'ni "0" xato yozilgan.

M-2. Quyidagi operatorlardagi takrorlanishlar sonini aniqlang.

Yechim:

Har bir **For I: = N1 To N2 Do** <takrorlanish tanasi>; parametrlilik takrorlash operatorida takrorlanishlar sonini aniqlash uchun takrorlash parametrining boshlang'ich N1 va oxirgi N2 qiymatlari orasidagi farqni hisoblab, 1 ni qo'shish kerak (downto xizmatchi so'zi ishlatilsa, quyidagi formuladagi N1 va N2 o'rnini almashadi):

$$\text{Takrorlanishlar soni} = N2 - N1 + 1$$

a) for k:=trunc(23/5) downto trunc(1/2) do m:=1991; takrorlash operatorida $\text{trunc}(23/5) - \text{trunc}(1/2) + 1 = \text{trunc}(4.6) - \text{trunc}(0.5) + 1 = 4 - 0 + 1 = 5$ ta

b) for s:=23 to 1 do m:=1963; takrorlash operatorida $1 - 23 + 1 = -21$, ya'ni biror marta ham takrorlanmaydi.

d) for J:=2 downto 19 do m:= 1950; takrorlash operatorida $2 - 19 + 1 = -16$, ya'ni biror marta ham takrorlanmaydi.

e) for d:=23 downto 1 do m:= 2009; takrorlash operatorida $23 - 1 + 1 = 23$ ta

f) for i:=abs(-25) to 25 do s:=s+i*i; takrorlash operatorida $25 - \text{abs}(-25) + 1 = 25 - 25 + 1 = 1$ ta.

g) for h:=round(9.6) downto trunc(3*3) do a:=21; takrorlash operatorida $\text{round}(9.6) - \text{trunc}(3*3) + 1 = 10 - 9 + 1 = 2$ ta.

M-3. $S=10+12+14+\dots+50$ yig'indini hisoblash dasturini tuzing.

Yechim:

Masaladagi yig'indi tahlil etilsa, qo'shiluvchilar ikkitaga ortayotganini ko'rish mumkin. Bu holda parametrlilik takrorlash operatorini qo'llab bo'lmaydi, chunki parametr har qadamda bittaga ortishi mumkin. Shuning uchun turli xil usullarni qo'llash kerak bo'ladi.

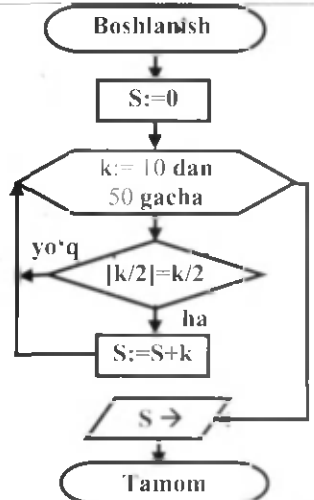
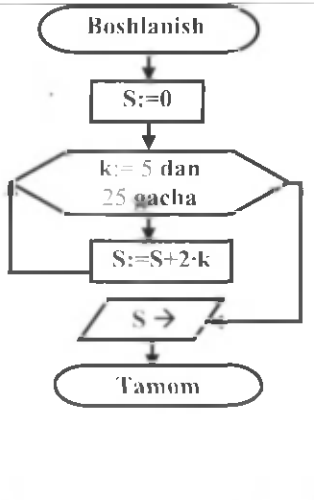
Masalan:

1-usul. Parametr qiymatini 10 dan 50 gacha bittalab o'zgartiriladi, lekin yig'indiga faqat juft qiymatlarni qo'shiladi.

2-usul. Parametr qiymatini 5 dan 25 gacha bittalab o'zgartiriladi, lekin yig'indiga parametr qiymatini ikkiga ko'paytirib qo'shiladi ($5 \cdot 2 = 10, 6 \cdot 2 = 12, \dots, 25 \cdot 2 = 50$).

Blok sxemasi (1-usul):

Blok sxemasi (2-usul):

	
<p><u>Dasturi (1-usul):</u></p> <pre> Var K, S: integer; Begin S:=0; For K:= 10 to 50 do If K mod 2 = 0 then S:=S+K; WriteLn('S= ', S); Readln; End.</pre>	<p><u>Dasturi (2-usul):</u></p> <pre> Var K, S: integer; Begin S:=0; For K:= 5 to 25 do S:=S+2*K; WriteLn('S= ', S); Readln; End.</pre>

M-4. $S = \frac{7}{11} + \frac{17}{21} + \frac{27}{31} + \dots + \frac{2007}{2011}$ yig'indini hisoblash dasturini tuzing (yo'llanma: J ni 10 ga bo'lganda qoldiq 7).

Yechim:

Masaladagi yig'indi tahlil etilsa, qo'shiluvchilar kasr ko'rinishida, ya'ni haqiqiy sonligini ko'rish mumkin. Bu holda parametrli takrorlash operatorini qo'llab bo'lmaydi, chunki parametr faqat butun son bo'lishi mumkin. Shuning uchun sun'iy usullarni qo'llash kerak bo'ladi. Masalan:

1-usul. Yo'llanmada qo'shiluvchini suratini aniqlash usuli aytilgan, chunki: $7 \bmod 10=7$, $17 \bmod 10=7$, ..., $2007 \bmod 10=7$. Mahraj esa suratdan 4 ta ortiq. Shuning uchun parametr sifatida butun qiymatli J

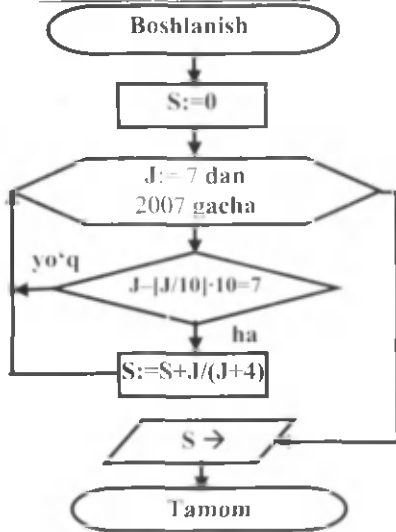
o'zgaruvchini olib, boshlang'ich qiymatini 7, oxirgi qiymatini 2007 deb olish va yig'indiga faqat 10 bo'linganda qoldig'i 7 bo'lgan sonlarni qo'shish lozim. Qoldiq hisoblash avval uchragan masalalarda ko'rilgan.

2-usul. Parametr K qiymatini 0 dan 200 gacha bittalab o'zgartiriladi, lekin yig'indidagi suratni hosil qilish uchun 10 ga ko'paytirib 7 ni qo'shiladi: $0 \cdot 10 + 7 = 7$, $1 \cdot 10 + 7 = 17$, ..., $200 \cdot 10 + 7 = 2007$. Mahraj esa suratdan 4 ta ortiq: $0 \cdot 10 + 11 = 11$, $1 \cdot 10 + 11 = 21$, ..., $200 \cdot 10 + 11 = 2011$. Bu usulda 1-usulga nisbatan kam amal bajariladi.

3-usul. Parametr M qiymatini 1 dan 201 gacha bittalab o'zgartiriladi, lekin yig'indidagi mahrajni hosil qilish uchun 10 ga ko'paytirib 1 ni qo'shiladi: $1 \cdot 10 + 1 = 11$, $2 \cdot 10 + 1 = 21$, ..., $201 \cdot 10 + 1 = 2011$. Surat esa: $1 \cdot 10 - 3 = 7$, $2 \cdot 10 - 3 = 17$, ..., $201 \cdot 10 - 3 = 2007$. Bu usulda 1-usulga nisbatan kam, 2-usul bilan deyarli bir xil amal bajariladi.

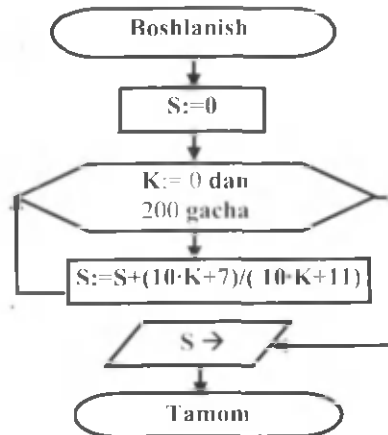
Izoh: Bu kabi masalalarni parametrli takrorlash operatorisiz ham oson hal etish mumkin (masalan, tarmoqlash va o'tish operatori yordamida, ya'ni 4-usulni qarang). Lekin hozirgi maqsad parametrli takrorlash operatorini qo'llash imkoniyatlarini ko'rib chiqishdir.

Blok sxemasi (1-usul):

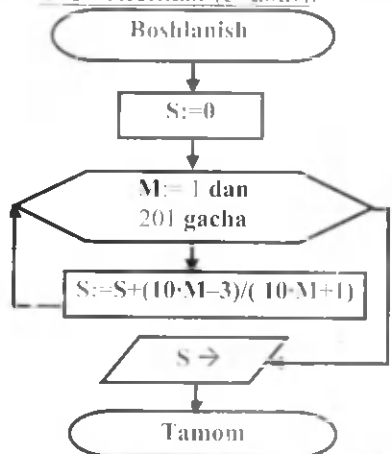
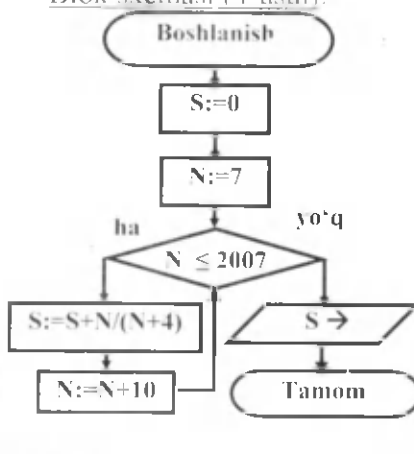


Dasturi (1-usul):

Blok sxemasi (2-usul):



Dasturi (2-usul):

<p>Var J: integer; S:real; Begin S:=0; For J:= 7 to 2007 do If J mod 10 = 7 then S:=S+J/(J+4); WriteLn('S= ', S:0:5); ReadLn; End.</p>	<p>Var K: integer; S:real; Begin S:=0; For K:= 0 to 200 do S:=S+(10*K+7)/(10*K+11); WriteLn('S= ', S:0:5); ReadLn; End.</p>
<p><u>Blok sxemasi (3-usul):</u></p>  <pre> graph TD Start([Boshlanish]) --> S0[S:=0] S0 --> Loop{M:= 1 dan 201 gacha} Loop --> Calc[S:=S+(10*M-3)/(10*M+1)] Calc --> Out[/S →/] Out --> End([Tamom]) Loop --> Calc </pre>	<p><u>Blok sxemasi (4-usul):</u></p>  <pre> graph TD Start([Boshlanish]) --> S0[S:=0] S0 --> N7[N:=7] N7 --> Dec{N ≤ 2007} Dec -- ha --> Calc[S:=S+N/(N+4)] Calc --> Nplus[N:=N+10] Nplus --> Dec Dec -- yo'q --> Out[/S →/] Out --> End([Tamom]) </pre>
<p><u>Dasturi (3-usul):</u></p> <p>Var M: integer; S:real; Begin S:=0; For M:= 1 to 201 do S:=S+(10*M-3)/(10*M+1); WriteLn('S= ', S:0:5); ReadLn; End.</p>	<p><u>Dasturi (4-usul):</u></p> <p>Label 1: Var N: integer; S:real; Begin S:=0; N:=7; 1: S:=S+N/(N+4); N:=N+10; If N<=2007 then goto 1; WriteLn('S= ', S:0:5); ReadLn; End.</p>

M-5. $P=1 \cdot 3 \cdot 5 \cdot \dots \cdot 23$ ko'paytmani hisoblash dasturini tuzing.

Yechim:

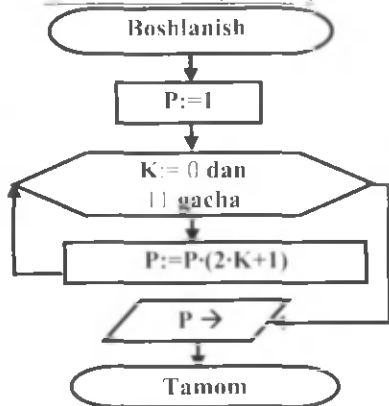
Masaladagi ko'paytma tahlil etilsa, ko'paytuvchilar ikkitaga ortmoqda. Bu holda ham parametrli takrorlash operatorini qo'llab bo'lmaydi, chunki parametr faqat bittaga ortishi mumkin. Shuning uchun sun'iy usullarni qo'llash kerak bo'ladi. Masalan:

1-usul. Parametr sifatidagi K ni 0 dan 11 gacha bittalab orttiriladi. Shunda: $2 \cdot 0 + 1 = 1$, $2 \cdot 1 + 1 = 3$, $2 \cdot 2 + 1 = 5$, ..., $2 \cdot 11 + 1 = 23$.

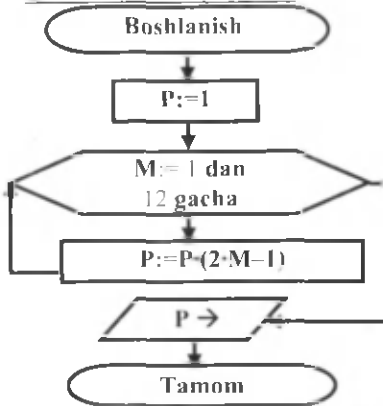
2-usul. Parametr sifatidagi M ni 1 dan 12 gacha bittalab orttiriladi. Shunda: $2 \cdot 1 - 1 = 1$, $2 \cdot 2 - 1 = 3$, $2 \cdot 3 - 1 = 5$, ..., $2 \cdot 12 - 1 = 23$.

Izoh: Masala yechimidagi parametrlar qiymati 0 dan kichik bo'lmagani va 127 dan katta bo'lmagani uchun parametr o'zgaruvchisi turini **byte**, ko'paytma katta son bo'lib ketishini hisobga olib natijaviy ko'paytma o'zgaruvchisi P ni word turida olish maqsadga muvofiq.

Blok sxemasi (1-usul):



Blok sxemasi (2-usul):



Dasturi (1-usul):

```

Var
  K: byte; P: word;
Begin
  P:=1;
  For K:= 0 to 11 do
    P:=P*(2*K+1);
  WriteLn('P= ', P);
  ReadLn;
End.
    
```

Dasturi (2-usul):

```

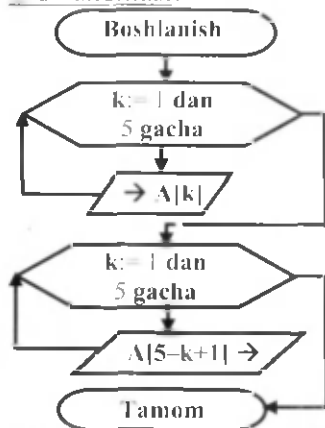
Var
  M: byte; P: word;
Begin
  P:=1;
  For M:= 1 to 12 do
    P:=P*(2*M-1);
  WriteLn('P= ', P);
  ReadLn;
End.
    
```

M-6. $A[1..5]$ massiv berilgan. Massiv elementlarini teskari tartibda chiqaruvchi dastur tuzing.

Yechim:

Bu masalada elementlari haqiqiy turda bo'lgan 5 ta elementli A massiv berilgan, demak, massiv kiritiladi. Shuning uchun takrorlash parametri k ning qiymati 1 dan to 5 ga o'zgaradi, takrorlanish tanasi massiv elementini muloqot usulida kiritish operatoridan iborat bo'ladi. Masala yechimida parametrli takrorlash operatori ikkinchi marta qo'llanilganda takrorlanish tanasi massiv elementlarini teskari tartibda chiqarishdan iborat bo'ladi. Algoritm tuzish uchun k parametr 1 dan 5 gacha bittalab ortganda $6-k$ soni 5 dan 1 gacha bittalab kamayishini hisobga olish kifoya. Paskal dasturlash tilida bu formulasiz ham masalani hal etish mumkin. Buning uchun **For k:=5 downto 1 do** parametrli takrorlash operatorini yozish mumkin.

Blok sxemasi:



Dasturi (1-usul):

```

Var
  k: integer;
  A: array[1..5] of real;
Begin
  For k:= 1 to 5 do
    begin write('A[', k, ']= ');
           readln(A[k]);
    end;
  For k:= 1 to 5 do
    WriteLn(A[6-k]:0:3);
  Readln;
End.
  
```

Dasturi (2-usul):

```

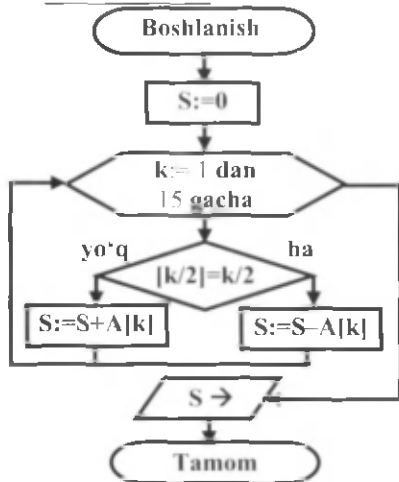
Var k: integer; A: array[1..5] of real;
Begin
  For k:= 1 to 5 do begin write('A[', k, ']= '); readln(A[k]); end;
  For k:= 5 downto 1 do WriteLn(A[k]:0:3); Readln;
End.
  
```

M-7. $A[1..15]$ massiv berilgan. Massivni toq indeksli elementlarini yig'indisidan juft indeksli elementlarini yig'indisini ayirib chiqaruvchi dastur tuzing.

Yechim:

Bu masalada elementlari haqiqiy turda bo'lgan 15 ta elementli A massiv berilgan, demak, massiv kiritiladi. Shuning uchun takrorlash parametri k ning qiymati 1 dan to 15 ga o'zgaradi, takrorlanish tanasi massiv elementini mulohazasizlik usulida kiritish operatoridan iborat bo'ladi. Masala yechimida parametrli takrorlash operatori ikkinchi marta qo'llanilganda takrorlanish tanasi massiv elementlari indeksini toq yoki juftligini tekshiruvchi to'liq ko'rinishdagi tarmoqlanish operatoridan iborat bo'ladi. Tarmoqlanish natijasiga ko'ra $S:=S+A[k]$ yig'indi yoki $S:=S-A[k]$ ayirma hisoblanadi.

Blok sxemasi:



Dasturi:

Var

k: integer; S: real;

A: array[1..15] of real;

Begin

For k:= 1 to 15 do begin

write('A[', k, ']= ');

readln(A[k]);

end;

For k:= 1 to 15 do

If k mod 2 = 0 Then S:=S-A[k]

Else S:=S+A[k];

WriteLn('S= ', S:0:3);

Readln;

End.

35 – dars. Shart bo'yicha takrorlash operatorlari mavzusiga

M-1. Quyidagi operatorlardagi xatoliklarni aniqlang va izohlang.

a) while 5*6 do SH:=sqr(2);

Javob:

While va **do** xizmatchi so'zlari orasida tekshirilishi zarur bo'lgan shart yozilishi kerak, 5*6 esa shart bo'la olmaydi.

b) WHILE 5>6 do Od:=Od+1;

Javob:

Bu operator yozilishida xato yo'q. Yozilgan 5>6 sharti "Yolg'on" qiymat qabul qiladi, shuning uchun takrorlanish soni 0 ta.

<p>d) Repeat $i < j$ Until $s := 0$; <u>Javob:</u> Repeat va Until xizmatchi soʻzlari orasida bajarilishi lozim boʻlgan operatorlar yozilishi kerak, lekin $i < j$ operator emas, Until xizmatchi soʻzidan keyin esa tekshirilishi zarur boʻlgan shart yozilishi kerak, $s := 0$ esa shart emas, u oʻzlashtirish operatoridir.</p>	<p>e) rEpEaT $s := 0$ Until $s := 0$; <u>Javob:</u> Until xizmatchi soʻzidan keyin esa tekshirilishi zarur boʻlgan shart yozilishi kerak, $s := 0$ esa oʻzlashtirish operatoridir, shart emas.</p>
---	--

M-2. Quyidagi operatorlardagi takrorlanishlar sonini aniqlang.

<p>a) $x := -5$; while $X > 0$ do $x := x + 2$; Oʻzgaruvchi x ning qiymati -5, yaʼni manfiy. Shuning uchun takrorlash operatoridagi $X > 0$ shart bajarilmaydi. Demak, takrorlanishlar soni 0 ta.</p>
<p>b) $x := -5$; while $X < 10$ do begin $x := x + 2$; $x := 2 * x$; end; Oʻzgaruvchi x ning qiymati -5, shuning uchun shart boʻyicha takrorlash operatoridagi $X < 10$ shart bajariladi. U holda do xizmatchi soʻzidan keyin yozilgan begin va end xizmatchi soʻzlari orasidagi operatorlar bajariladi. Natijada birinchi takrorlanishda quyidagi qiymatlar hosil boʻladi: $x := x + 2 = -5 + 2 = -3; x := 2 * x = 2 * (-3) = -6;$ Oʻzgaruvchi x ning qiymati -6, shuning uchun takrorlash operatoridagi $X < 10$ shart bajariladi. Ikkinchi takrorlanishda quyidagi qiymatlar hosil boʻladi: $x := x + 2 = -6 + 2 = -4; x := 2 * x = 2 * (-4) = -8;$ Bundan koʻrinadiki, oʻzgaruvchi x ning qiymati har takrorlanishda kamayib bormoqda. Demak, har bir takrorlanishdan keyin $X < 10$ shart bajarilaveradi. Bu esa takrorlanish cheksiz marta roʻy berishi mumkinligini bildiradi. Lekin, Paskal dasturlash tilida har bir oʻzgaruvchi turida qiymat boʻyicha quyi va yuqori chegara mavjud. Shunga koʻra oʻzgaruvchi x ning qiymati biror qadamda chegaradan chiqib ketadi. Masalan, agar x integer turdagi oʻzgaruvchi boʻlsa quyi chegara -32768 ga teng. Dastur yordamida tekshirib koʻrish mumkinki, 14 takrorlanishda x ning qiymati -16388 ga, 15 takrorlanishda esa x ning qiymati 32764 ga teng boʻladi, bu esa $X < 10$ sharti bajarilmasligini bildiradi.</p>

Hisoblash mumkinki, 15 takrorlanishda: $x:=x+2=-16388+2=-16386$; $x:=2*x=2*(-16386)=-32772$. Endi chegaradan ortib ketgan qism x o'zgaruvchining yuqori chegarasi tomondan hisoblanadi, ya'ni yuqori chegaradan quyi chegaradan keyingi son bilan hosil bo'lgan son ayirmasi ayiriladi: $32767-(-32769-(-32772))=32767-(-32769+32772)=32767-3=32764$.

d) $i:=0$; while $i*i \leq 1.2$ do $i:=i+0.1$;

O'zgaruvchi i ning qiymati 0, shuning uchun shart bo'yicha takrorlash operatoridagi $(i*i=0*0=)0 \leq 1.2$ shart bajariladi. U holda **do** xizmatchi so'zidan keyin yozilgan $i:=i+0.1$; o'zlashtirish operatori bajariladi.

Birinchi takrorlanishda: $i:=i+0.1=0+0.1=0.1=1*0.1$. Shart yana bajarilgani uchun $(0.1*0.1=0.01 \leq 1.2)$ takrorlanish davom etadi.

Ikkinchi takrorlanishda: $i:=i+0.1=0.1+0.1=0.2=2*0.1$. O'zgaruvchi i ning qiymati har shart bajarilganda 0.1 ga ortaveradi va 11-takrorlanishqa $11*0.1=1.1$ ga teng bo'ladi. O'zgaruvchi i ning bu qiymatida $i*i=1.1*1.1=1.21$ bo'lgani uchun $i*i \leq 1.2$ shart bajarilmaydi va takrorlanish to'xtaydi.

e) $k:=5$; while $k/5 \leq 2.5$ do $k:=k+1.5$;

O'zgaruvchi k ning qiymati 5, shuning uchun shart bo'yicha takrorlash operatoridagi $(k/5=5/5=)1 \leq 2.5$ shart bajariladi. U holda **do** xizmatchi so'zidan keyin yozilgan $k:=k+1.5$; o'zlashtirish operatori bajariladi.

Birinchi takrorlanishda: $k:=k+1.5=5+1.5=6.5=5+1*1.5$. Shart bajarilgani uchun $(6.5/5=1.3 \leq 2.5)$ takrorlanish davom etadi.

Ikkinchi takrorlanishda: $k:=k+1.5=6.5+1.5=8=5+2*1.5$. O'zgaruvchi k ning qiymati har shart bajarilganda 1.5 ga ortaveradi va 6-takrorlanishda $5+6*1.5=5+9=14$ ga teng bo'ladi. O'zgaruvchi k ning bu qiymatida $k/5=14/5=2.8$ bo'lgani uchun $k/5 \leq 2.5$ shart bajarilmaydi va takrorlanish to'xtaydi.

f) $t:=100$; repeat $t:=t/10$; until $t \leq 0.1$;

Repeat operatori shart tekshirishdan avval bir marta takrorlanish tanasidagi operatorlarni bajarish imkonini beradi va keyin shartni tekshiradi. Bu birinchi takrorlanishdir.

O'zgaruvchi t ning qiymati 100, shuning uchun shart bo'yicha takrorlash operatoridagi $(t:=t/10=100/10=)10 \leq 0.1$ shart bajarilmaydi, ya'ni "Yolg'on" qiymat qabul qiladi. Ma'lumki, **Repeat** operatori

shart qiymati “Yolg‘on” bo‘lganda ishlaydi.

Ikkinchi takrorlanishda: $t:=t/10=10/10=1$. Shart “Yolg‘on” qiymat qabul qilgani uchun ($t=1 \leq 0.1$) takrorlanish davom etadi.

Uchinchi takrorlanishda: $t:=t/10=1/10=0.1$. O‘zgaruvchi t ning bu qiymatida shart “Rost” qiymat qabul qiladi: ($t=0.1 \leq 0.1$). Shuning uchun takrorlanish to‘xtaydi.

g) $x:=0$; repeat $x:=x+1/10$; until $\text{sqr}(x) \geq 6/5$;

Repeat operatori shart tekshirishdan avval bir marta takrorlanish tanasidagi operatorlarni bajarish imkonini beradi va keyin shartni tekshiradi. Bu birinchi takrorlanishdir.

O‘zgaruvchi x ning qiymati 0 va $x:=x+1/10=0+0.1=0.1=1*0.1$, shuning uchun shart bo‘yicha takrorlash operatoridagi ($0.01=\text{sqr}(x) \geq 6/5(=1.2)$) shart bajarilmaydi, ya‘ni “Yolg‘on” qiymat qabul qiladi. Aytib o‘tilganidek, **Repeat** operatori shart qiymati “Yolg‘on” bo‘lganda ishlaydi.

Ikkinchi takrorlanishda: $x:=x+1/10=0.1+0.1=0.2=2*0.1$ va ($0.04=\text{sqr}(x) \geq 6/5(=1.2)$). Shart “Yolg‘on” qiymat qabul qilgani uchun takrorlanish davom etadi.

Uchinchi takrorlanishda: $x:=x+1/10=0.2+0.1=0.3=3*0.1$ va ($0.09=\text{sqr}(x) \geq 6/5(=1.2)$). Shart “Yolg‘on” qiymat qabul qilgani uchun takrorlanish davom etadi. O‘zgaruvchi x ning qiymati har shart “Yolg‘on” qiymat qabul qilganda 0.1 ga ortaveradi va 11-takrorlanishda $11*0.1=1.1$ ga teng bo‘ladi. O‘zgaruvchi x ning bu qiymatida shart “Rost” qiymat qabul qiladi: ($1.21=\text{sqr}(x) \geq 6/5(=1.2)$). Shuning uchun takrorlanish to‘xtaydi.

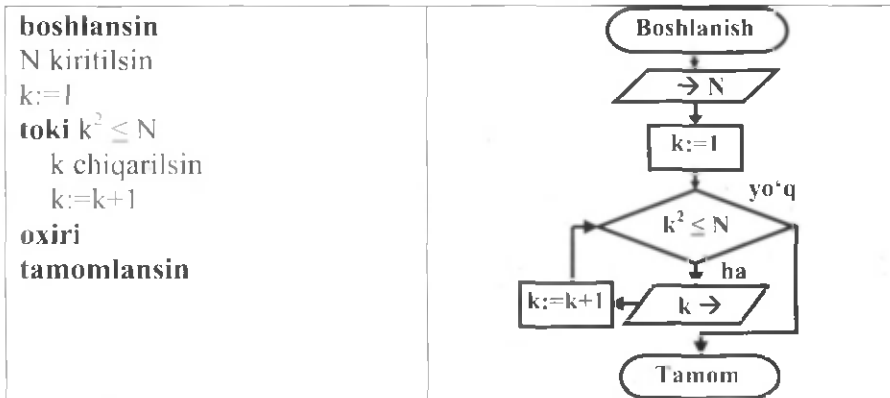
M-3. N natural son berilgan. Kvadrati N dan katta bo‘lmagan barcha natural sonlarni chiqaruvchi dastur tuzing.

Yechim:

Bu masalada takrorlanishni tashkil etish uchun parametrli takrorlash operatoridan foydalanish mumkin, lekin avval takrorlash parametri uchun oxirgi qiymatni aniqlab olish kerak bo‘ladi. Masala shartini o‘zi k parametr uchun $k^2 \leq N$ shart o‘rinli bo‘lganda takrorlanish bajarilishi kerakligini ko‘rsatmoqda. Shart tekshirish asosida takrorlanish tashkil etilayotgani uchun **While–do** shart bo‘yicha takrorlash operatoridan foydalanish qulay.

So‘z orqali:

Blok sxemasi:



Dasturi:

Var

k, N: word; {chunki, k va N natural son}

Begin

write('N= '); readln(N);

k:= 1;

While $\text{sqr}(k) \leq N$ do begin WriteLn(k); k:=k+1; end;

Readln;

End.

M-4. $y = x * \sin x$ funksiyaning qiymatlarini $[-\pi, \pi]$ oraliqda 0.3 qadam bilan hisoblash dasturini tuzing.

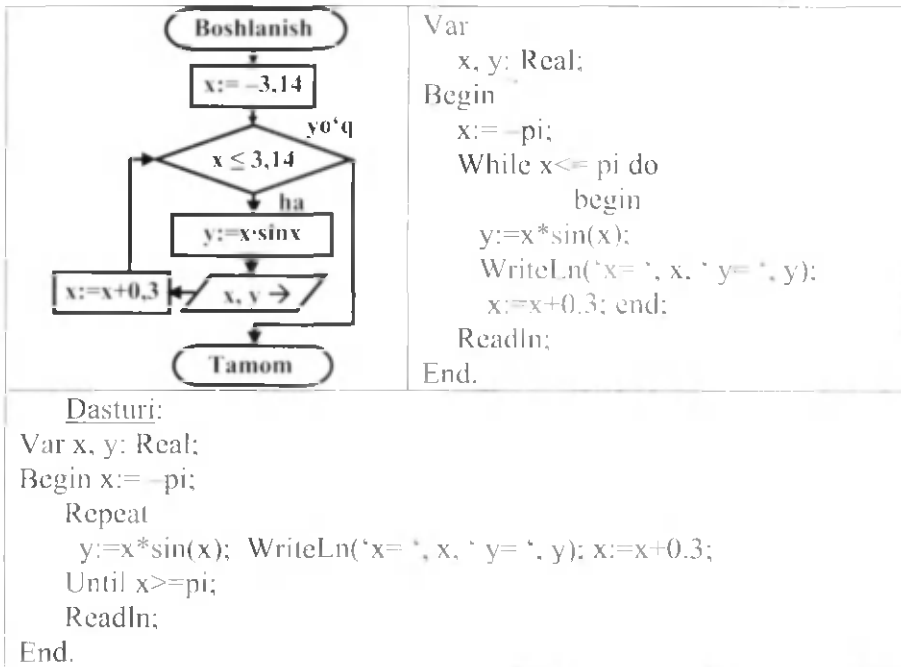
Yechim:

Bu masalada takrorlanishni tashkil etish uchun parametrlil takrorlash operatoridan foydalanish murakkab, chunki takrorlanish parametrining boshlang'ich va oxirgi qiymatlari haqiqiy son, parametr o'zgarish qadami ham 1 ga teng emas. Lekin masala **While-do** shart bo'yicha takrorlash operatori yordamida osongina hal etiladi, chunki unda o'zgarish qadami ham chegaralar ham haqiqiy son bo'lishi mumkin.

Masalani hal etish uchun **Repeat-Until** shart bo'yicha takrorlash operatoridan ham foydalanish mumkin. Bu usulda takrorlanish bir marta ortiq bajarilishi mumkinligini va **Repeat** shart "yolg'on" qiymat qabul qilganda takrorlanishni davom ettirishini e'tiborga olish muhim.

Blok sxemasi (While ga):

Dasturi:



M-5*. N natural son va A_1, A_2, \dots, A_N butun sonlar ketma-ketligi berilgan. Ularni ketma-ket qoʻshib borib, yigʻindi berilgan N butun sondan ortishi bilan ekranga chiqaruvchi dastur tuzing. Agar barcha sonlar yigʻindisi N dan oshmasa, bu haqida habar chiqarilsin.

Yechim:

Bu masalada takrorlanishni tashkil etish uchun parametrlilik takrorlash operatoridan foydalanish mumkin, buning uchun avval takrorlanish parametri uchun oxirgi qiymatni aniqlab olish kerak boʻladi. Lekin bu masalada takrorlanish parametri uchun oxirgi qiymatni aniqlab olishning ilojisi yoʻq.

Masala shartiga koʻra A_1, A_2, \dots, A_N butun sonlar ketma-ketligi hadlarini ($A[1..N]$ massiv elementlarini) S butun turdagi oʻzgaruvchiga yigʻib boriladi. Har bir element qoʻshilganda $S \leq N$ shart tekshiriladi. Shart tekshirish asosida takrorlanish tashkil etilayotgani uchun **While-do** shart boʻyicha takrorlash operatoridan foydalanish qulay. Masalani hal etish uchun **Repeat-Until** shart boʻyicha takrorlash operatoridan ham foydalanish mumkin. Bu usulda takrorlanish bir marta ortiq bajarilishi mumkinligini va **Repeat** shart “yolgʻon” qiymat qabul qilganda

takrorlanishni davom ettirishini e'tiborga olish muhim.

Dasturda shuni nazarda tutish kerakki, massiv elementlari tugagan-da ham $S \leq N$ shart bajarilmasligi mumkin. Shuning uchun massiv elementlari soni tugashi $k \leq N$ shartini ham tekshirib turish zarur. Buning uchun **While-do** operatorida **AND**, **Repeat-Until** operatorida **OR** mantiqiy amalidan foydalanish mumkin.

N natural son bo'lgani uchun **Word** turida, A_1, A_2, \dots, A_N butun sonlar bo'lgani uchun **Integer** turida tavsiflash mumkin. Tavsiflashda massiv elementlari soni aniq ko'rsatilishi kerak, shuning uchun ularni sonini 1000 ta berish mumkin.

Dasturi:

Var

N, k : Word; S : Integer; A : array[1..1000] of Integer;

Begin

write('N= '); readln(N);

For $k:=1$ to N do begin write('A[',k,']= '); readln(A[k]); end;

$k:=1$; $S:=A[1]$;

While ($S \leq N$) AND ($k \leq N$) do begin $k:=k+1$; $S:=S+A[k]$; end;

IF $S > N$ Then Write('S= ', S)

Else Write('Yig'indi ', N, ' dan oshmaydi');

Readln;

End.

Dasturi:

Var

N, k : Word; S : Integer; A : array[1..1000] of Integer;

Begin

write('N= '); readln(N);

For $k:=1$ to N do begin write('A[',k,']= '); readln(A[k]); end;

$k:=0$; $S:=0$;

Repeat $k:=k+1$; $S:=S+A[k]$;

Until ($S > N$) OR ($k >= N$);

IF $S > N$ Then Write('S= ', S)

Else Write('Yig'indi ', N, ' dan oshmaydi');

Readln;

End.

36 – dars. Shart bo'yicha takrorlash operatorlari mavzusini takrorlash mavzusiga

T-1. $S=0.5+1.5+2.5+\dots+98.5+99.5$ yig'indini hisoblash dasturini tuzing.

<p><u>Yechim:</u> Yig'indida barcha sonlar haqiqiy. Shuning uchun masalani hal etishda While-do yoki Repeat-Until shart bo'yicha takrorlash operatorlaridan foydalanish qulay. Lekin parametrlri takrorlash operatoridan foydalanishning ham imkoni bor.</p>	<p><u>Dasturi (While):</u> Var S, k: Real; Begin k:= 0.5; S:=0; While k<= 99.5 do begin S:=S+k; k:=k+1; end; Writeln('S= ', S:0:5); Readln; End.</p>
<p><u>Dasturi (Repeat):</u> Var S, k: Real; Begin k:= 0.5; S:=0; Repeat S:=S+k; k:=k+1; Until k>99.5; Writeln('S= ', S:0:5); Readln; End.</p>	<p><u>Dasturi (For):</u> Var S: Real; k:byte; Begin S:= 0; For k:=0 to 99 do S:=S+k*0.5; Writeln('S= ', S:0:5); Readln; End.</p>

T-2. $S=1\cdot2+3\cdot4+5\cdot6+\dots+101\cdot102$ yig'indini hisoblash dasturini tuzing.

<p><u>Yechim:</u> Yig'indidagi har bir qo'shiluvchi toq son va juft son ko'paytmasidan iborat bo'lib, o'zgarish qadami 2 ga teng. Shuning uchun masalani hal etishda While-do yoki Repeat-Until shart bo'yicha takrorlash operatorlaridan foydalanish qulay. Lekin parametrlri takrorlash operatoridan foydalanishning ham imkoni bor. Ko'paytmalar yig'indisi yoki juda ko'p ko'paytuvchilar qatnashgan ko'paytmalarda yig'indi yoki ko'paytma uchun belgilangan o'zgaruvchi turini tanlashda ehtiyot bo'lish kerak. Chunki, natija qiymati o'zgaruvchi turiga mos aniqlangan chegaradan chiqib ketsa xato natijaga ega bo'lamiz. Masalan, ushbu masalada yig'indi S ni Word turidagi butun son deb olinsa, qiymatlar chegarasi 0..65535 bo'lgani uchun xato natijaga olib keladi.</p>

Xatolikni ko'rish uchun tarqorlanish tanasiga Write('k= ', k, S= ' ', S, ' ! '); operatorini kiritib, natijalarni tahlil etish oson. Natijalarning ekranda aks etishi quyida keltirilgan. Xato natijaga olib kelgan tavsiflashga mos qiymatlarni tahlil etib, k=73 bo'lganda S=63492, k=75 bo'lganda S=3358 ekanligini ko'rish mumkin. Bunday bo'lishi quyidagicha izohlanadi:

Dastur ishlaganda S o'zgaruvchining qiymati ortib boradi. Yig'indiga k=75 bo'lgan had qo'shilganda 68894 soni hosil bo'lib, S o'zgaruvchining yuqori chegarasi 65535 dan ortib ketadi. Shuning natijasida chegaradan ortib ketgan qism S o'zgaruvchining quyi chegarasi tomondan hisoblanadi, ya'ni quyi chegara 0 dan yuqori chegaradan keyingi son bilan hosil bo'lgan son ayirmasi ayriladi: $0 - (65536 - 68894) = 0 - (-3358) = 3358$.

Xato dastur (While):

```
Var S: Word; k:byte;
Begin clrscr;
k:= 1; S:=0;
While k<= 101 do begin
Write('k= ', k, ' S= ', S, ' ! ');
S:=S+k*(k+1); k:=k+2; end;
Readln;
End.
```

To'g'ri dastur (While):

```
Var S: LongInt; k:byte;
Begin clrscr;
k:= 1; S:=0;
While k<= 101 do begin
Write('k= ', k, ' S= ', S, ' ! ');
S:=S+k*(k+1); k:=k+2; end;
Readln;
End.
```

To'g'ri dastur (Repeat):

```
Var S: LongInt; k:byte;
Begin clrscr;
k:= 1; S:=0;
Repeat S:=S+k*(k+1); k:=k+2;
Until k>=101;
Writeln('S= ', S);
Readln;
End.
```

To'g'ri natijaning ekrandagi ko'rinishi:

```

k= 1 S= 0 ! k= 3 S= 2 ! k= 5 S= 14 ! k= 7 S= 44 ! k= 9 S= 100 ! k= 11 S= 190 !
k= 13 S= 322 ! k= 15 S= 504 ! k= 17 S= 744 ! k= 19 S= 1050 ! k= 21 S= 1430 !
k= 23 S= 1892 ! k= 25 S= 2444 ! k= 27 S= 3094 ! k= 29 S= 3850 ! k= 31 S= 4720 !
k= 33 S= 5712 ! k= 35 S= 6834 ! k= 37 S= 8094 ! k= 39 S= 9500 ! k= 41 S= 11060 !
k= 43 S= 12782 ! k= 45 S= 14674 ! k= 47 S= 16744 ! k= 49 S= 19000 !
k= 51 S= 21450 ! k= 53 S= 24102 ! k= 55 S= 26964 ! k= 57 S= 30044 !
k= 59 S= 33350 ! k= 61 S= 36890 ! k= 63 S= 40672 ! k= 65 S= 44704 ! k= 67 S= 48994 !
k= 69 S= 53550 ! k= 71 S= 58380 ! k= 73 S= 63492 ! k= 75 S= 68894 !
k= 77 S= 74594 ! k= 79 S= 80600 ! k= 81 S= 86920 ! k= 83 S= 93562 ! k= 85 S= 100534 !
k= 87 S= 107844 ! k= 89 S= 115500 ! k= 91 S= 123510 ! k= 93 S= 131882 !
k= 95 S= 140624 ! k= 97 S= 149744 ! k= 99 S= 159250 ! k= 101 S= 169150 !

```

Xato natijaning ekrandagi ko'rinishi:

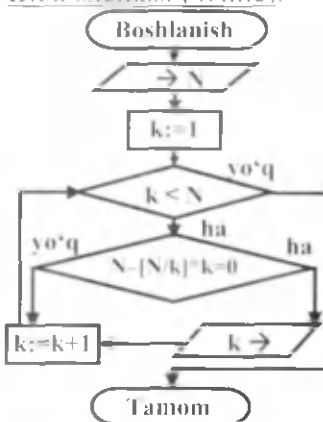
```
Turbo Pascal
k= 1 S= 0 ! k= 3 S= 2 ! k= 5 S= 14 ! k= 7 S= 44 ! k= 9 S= 100 ! k= 11 S= 190 !
k= 13 S= 322 ! k= 15 S= 504 ! k= 17 S= 744 ! k= 19 S= 1050 ! k= 21 S= 1430 !
k= 23 S= 1892 ! k= 25 S= 2444 ! k= 27 S= 3094 ! k= 29 S= 3850 ! k= 31 S= 4720 !
k= 33 S= 5712 ! k= 35 S= 6834 ! k= 37 S= 8094 ! k= 39 S= 9500 ! k= 41 S= 11060 !
k= 43 S= 12782 ! k= 45 S= 14674 ! k= 47 S= 16744 ! k= 49 S= 19000 !
k= 51 S= 21450 ! k= 53 S= 24102 ! k= 55 S= 26964 ! k= 57 S= 30044 !
k= 59 S= 33350 ! k= 61 S= 36090 ! k= 63 S= 40672 ! k= 65 S= 44704 ! k= 67 S= 48994 !
k= 69 S= 53550 ! k= 71 S= 58300 ! k= 73 S= 63492 ! k= 75 S= 69358 !
k= 77 S= 9058 ! k= 79 S= 15064 ! k= 81 S= 21304 ! k= 83 S= 28026 ! k= 85 S= 34998 !
k= 87 S= 42308 ! k= 89 S= 49964 ! k= 91 S= 57974 ! k= 93 S= 67100 !
k= 95 S= 9552 ! k= 97 S= 18672 ! k= 99 S= 28178 ! k= 101 S= 38078 !
```

T-3. N natural sonining barcha bo'luvchilarini chiqaruvchi dastur tuzing.

Yechim:

Bu masalada takrorlanishni tashkil etish uchun parametrli takrorlash operatori, shart bo'yicha takrorlash operatorlaridan foydalanish mumkin. Masala shartiga ko'ra N sonini 1, 2, 3, ..., N sonlariga bo'lganda qolgan qoldiqni 0 bilan taqqoslash zarur bo'ladi.

Blok sxemasi (While):



Dasturi:

```

Var
  k, N: word;
Begin
  write('N='): readln(N);
  k:= 1;
  While k<=N do
  begin
  IF N mod k=0 then Write(k, ' ');
  k:=k+1;
  end; Readln;
End.
    
```

Dasturi:

```

Var
  k, N: word;
Begin
  write('N='): readln(N);
  k:= 1;
  Repeat
  IF N mod k=0 then Write(k, ' ');
  k:=k+1;
  Until k>N;
Readln;
End.
    
```


T-4. Berilgan N natural sonni nechta raqamdan iborat ekanligini aniqlovchi dastur tuzing (yo'llanma: nechta marta $N=N \text{ div } 10$ bajarilsa $N=0$ bo'ladi?).

Yechim:

Yo'llanmani aniq songa tatbiqi:

$N=349$; $S=0$;

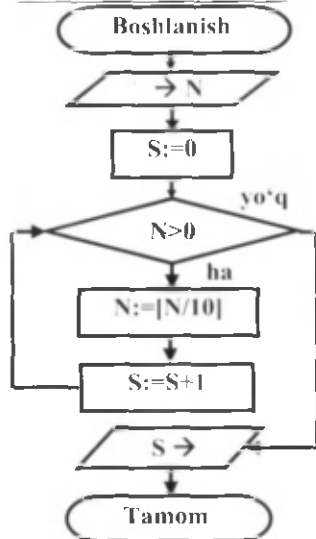
1) $N=N \text{ div } 10=349 \text{ div } 10=34$; $S=1$;

2) $N=N \text{ div } 10=34 \text{ div } 10=3$; $S=2$;

3) $N=N \text{ div } 10=3 \text{ div } 10=0$; $S=3$.

Demak, kiritilgan sonni raqamlari soni shu son 0 ga aylanguncha 10 ga nechta marta bo'linishiga bog'liq ekan. Masala shartida N natural son deyilganligi While operatorini qo'llashni yengillashtiradi. Agar N butun son deyilganda While qatnashgan dasturga tarmoqlanish operatorini yozish kerak bo'lar edi, chunki $N=0$ da While bir marta ham ishlamas edi va $S=1$ o'rniga $S=0$ javob olinar edi.

Blok sxemasi (While):



Dasturi:

Var S: byte; N: word;

Begin

write('N= '); readln(N);

S:= 0;

While N>0 do

begin N:=N div 10; S:=S+1; end;

Write('S= ', S);

Readln;

End.

Dasturi:

Var S: byte; N: word;

Begin

write('N= '); readln(N);

S:= 0;

Repeat

N:=N div 10; S:=S+1;

Until N=0;

Write('S= ', S); Readln;

End.

T-5. N natural son berilgan. 1 dan N gacha bo'lgan natural sonlar ichida oxirgi raqami 3 ga karrali sonlarni chiqaruvchi dastur tuzing.

Yechim:

Biror K sonini oxirgi raqamini ajratib olish uchun quyidagi formuladan foydalanish mumkin: **raqam**= $K \text{ mod } 10$. Masalan, ra-

Dasturi:

Var

bor: boolean;

K , N : word; raqam:byte;

Begin

<p>qam=27046 mod 10=6. Raqamni 3 ga bo'linish sharti raqam mod 3=0 qisqa ko'rinishdagi tarmoqlanish operatorida tekshiriladi. Endi 1 dan N gacha bo'lgan sonlarni bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mumkin. Agar kiritilgan songacha bo'lgan sonlar ishida birorta ham oxirgi raqami 3 karra-li son bo'lmasa, bor mantiqiy o'zgaruvchi orqali javob berish mumkin.</p>	<pre>write('N= '); readln(N); Write('Oxirgi raqami 3 ga kar- rali sonlar: '); K:= 1; bor:=true; While K<=N do BEGIN raqam:=K mod 10; IF raqam mod 3 =0 then begin Write(' ', K); bor:=false; end; K:=K+1; END; IF bor then Write('yo`q. '); Readln; End.</pre>
<p><u>Dasturi:</u> Var bor: boolean; K, N: word; raqam:byte; Begin write('N= '); readln(N); Write('Oxirgi raqami 3 ga kar- rali sonlar: '); K:= 1; bor:=true; REPEAT raqam:=K mod 10; IF raqam mod 3 =0 then begin Write(' ', K); bor:=false; end; K:=K+1; UNTIL K>N; IF bor then Write('yo`q. '); Readln; End.</p>	<p><u>Dasturi:</u> Var bor: boolean; K, N: word; raqam:byte; Begin write('N= '); readln(N); Write('Oxirgi raqami 3 ga kar- rali sonlar: '); bor:=true; For K:=1 to N do BEGIN raqam:=K mod 10; IF raqam mod 3 =0 then begin Write(' ', K); bor:=false; end; END; IF bor then Write('yo`q. '); Readln; End.</p>

T-6. Ikki xonali natural sonlar ichidan raqamlari yig'indisi juft bo'lgan sonlarni chiqaruvchi dastur tuzing (yo'llanma: K sonning birlik raqami $K1=K \bmod 10$, o'nlik raqami $K10=K \div 10$).

<p><u>Yechim:</u> Yo'llanmaga ko'ra ikki xonali K</p>	<p><u>Dasturi:</u> Var</p>
--	---------------------------------

<p>natural sonning birlik raqami $K1=K \bmod 10$, o'nlik raqami $K10=K \div 10$. Bu sonni raqamlari yig'indisi juftligi $(K1+K10) \bmod 2=0$ sharti orqali qisqa ko'rinishdagi tarmoqlanish operatorida tekshiriladi. Masalan, $K=63$ bo'lsa, $K1=K \bmod 10=63 \bmod 10=3$, $K10=K \div 10=63 \div 10=6$, $(K1+K10) \bmod 2=(6+3) \bmod 2=9 \bmod 2=1$, ya'ni toq emas. Ikki xonali sonlar 10 dan boshlanadi va 99 da tugaydi. Dasturda 10 dan 99 gacha bo'lgan sonlarni bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mumkin.</p>	<pre> K, K1, K10:byte; Begin WriteLn('Bu sonlarni raqamlari yig'indisi juft:'); K:= 10; While K<=99 do BEGIN K1:=K mod 10; K10:=K div 10; IF (K1+K10) mod 2 =0 then Write(K, ' '); K:=K+1; END; Readln; End.</pre>
<p><u>Dasturi:</u> Var K, K1, K10:byte; Begin WriteLn('Bu sonlarni raqamlari yig'indisi juft:'); K:= 10; REPEAT K1:=K mod 10; K10:=K div 10; IF (K1+K10) mod 2 =0 then Write(K, ' '); K:=K+1; UNTIL K>99; Readln; End.</p>	<p><u>Dasturi:</u> Var K, K1, K10:byte; Begin WriteLn('Bu sonlarni raqamlari yig'indisi juft:'); For K:=10 to 99 do BEGIN K1:=K mod 10; K10:=K div 10; IF (K1+K10) mod 2 =0 then Write(K, ' '); END; Readln; End.</p>

T-7. A_1, A_2, \dots, A_N butun sonlar ketma-ketligi berilgan. Shu ketma-ketlikning toq elementlari ko'paytmasidan juft elementlari yig'indisini ayiruvchi dastur tuzing.

Yechim:

Bu masalada N natural son va A_1, A_2, \dots, A_N butun sonlar ketma-ketligi hadlari ($A[1..N]$ massiv elementlari) kiritiladi. Massiv elementlarini bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mumkin.

Masala shartiga ko'ra massivning toq qiymatli elementlarini **P** butun turdagi o'zgaruvchiga ko'paytirib boriladi, juft qiymatli elementlarini **S** butun turdagi o'zgaruvchiga yig'ib boriladi va natija sifatida ekranga **P-S** chiqariladi. Massiv elementlari qiymati toq yoki juft ekanini $A[K] \bmod 2 = 0$ shart yordamida tekshirish uchun to'liq ko'rinishdagi tarmoqlanish operatori qo'llaniladi.

N o'zgaruvchini qiymatlari natural son bo'lgani uchun **Word** turida. A_1, A_2, \dots, A_N ketma-ketlikni qiymatlari butun sonlar bo'lgani uchun **Integer** turida tavsiflash mumkin. Tavsiflashda massiv elementlari soni aniq ko'rsatilishi kerakligi uchun ularni sonini 1000 ta berish mumkin.

<u>Dasturi:</u>	<u>Dasturi:</u>
<pre> Var N, k: Word; S, P: LongInt; A: array[1..1000] of Integer; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[.k.]= '); readln(A[k]); end; k:=1; S:=0; P:=1; While k<=N do begin IF A[k] mod 2=0 Then S:=S+A[k] Else P:=P*A[k]; k:=k+1; end; Writeln('Javob: ', P-S); Readln; End.</pre>	<pre> Var N, k: Word; S, P: LongInt; A: array[1..1000] of Integer; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[.k.]= '); readln(A[k]); end; k:=1; S:=0; P:=1; Repeat IF A[k] mod 2=0 Then S:=S+A[k] Else P:=P*A[k]; k:=k+1; Until k>N; Writeln('Javob: ', P-S); Readln; End.</pre>

T-8*. N natural son va $A[1..N]$ massiv berilgan. K-elementi A massivning birinchi K ta elementining o'rtta arifmetigiga teng bo'lgan $B[1..N]$ massivni hosil qiluvchi dastur tuzing (yo'llanma: $B[K] = (A[1] + A[2] + \dots + A[K]) / K$).

Yechim:

Bu masalada N natural son va A_1, A_2, \dots, A_N haqiqiy sonlar ketma-ketligi hadlari ($A[1..N]$ massiv elementlari) kiritiladi. Massiv elementlarini bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mumkin.

Masala yechimi bo'lgan $B[1..N]$ massivni hosil qilish uchun quyidagi usuldan foydalanish qulay: $S=A[1]$; $B[1]=S/1$; $S=A[1]+A[2]$; $B[2]=S/2$; ...; $S=A[1]+A[2]+...+A[K]$; $B[K]=S/K$.

N o'zgaruvchini qiymatlari natural son bo'lgani uchun **Word** turida, $A[1..N]$ va $B[1..N]$ massivlarni qiymatlari haqiqiy sonlar (chunki boshqa xususiyat aytilmagan) bo'lgani uchun **Real** turida tavsiflash mumkin. Tavsiflashda massiv elementlari soni aniq ko'rsatilishi kerak, shuning uchun ularni sonini 1000 ta berish mumkin.

<p><u>Dasturi:</u></p> <pre> Var N, k: Word; S: Real; A, B: array[1..1000] of Real; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[',k,']= '); readln(A[k]); end; k:=1; S:=0; While k<=N do begin S:=S+A[k]; B[k]:=S/k; Writeln('B[',k,']= ', B[k]:0:5); k:=k+1; end; Readln; End.</pre>	<p><u>Dasturi:</u></p> <pre> Var N, k: Word; S: Real; A, B: array[1..1000] of Real; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[',k,']= '); readln(A[k]); end; k:=1; S:=0; Repeat S:=S+A[k]; B[k]:=S/k; Writeln('B[',k,']= ', B[k]:0:5); k:=k+1; Until k>N; Readln; End.</pre>
---	--

T-9*. 1 dan katta A son berilgan. $7^k > A$ shart bajariladigan eng kichik manfiy mas butun K sonni topish dasturini tuzing.

Yechim:

Bu masalada A musbat haqiqiy son kiritiladi. Kerakli eng kichik natural K ($A > 1$ bo'lgani uchun $K > 0$ bo'ladi) sonini aniqlash maqsadida 1 dan boshlab ($Dar7=$) $7^k > A$ shart bajarilguncha sanash uchun faqat shart bo'yicha takrorlash operatorlarini qo'llash mumkin. K va $Dar7$ o'zgaruvchini qiymatlari manfiy mas butun son bo'lgani uchun **Word** turida. A o'zgaruvchini qiymatlari musbat haqiqiy son bo'lgani uchun **Real** turida tavsiflash mumkin.

<p><u>Dasturi:</u></p> <pre> Var K, Dar7: Word; A: Real; Begin write('A= '); readln(A);</pre>	<p><u>Dasturi:</u></p> <pre> Var K, Dar7: Word; A: Real; Begin write('A= '); readln(A);</pre>
---	---

<pre> K:=1; Dar7:=7; While Dar7<=A do begin K:=K+1; Dar7:= 7*Dar7; end; WriteLn('K= ', K); Readln; End.</pre>	<pre> K:=0; Dar7:=1; Repeat Dar7:= 7*Dar7; K:=K+1; Until Dar7>A; WriteLn('K= ', K); Readln; End.</pre>
<p>Bu masala yechimini chiziqli strukturali ko'rinishda quyidagicha tasvirlash mumkin:</p> <pre> Var A: Real; K: Word; Begin write('A= '); readln(A); K:=trunc(ln(A)/ln(7))+1; WriteLn('K= ', K); Readln; End.</pre>	

37 – dars. Takrorlashga oid topshiriqlar mavzusiga

T-1. Quyidagi yig'indining qiymati berilgan M natural sondan ortiq bo'lguncha hisoblaydigan dastur tuzing:

$$y = \frac{1}{3} - \frac{1}{10} + \frac{1}{21} - \dots + \frac{(-1)^{j+1}}{j \cdot (2+j+1)} - \dots$$

Yechim:

Masala shartiga ko'ra qo'shiluvchilarni (har bir nisbatni ishorali va ishorasiz qo'shiluvchi deyish mumkin) **y** haqiqiy turdagi o'zgaruvchiga yig'ib boriladi. Ishorani ko'rsatish uchun **ishora** nomli o'zgaruvchi kiritilsa, yig'indidagi hadlar quyidagicha ifodalanadi:

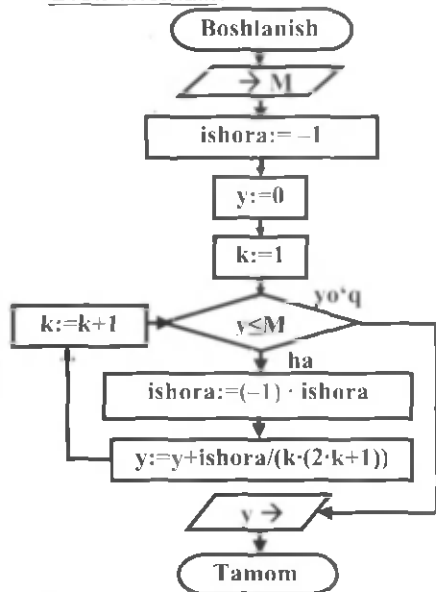
$$\begin{aligned} \text{ishora} &:= -1; \text{ishora} = (-1) * \text{ishora}, 1/3 = \text{ishora}/3; \text{ishora} = (-1) * \text{ishora}, \\ -1/10 &= \text{ishora}/10; \dots; \text{ishora} = (-1) * \text{ishora}, \\ (-1)^{k+1} / (k * (2 * k + 1)) &= \text{ishora} / (k * (2 * k + 1)). \end{aligned}$$

Har bir qo'shiluvchi qo'shilganda $y \leq M$ shart tekshiriladi. Shart tekshirish asosida takrorlanish tashkil etilayotgani uchun **While-do** shart bo'yicha takrorlash operatoridan foydalanish qulay. Kiritilayotgan M soni natural son bo'lgani uchun **Word** turida tavsiflash mumkin.

Kasrni mahrajidagi ko'paytma butun sonlar chegarasidan chiqib ketishini qisman oldini olish uchun quyidagicha o'zgartirish maqsadga muvofiq:

$$\frac{(-1)^{k+1}}{k \cdot (2 \cdot k + 1)} = \frac{(-1)^{k+1}}{k + 1} \cdot \left(\frac{1}{k} - \frac{1}{2 \cdot k + 1} \right)$$

Blok-sxemasi:



Dasturi:

Var
 M, k: Word; y, a, b, d: Real;
 ishora: ShortInt;
 Begin
 write('M= '); readln(M);
 ishora := -1;
 y := 0; k := 1;
 While y <= M do BEGIN
 ishora := (-1) * ishora;
 a := 1 / (k + 1); b := 1 / k;
 d := 1 / (2 * k + 1);
 y := y + ishora * a * (b - d);
 k := k + 1;
 END;
 Write('y = ', y);
 Readln;
 End.

T-2. 1-mashqni parametrli takrorlash operatori, shart bo'yicha takrorlash operatorlari yordamida turlicha hal eting.

Yechim:

Bu masalada takrorlanishni tashkil etishda parametrli takrorlash operatoridan foydalanish uchun avval takrorlanish parametri uchun oxirgi qiymatni aniq tanlab olish kerak bo'ladi. Lekin bu masalada takrorlanish parametri uchun oxirgi qiymatni M soniga bog'lab aniq tanlab olishning ilojisi yo'q. Shuning uchun parametr j ni oxirgi qiymatini Paskal dasturlash tilidagi eng katta butun son 2147483647 deb olishdan boshqa iloj yo'q. Bu holda j parametri LongInt turidagi o'zgaruvchi bo'ladi. Agar ifodalar ichida eng katta qiymat hosil qiluvchi $2 \cdot j + 1$ ifodani hisobga olinsa, u holda j parametri chegarasi $(2147483647 - 1) : 2 = 1073741823$ bo'lishi mumkin.

Repeat – Until shart bo'yicha takrorlash operatoridan foydalanish uchun avvalgi mashqdagi dasturni salgina o'zgartirish kifoya.

Avvalgi mashqdagi kabi har bir qo'shiluvchi qo'shilganda $y \leq M$

shart tekshiriladi. Kiritilayotgan M soni natural son bo'lgani uchun **Word** turida tavsiflanadi.

<p><u>Dasturi:</u> Label I; Var ishora: ShortInt; J: LongInt; M: Word; y, a, b, d: Real; Begin write(*M= *); readln(M); ishora:= -1; y:=0; For J:=1 to 1073741823 do BEGIN ishora:=(-1)*ishora; a:=1/(J+1); b:=1/J; d:=1/(2*j+1); y:=y+ishora*a*(b-d); If y>M then goto I; END; I: Write(*y= *, y); Readln; End.</p>	<p><u>Dasturi:</u> Var ishora: ShortInt; M, k: Word; y, a, b, d: Real; Begin write(*M= *); readln(M); ishora:= -1; y:=0; k:=1; Repeat ishora:=(-1)*ishora; a:=1/(k+1); b:=1/k; d:=1/(2*k+1); y:=y+ishora*a*(b-d); k:=k+1; Until y>M; Write(*y= *, y); Readln; End.</p>
--	--

T-3. N natural son berilgan. $(1/2)$, $(3/4)$, $(5/6)$, $(7/8)$, ... ketma-ketlikning N ta hadi yig'indisini topuvchi dastur tuzing.

<p><u>Yechim:</u> Bu masalada takrorlanishni tashkil etishda barcha takrorlash operatorlaridan foydalanish mumkin. Masalada e'tiborni ketma-ketlik hadi bilan ketma-ketlik hadi tartib raqami orasidagi bog'lanishga qaratish kerak: 1-had: $(2 \cdot 1 - 1) / (2 \cdot 1) = 1/2$; 2-had: $(2 \cdot 2 - 1) / (2 \cdot 2) = 3/4$; ...; k-had: $(2 \cdot k - 1) / (2 \cdot k)$.</p>	<p><u>Dasturi:</u> Var N, k: Word; S: Real; Begin write(*N= *); readln(N); S:=0; For k:=1 to N do S:=S+(2*k-1)/(2*k); Write(*S= *, S); Readln; End.</p>
<p><u>Dasturi:</u> Var N, k: Word; S: Real; Begin write(*N= *); readln(N); S:=0; k:=1; While k<=N do begin S:=S+(2*k-1)/(2*k);</p>	<p><u>Dasturi:</u> Var N, k: Word; S: Real; Begin write(*N= *); readln(N); S:=0; k:=1; Repeat S:=S+(2*k-1)/(2*k);</p>

<pre> k:=k+1; end; Write('S= ', S); Readln; End.</pre>	<pre> k:=k+1; Until k>N; Write('S= ', S); Readln; End.</pre>
---	---

T-4. A musbat son berilgan. Agar k-kvadratning tomoni $\frac{A}{k}$ bo'lsa ($k=1,2,\dots$), k ning qanday qiymatida barcha kvadratlarning yuzalari yig'indisi birinchi marta A^2 dan katta bo'lishini aniqlovchi dastur tuzing.

Yechim:

Bu masalada takrorlanishni tashkil etish uchun shart bo'yicha takrorlash operatorlaridan foydalanish qulay. Masala shartiga ko'ra 1-kvadratning tomoni $A/1=A$, 2-kvadratning tomoni $A/2$, 3-kvadratning tomoni $A/3$, va hokazo. K ta kvadratning yuzalari yig'indisi S bo'lsa, u holda $S=A^2+(A/2)^2+(A/3)^2+\dots+(A/k)^2$ bo'ladi. Takrorlanish sharti **While-do** uchun $S < A^2$, **Repeat-Until** uchun $S \geq A^2$.

<p><u>Dasturi:</u></p> <pre> Var K: Word; A, S: Real; Begin write('A= '); readln(A); K:=0; S:=0; While S<sqr(A) do BEGIN K:=K+1; S:=S+sqr(A/K); END; Write('K= ', K); Readln; End.</pre>	<p><u>Dasturi:</u></p> <pre> Var K: Word; A, S: Real; Begin write('A= '); readln(A); K:=0; S:=0; Repeat K:=K+1; S:=S+sqr(A/K); Until S>=sqr(A); Write('K= ', K); Readln; End.</pre>
---	--

T-5. P ga karrali va unga teng bo'lmagan biror uch xonali sonni aniqlovchi dastur tuzing.

<p><u>Yechim:</u></p> <p>Uch xonali sonlar 100 dan boshlanadi va 999 da tugaydi. Dasturda K butun o'zgaruvchi yordamida 100 dan 999 gacha bo'lgan sonlarni bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mum-</p>	<p><u>Dasturi:</u></p> <pre> Label 1; Var K, P: Word; Begin write('P= '); readln(P); K:= 100;</pre>
--	---

<p>kin. K soni berilgan P soniga karrali bo'lishi uchun $K \bmod P=0$ bo'lishi shart. K sonining P soniga teng bo'lmashlik sharti $K \neq P$. Masala shartida biror uch xonali son haqida so'z yuritilgan, shuning uchun talab qilingan birinchi son topilgach dasturdan chiqiladi.</p>	<pre>While K<=999 do BEGIN IF (K<>P) and (K mod P=0) then goto 1; K:=K+1; END; 1: Write('K= ', K); Readln; End.</pre>
<p><u>Dasturi:</u> Label 1; Var K, P: Word; Begin write('P= '); readln(P); K:=100; REPEAT IF (K<>P) and (K mod P=0) then goto 1; K:=K+1; UNTIL K>999; 1: Write('K= ', K); Readln; End.</p>	<p><u>Dasturi:</u> Label 1; Var K, P: Word; Begin write('P= '); readln(P); For K:=100 to 999 do IF (K<>P) and (K mod P=0) then goto 1; 1: Write('K= ', K); Readln; End.</p>

T-6. N natural son va $A[1..N]$ massiv berilgan. Massivning eng katta va eng kichik elementini topuvchi dastur tuzing.

<p><u>Yechim:</u> Masaladagi N natural son va A_1, A_2, \dots, A_N haqiqiy sonlar ketma-ketligi hadlari ($A[1..N]$ massiv elementlari) kiritiladi. Massivning birinchi elementini MAX va MIN deb olib massivning qolgan elementlarini bittalab qarab chiqish uchun ixtiyoriy takrorlash operatorini qo'llash mumkin. N o'zgaruvchini qiymati natural son bo'lgani uchun Word turida, $A[1..N]$ massivni qiymatlari</p>	<p><u>Dasturi:</u> Var N, k: Word; MAX, MIN: Real; A: array[1..1000] of Real; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[',k,']= '); readln(A[k]); end; MAX:=A[1]; MIN:=A[1]; k:=2; While k<=N do BEGIN If MAX<A[k] Then</p>
--	---

<p>haqiqiy sonlar (chunki boshqa xususiyat aytilmagan) bo'lgani uchun Real turida tavsiflash mumkin. Tavsiflashda massiv elementlari soni aniq ko'rsatilishi kerak, shuning uchun ularni sonini 1000 ta berish mumkin.</p>	<pre> MAX:=A[k]; If MIN>A[k] Then MIN:=A[k]; k:=k+1; END; WriteLn('MAX=',MAX:0:5); WriteLn('MIN=',MIN:0:5); Readln; End.</pre>
<p><u>Dasturi:</u></p> <pre> Var N, k: Word; MAX, MIN: Real; A: array[1..1000] of Real; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[' ,k,']= '); readln(A[k]); end; MAX:=A[1]; MIN:=A[1]; k:=2; Repeat If MAX<A[k] Then MAX:=A[k]; If MIN>A[k] Then MIN:=A[k]; k:=k+1; Until k>N; WriteLn('MAX=',MAX:0:5); WriteLn('MIN=', MIN:0:5); Readln; End.</pre>	<p><u>Dasturi:</u></p> <pre> Var N, k: Word; MAX, MIN: Real; A: array[1..1000] of Real; Begin write('N= '); readln(N); For k:=1 to N do begin write('A[' ,k,']= '); readln(A[k]); end; MAX:=A[1]; MIN:=A[1]; For k:=2 To N do BEGIN If MAX<A[k] Then MAX:=A[k]; If MIN>A[k] Then MIN:=A[k]; END; WriteLn('MAX=',MAX:0:5); WriteLn('MIN=', MIN:0:5); Readln; End.</pre>

T-7. Berilgan M natural sonni raqamlari yig'indisini topuvchi dastur tuzing.

<p><u>Yechim:</u></p> <p>Avvalroq ko'rilgan masalalardagi yo'llanmalarni aniq songa tatbiqi: N=349; S=0; 1) S=S+(N mod 10)=0+(349 mod 10)</p>	<p><u>Blok sxemasi (While):</u></p>
---	-------------------------------------

<p>$=0+9=9; N=N \text{ div } 10=349 \text{ div } 10=34;$ 2) $S=S+(N \text{ mod } 10)=9+(34 \text{ mod } 10)$ $=9+4=13; N=N \text{ div } 10=34 \text{ div } 10=3;$ 3) $S=S+(N \text{ mod } 10)=13+(3 \text{ mod } 10)$ $=13+3=16; N=N \text{ div } 10=3 \text{ div } 10=0.$</p> <p>Demak, kiritilgan sonni 10 ga bo'lgandagi qoldiqlarni S butun o'zgaruvchiga yig'iladi, sonni esa 0 ga aylanguncha 10 ga bo'lib boriladi. Dasturda $N>0$ bo'lguncha amallar bajarilayotgani sababli takrorlanishni tashkil etishda shart bo'yicha takrorlash operatorlaridan foydalanish maqsadga muvofiq.</p>	<pre> graph TD Start([Boshlanish]) --> Input[/N/] Input --> S0[S:=0] S0 --> Cond{N>0} Cond -- yo'q --> Output[/S ->/] Output --> End([Tamom]) Cond -- ha --> SCalc[S:=S+N-[N/10]*10] SCalc --> NCalc[N:=[N/10]] NCalc --> Cond </pre>
<p><u>Dasturi:</u> Var S: byte; N: word; Begin write('N= '); readln(N); S:= 0; While N>0 do begin S:=S+(N mod 10); N:=N div 10; end; Write('S= ', S); Readln; End.</p>	<p><u>Dasturi:</u> Var S: byte; N: word; Begin write('N= '); readln(N); S:= 0; Repeat S:=S+(N mod 10); N:=N div 10; Until N=0; Write('S= ', S); Readln; End.</p>

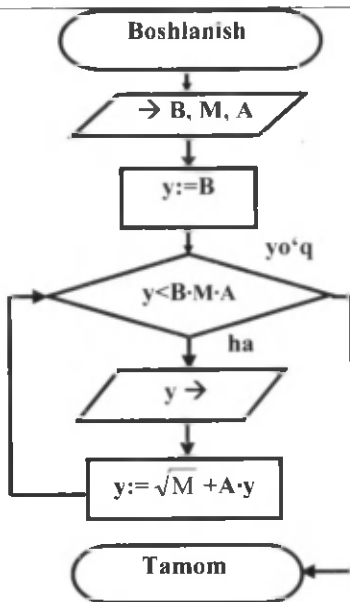
T-8*.B, M, A natural son berilgan. Ketma-ketlik $Y_1 = B;$
 $Y_i = \sqrt{M} + A \cdot Y_{i-1}, i=2,3,\dots$ qonuniyat asosida hosil qilinadi. Ketma ketlikning $B \cdot M \cdot A$ sonidan kichik barcha hadlarini chop etuvchi dastur tuzing.

<u>Yechim:</u>	<u>Blok sxemasi (While):</u>
----------------	------------------------------

Masala shartiga e'tibor berilsa dasturda massiv hosil qilish kerakdek ko'rinadi. Lekin, talab qilingan shart massivning nechanchi hadigacha bajarilib borishi noma'lum, ya'ni massiv elementlari soni avvaldan ma'lum emas. Berilgan formulalarda ikkinchi haddan boshlab har bir had faqat o'zidan oldingi hadga bog'liq ekanligi aks etgan. Shuning uchun ketma-ketlikni hadlarini birgina o'zgaruvchi orqali ifodalash mumkin:

1-had: $y:=B$; keyingi hadlar:
 $y:=\text{sqrt}(M)+A*y$;

Dasturda $y < B \cdot M \cdot A$ bo'lguncha amallar bajarilayotgani sababli takrorlanishni tashkil etishda shart bo'yicha takrorlash operatorlaridan foydalanish maqsadga muvofiq.



Dasturi:

Var B, M, A: word; y: real;
 Begin
 write('B= '); readln(B);
 write('M= '); readln(M);
 write('A= '); readln(A);
 writeln('Javob:');
 y:= B;
 While y<B*M*A do begin
 Write(y, ' '); y:=sqrt(M)+A*y; end;
 Readln;
 End.

Dasturi:

Var B, M, A: word; y: real;
 Begin
 write('B= '); readln(B);
 write('M= '); readln(M);
 write('A= '); readln(A);
 writeln('Javob:'); y:= B;
 Repeat Write(y, ' ');
 y:=sqrt(M)+A*y;
 Until y>=B*M*A;
 Readln;
 End.

T-9*. Mijoz bankka B so'm pul qo'ydi. Bankdagi pulga yiliga M foiz ustama qo'shiladi. Necha yildan keyin mijozning puli A so'mdan oshishini aniqlovchi dastur tuzing.

Yechim:

Masalani tahlili quyidagi natijalarni beradi:

Blok sxemasi (While):

<p>1 yildan so'ng bankdagi pul: $B[1]=B+B \cdot M/100$; 2 yildan so'ng bankdagi pul: $B[2]=B[1]+B[1] \cdot M/100$; 3 yildan so'ng bankdagi pul: $B[3]=B[2]+B[2] \cdot M/100$; ... Demak, bu masala ham avvalgi masala kabi yechilar ekan. Dasturda masala yechimi yil nomli byte turidagi o'zgaruvchi orqali ifodalangan.</p>	
<p><u>Dasturi:</u> Var B, M, A: word; yil: byte; Begin write('B= '); readln(B); write('M= '); readln(M); write('A= '); readln(A); yil:=0; While B<=A do begin yil:=yil+1; B:=B+B*M/100; end; Write('Yil= ', yil); Readln; End.</p>	<p><u>Dasturi:</u> Var B, M, A: word; yil: byte; Begin write('B= '); readln(B); write('M= '); readln(M); write('A= '); readln(A); yil:=0; Repeat yil:=yil+1; B:=B+B*M/100; Until B>A; Readln; End.</p>

T-10*. Kichik korxonada 1-kun B dona tovar ishlab chiqardi. Keyingi har bir kun oldingi kundagiga nisbatan M dona ortiq tovar ishlab chiqardi. Ishlab chiqarilgan barcha tovarning soni rejalangan A donadan birinchi marta necha kundan keyin ortganini aniqlovchi dastur tuzing.

<p><u>Yechim:</u> Barcha tovar butun turdagi S o'zgaruvchi orqali ifodalansa, u holda avval S=0 bo'ladi. Masalada ikkinchi kundan boshlab har kungi ishlab chiqarilayotgan tovar oldingi kundagidan M taga ortishi aytilgan.</p>	
<p>Agar bir kunda ishlab chiqari-</p>	<p>Blok sxemasi (While):</p>

layotgan tovar sonini BT butun turdagi o'zgaruvchilar orqali ifodalansa, u holda masala shartiga ko'ra:

1-kun: $BT=B$. $S=S+BT=0+B=B$;

2-kun: $BT=BT+M=B+M$,

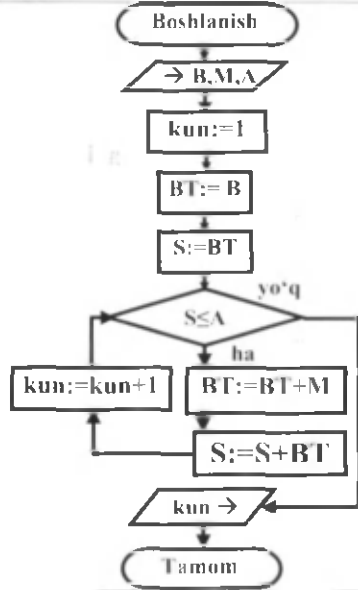
$S=S+BT=B+(B+M)=2\cdot B+M$;

3-kun: $BT=BT+M=B+M+M=B+2\cdot M$,

$S=S+BT=2\cdot B+M+(B+2\cdot M)=3\cdot B+3\cdot M$;

...

Demak, bu masala ham avvalgi masalalar kabi yechilar ekan. Dasturda masala yechimi **kun** nomli **Word** turidagi o'zgaruvchi orqali ifodalangan. Masalada B, M, A sonlar donani bildirgani uchun ular ham **Word** turidagi o'zgaruvchi bo'ladi.



Dasturi:

Var

B, M, A, BT, S, kun: word;

Begin

write('B= '); readln(B);

write('M= '); readln(M);

write('A= '); readln(A);

kun:=1; BT:=B;

S:=BT;

While S<=A do begin

BT:=BT+M;

S:=S+BT; kun:=kun+1;

end;

Write('Kun= ', kun); Readln;

End.

Dasturi:

Var

B, M, A, BT, S, kun: word;

Begin

write('B= '); readln(B);

write('M= '); readln(M);

write('A= '); readln(A);

kun:=0; BT:=B;

S:=0;

Repeat S:=S+BT;

BT:=BT+M; kun:=kun+1;

Until S>A;

Write('Kun= ', kun);

Readln;

End.

T-11*. Sanayotgan kishi atrofida doira shaklida B ta tartib raqami berilgan odam turibdi. Sanovchi M gacha sanab borgach, M-sanalgan odam doiradan chiqadi va sanovchi kishi keyingi odamni sanashni 1 dan boshlaydi. Sanash 1 ta odam qolguncha davom etadi. Oxirida nechanchi tartib raqamli odam qolganini aniqlovchi dastur tuzing.

Yechim:

Sanalayotgan (**Word** turdagi o'zgaruvchi) B ta odamning har biri tartib raqamiga ega bo'lgani uchun A_1, A_2, \dots, A_B butun sonlar massiviga quyidagicha mos qo'yish mumkin: massiv elementi qiymati odam doirada qolgan bo'lsa 1 ga, doiradan chiqqan bo'lsa 0 ga teng bo'lsin. Massiv elementlarini bittalab qarab chiqish uchun K nomli **Word** turdagi o'zgaruvchi kiritib, ixtiyoriy takrorlash operatorini qo'llash mumkin. Doiradan chiqish sanoqda ishtirok etmaslik bilan bir xil, ya'ni agar $A[K]=0$ bo'lsa, bu elementga mos K-odam doiradan chiqqan va sanoqda ishtirok etmaydi. Doirada qolgan odamlarni sonini saqlab turish uchun **Son** nomli, sanash uchun S nomli **Word** turdagi o'zgaruvchilar kiritish maqsadga muvofiq.

Masala yechimi quyidagicha aks etadi:

Avval barcha odam doirada bo'lgani uchun $A[1..B]$ massiv elementlariga 1 qiymat beriladi. Sanoq $A[1]$ dan boshlanadi va qiymati 1 ga teng har bir element qaralganda S ga bir qo'shiladi hamda $S=M$ shart tekshiriladi. Agar $S=M$ tenglik "Rost" qiymat qabul qilsa, u holda oxirgi qaralgan elementga 0 qiymat beriladi va $S=0$ deb olinadi, ya'ni sanoq boshidan boshlanadi. Agar $K=B$ bo'lsa, u holda $K=1$ deb olinadi, bu esa sanash doiraviy davom etishini bildiradi.

Dasturi:

```
Var B, M, K, S, Son: Word;  
A: array[1..10000] of byte;  
Begin  
  write('B= '); readln(B);  
  write('M= '); readln(M);  
  For K:=1 to B do A[K]:=1;  
  K:=0; S:=0; Son:=B;  
  While Son>1 do  
    BEGIN K:=K+1;  
      IF A[K]=1 THEN S:=S+1;  
      IF S=M THEN begin  
        Son:=Son-1; A[K]:=0;  
        S:=0;      end;  
      IF K=B then K:=0;  
    END;  
  For K:=1 to B do
```

Dasturi:

```
Var B, M, K, S, Son: Word;  
A: array[1..10000] of byte;  
Begin  
  write('B= '); readln(B);  
  write('M= '); readln(M);  
  For K:=1 to B do A[K]:=1;  
  K:=0; S:=0; Son:=B;  
  Repeat  
    K:=K+1;  
    IF A[K]=1 THEN S:=S+1;  
    IF S=M THEN begin  
      Son:=Son-1; A[K]:=0; S:=0;  
      end;  
    IF K=B then K:=0;  
  Until Son=1;  
  For K:=1 to B do
```


IF A[K]=1 then WriteLn('Qolgan odam tartib raqami=', K); Readln; End.	IF A[K]=1 then WriteLn('Qolgan odam tartib raqami=', K); Readln; End.
---	---

38 – dars. Belgili va satrli miqdorlar bilan ishlash mav- zusiga

M-1. Quyidagi funksiyalar bajarilish natijasini aniqlang.

a) Concat('o', 'na')='o'+ 'na'='ona'; – so‘zlarini bir-biriga uladi
b) Concat('ya', 'sha', 'sin')='ya'+ 'sha'+ 'sin'='yashasin'; – so‘zlarini bir-biriga uladi
d) a:='dunyo'; Length(a)=Length('dunyo')=5; – “dunyo” so‘zidagi belgilar soni
e) Pos('o', 'bahor')= 4; – “o” belgisi “bahor” matnida birinchi marta 4-o‘rinda kelgan

M-2. Satrli o‘zgaruvchi S ning qiymati ‘Informatika’ bo‘lsin. Quyidagi amallar bajarilgach uning qiymatini aniqlang:

a) Delete(s,5,7)= Delete('Informatika',5,7)= Delete('Informatika',5,7)= 'Info'; – ‘Informatika’ matnini 5-belgisidan boshlab 7 ta belgisini o‘chirdi
b) Delete(s,1,2)=Delete('Informatika',1,2)='formatika'; – ‘Informatika’ matnini 1-belgisidan boshlab 2 ta belgisini o‘chirdi Delete(s,6,4)=Delete('formatika',6,4)= Delete('formatika',6,4)= 'forma'; – ‘formatika’ matnini 6-belgisidan boshlab 4 ta belgisini o‘chirdi

M-3. Kiritilgan so‘zdan “ona” so‘zini hosil qilish imkoniyatini aniqlovchi dastur tuzing.

Yechim:

Kiritilayotgan so‘z A satrda saqlanadi. Hosil qilinishi kerak bo‘lgan “ona” so‘zi 3 ta “o”, “n” va “a” belgilaridan iborat. Dasturning vazifasi shu uchta belgi bir vaqtda A satrda bor yoki yo‘qligini aniqlash. Agar shu belgilardan birortasi A satrda uchramasa, u holda “ona” so‘zini hosil qilib bo‘lmaydi.

Masala yechimini beruvchi dasturni turlicha tuzish mumkin. Masa-

lan:

1) Dastur agar A satrda "o" belgisi uchrasa (ya'ni, $\text{Pos}('o', A) > 0$) $s=1$ deb oladi, keyin A satrda "n" belgisi uchrasa (ya'ni, $\text{Pos}('n', A) > 0$) $s=s+1$ va A satrda "a" belgisi uchrasa (ya'ni, $\text{Pos}('a', A) > 0$) $s=s+1$ deb oladi. Agar $s=3$ bo'lsa, demak barcha belgilar bor, ya'ni "ona" so'zini hosil qilish mumkin, aks holda mumkin emas.

2) Ma'lumki, matnda biror belgi qatnashmasa shu belgini izlayotgan Pos funksiyasi qiymati 0 ga teng bo'ladi. Shuning uchun $\text{Pos}('o', A) \cdot \text{Pos}('n', A) \cdot \text{Pos}('a', A) > 0$ bo'lsagina "ona" so'zini hosil qilish mumkin, aks holda mumkin emas.

Dasturi:

Var s: byte; A: String;

Begin

Write('So'zni kiriting: ');

Readln(A);

IF $\text{Pos}('o', A) > 0$ THEN $s:=1$;

IF $\text{Pos}('n', A) > 0$ THEN $s:=s+1$;

IF $\text{Pos}('a', A) > 0$ THEN $s:=s+1$;

IF $s=3$ THEN

Write('Hosil qilish mumkin')

ELSE

Write('Hosil qilish mumkin emas');

Readln;

End.

Dasturi:

Var

A: String;

Begin

Write('So'zni kiriting: ');

Readln(A);

IF $\text{Pos}('o', A) * \text{Pos}('n', A) * \text{Pos}('a', A) > 0$ THEN

Write('Hosil qilish mumkin');

ELSE

Write('Hosil qilish mumkin emas');

Readln;

End.

39 – dars. Belgili va satrli miqdorlar bilan ishlash mavzusini takrorlash mavzusiga

T-1. Berilgan so'zning belgilari orasiga bittadan probel qo'shib chiquvchi dastur tuzing.

Yechim:

Kiritilayotgan so'z A satrda saqlanadi. Satr uzunligi 255 ta belgidan oshmaydi, shuning uchun uni uzunligini **byte** turidagi o'zgaruvchiga o'zlashtirish mumkin. Agar belgilar orasiga probel qo'shilsa, satr uzunligi 255 dan ortib ketishi mumkin. Shuning uchun probel belgisini chiqarish operatorida qo'shish maqsadga muvofiq bo'ladi.

Masala yechimini beruvchi dasturni turlicha tuzish mumkin. Masalan:

1) A satrning uzunligiga mos biror takrorlash operatori yordamida satrning k-belgisi nusxasi Copy funksiyasi orqali B satrga olinib chiqarish operatorida Write(B, ' ') kabi yoziladi.

2) Ma'lumki, satr massiv xususiyatiga ega. Shuning uchun Copy funksiyasidan foydalanmasdan A satrning k-belgisini A[k] kabi ifodalash mumkin va chiqarish operatorida Write(A[k], ' ') kabi yoziladi.

<u>Dasturi:</u>	<u>Dasturi:</u>
Var A, B: String; k: byte; Begin Write('So`zni kiriting: '); Readln(A); For k:=1 to Length(A) do begin B:=Copy(A, k, 1); Write(B, ' '); end; Readln; End.	Var A: String; k: byte; Begin Write('So`zni kiriting: '); Readln(A); For k:=1 to Length(A) do Write(A[k], ' '); Readln; End.

T-2. S satr berilgan. Undagi "b" harflari sonini aniqlovchi dastur tuzing.

Yechim:

Kiritilayotgan so'z S satrda saqlanadi. Satr uzunligi 255 ta belgidan oshmaydi, shuning uchun uni uzunligini **byte** turidagi o'zgaruvchiga o'zlashtirish mumkin. Masala yechimini beruvchi dasturni turlicha tuzish mumkin. Masalan:

1) A satrning uzunligiga mos parametrli takrorlash operatori yordamida satrning k-belgisi "b" belgi bilan taqqoslanadi, ya'ni $S[k]='b'$, va tenglik bajarilsa sanoq bittaga oshiriladi. Bu holda satrning har bir belgisi "b" belgi bilan taqqoslanadi.

2) Satrdagi "b" belgisi o'rni Pos funksiyasi yordamida aniqlanadi. Keyin sanoq bittaga oshiriladi va topilgan belgi o'chiriladi. Jarayon toki satrda "b" belgisi batamom o'chirilmaguncha, ya'ni $Pos('b', S) > 0$ bo'lsa, takrorlanaveradi. Tushunarliki, bu holda shart bo'yicha takrorlash operatori **While** dan foydalaniladi. Bu usulda bajariladigan amallar soni anchagina kamayadi.

Ikkala usulni $S='bombarbia'$ bo'lganda taqqoslaymiz: 1-usulda 9 marta k ni o'zgartirish, 9 marta taqqoslash va 3 marta sanoq bajariladi, jami 21 ta amal; 2-usulda 4 marta tekshirish, 3 marta sanoq va 3 marta

o'chirish, ja'mi 10 ta amal.

Dasturi:

```
Var S: String; k, sanoq: byte;
Begin
  sanoq:=0;
  Write('So'zni kiriting: ');
  Readln(S);
  For k:=1 to Length(S) do
    If S[k]='b' then sanoq:=sanoq+1;
  Write('b harflari soni ',sanoq);
  Readln;
End.
```

Dasturi:

```
Var S: String; sanoq: byte;
Begin sanoq:=0;
  Write('So'zni kiriting: ');
  Readln(S);
  While Pos('b', S)>0 do begin
    sanoq:=sanoq+1;
    delete(S, Pos('b', S),1); end;
  Write('b harflari soni ', sanoq);
  Readln;
End.
```

T-3. A[1..N] satrli massiv berilgan. Massiv elementlari ichidan "m" harfdan boshlanadiganlarini ekranga chiqaruvchi dastur tuzing.

Yechim:

N natural son va A[1..N] satrli massiv kiritiladi. Parametrlri takrorlash operatori yordamida massivni har bir elementining 1-belgisi "m" belgi bilan taqqoslanadi, ya'ni $A[k][1]='m'$ (yoki $\text{Copy}(A[k],1,1)='m'$), va tenglik bajarilsa mos element ekranga chiqariladi. Massiv elementlari soni 253 tadan ortiq e'lon qilinsa, Paskal dasturi ma'lumotlar xotira chegarasidan chiqib ketganligi haqida xabar beradi.

Dasturi:

```
Var N, k: integer;
  A: array[1..253] of String;
Begin Write('N ni kiriting: ');
  Readln(N);
  For k:=1 to N do begin
    Write('A[', k, ']='');
    Readln(A[k]); end;
  For k:=1 to N do begin
    If A[k][1]='m' Then
      Write(A[k], ' ');
    end;
  Readln;
End.
```

Dasturi:

```
Var N, k: integer;
  A: array[1..253] of String;
Begin Write('N ni kiriting: ');
  Readln(N);
  For k:=1 to N do begin
    Write('A[', k, ']='');
    Readln(A[k]); end;
  For k:=1 to N do begin
    If Copy(A[k],1,1)='m' Then
      Write(A[k], ' ');
    end;
  Readln;
End.
```

T-4*. S satr faqat raqamlardan iborat. Satrdagi raqamlardan eng katta sonni hosil qiladigan dastur tuzing.

Yechim:

Misol uchun S= "5520122" bo'lsin. U holda eng katta son SS="5522210" bo'ladi. Demak, satrdagi raqamlardan eng katta sonni hosil qilish uchun uni raqamlarini kamayish tartibida yozish kerak ekan. Bu vazifani turli usullarda bajarish mumkin. Masalan:

1) B= "9876543210" satr kiritiladi. Avval B satrning birinchi belgisiga teng S satr belgilarini barchasini SS satrga, keyin ikkinchi belgisiga teng belgilarini va hokazo.

2) S satrni kamayish tartibida saralanadi. Paskal dasturlash tili belgilarni ham taqqoslash imkoniyatiga ega bo'lib, bunda dastur taqqoslashni belgilarni ASCII kodi asosida bajaradi. Masalan, '7'<'5' belgilarni taqqoslash uchun Paskal dasturi bu belgilarni ASCII kodini (7→55, 5→53) olib 55<53 kabi taqqoslaydi.

Dasturi:

```
Var
S, SS, B: String; N, k, m: byte;
Begin
Write('S ni kiriting: '); Readln(S);
B:= '9876543210';
N:=Length(S); SS:= "";
For k:=1 to 10 do
For m:=1 to N do
If B[k]=S[m] Then
SS:=SS+S[m];
Write('Javob: ', SS);
Readln;
End.
```

Dasturi:

```
Var
N, k, m: byte;
S: String; B: char;
Begin Write('S ni kiriting: ');
Readln(S); N:=Length(S);
For k:=1 to N-1 do
For m:=k+1 to N do
If S[k]<S[m] Then begin
B:=S[k]; S[k]:=S[m];
S[m]:=B; end;
Write('Javob: ', S);
Readln;
End.
```

T-5*. A satrdagi belgilarni faqat o'rnini almashtirib B satrni hosil qilish mumkin yoki yo'qligini aniqlovchi dastur tuzing.

Yechim:

Masalan:

A="Lola" va B="lola" bo'lsa, "Mumkin emas"; A="Lola" va B="oLla" bo'lsa, "Mumkin"; A="lola" va B="loola" bo'lsa, "Mumkin emas"; A="lolaa" va B="loola" bo'lsa, "Mumkin emas". Misollardan

ko'rish mumkinki, agar A va B satrlarni uzunligi teng bo'lmasa, javob "Mumkin emas". Shuning uchun A va B satrlarni uzunliklari teng holni qarash yetarli.

Bu masalani turli usullarda hal etish mumkin. Masalan:

1) A va B satr belgilari o'sish (yoki kamayish) tartibida saralanadi va taqqoslanadi. Agar saralangan satrlar teng bo'lsa "Mumkin", aks holda "Mumkin emas".

2) A satrni har bir belgisi Pos funksiyasi yordamida B satrdan izlanadi: agar B satrda izlanayotgan belgi uchramasa, "Mumkin emas", agar uchrasa A satrdan ham B satrdan ham o'chiriladi va jarayon A satr bo'sh satrga aylanguncha davom ettiriladi. Bu holda B satr ham bo'sh satrga aylanadi.

Dasturi:

```
Label 1,2;
Var A, B: String; D: char;
N, k, m: byte;
Begin
  Write('A satrni kiriting: ');
  Readln(A);
  Write('B satrni kiriting: ');
  Readln(B); N:=Length(A);
  IF N<>Length(B) Then goto 1;
  For k:=1 to N-1 do
    For m:=k+1 to N do BEGIN
      If A[k]>A[m] Then begin
        D:=A[k]; A[k]:=A[m];
        A[m]:=D; end;
      If B[k]>B[m] Then begin
        D:=B[k]; B[k]:=B[m];
        B[m]:=D; end;
    END;
  IF A=B THEN begin
    Write('Mumkin'); goto 2; end;
  1: Write('Mumkin emas ');
  2: Readln;
End.
```

Dasturi:


```
Label 1, 2;
Var
  A, B: String;
Begin
  Write('A satrni kiriting: ');
  Readln(A);
  Write('B satrni kiriting: ');
  Readln(B);
  IF Length(A)<>Length(B)
  Then goto 1;
  While A<> "" do
  BEGIN
    If Pos(A[1], B)>0 THEN
      begin
        Delete(B,Pos(A[1],B),1);
        Delete(A, 1, 1);
      end
    ELSE goto 1;
  END;
  Write('Mumkin'); goto 2;
  1: Write('Mumkin emas ');
  2: Readln;
End.
```

40 – dars. Paskalda ekranni grafik holatga o'tkazish mavzusiga

M-1. Ekranni grafik holatga o'tkazuvchi va `Enter` klavishi bosilganda yana matn holatiga qaytaruvchi dastur tuzing.

<p><u>Yechim:</u> Ekranni grafik holatga o'tkazish uchun dasturning asosiy qismida <code>InitGraph(Gd, Gm, "")</code> prosedurasi yoziladi, grafik holatdan chiqish uchun esa <code>CloseGraph</code> prosedurasidan foydalaniladi. Dasturni <code>Enter</code> klavishi bosilguncha kuttirish uchun <code>CloseGraph</code> prosedurasidan avval <code>Readln;</code> operatori yoziladi.</p>	<p><u>Dasturi:</u> <code>Uses Graph;</code> <code>Var</code> <code>Gd, Gm: Integer;</code> <code>Begin</code> <code>gd:=0;</code> <code>initgraph(gd, gm, "");</code> <code>Readln;</code> <code>CloseGraph;</code> <code>End.</code></p>
--	---

M-2. Ekranning to'rtta burchagida sariq rangli nuqta hosil qiluvchi dastur tuzing.

<p><u>Yechim:</u> Grafik ekran (0..639x0..479) nuqtalar to'plamidan iborat. Shuning uchun ekranni burchakdagi nuqtalari quyidagilar: chap yuqori nuqta (0,0); o'ng yuqori nuqta (639,0); chap quyi nuqta (0,479); o'ng quyi nuqta (639,479). Ekran grafik holatga o'tkazilgach (avvalgi mashqdagi kabi) PutPixel(X,Y,Rang) protsedurasi yordamida nuqtalar joylashtiramiz. Bunda nuqtalar sariq rangda aks etishi uchun rangni Yellow (yoki 14) kabi tanlanadi. Dasturni <code>Enter</code> klavishi bosilguncha kuttirish uchun <code>CloseGraph</code> prosedurasidan avval <code>Readln;</code> operatori yoziladi.</p>	
<p><u>Dasturi:</u> <code>Uses graph;</code> <code>var gd, gm: integer;</code> <code>begin</code> <code>gd:=0; initgraph(gd, gm, "");</code> <code>PutPixel(0,0,yellow);</code> <code>PutPixel(639,0,yellow);</code> <code>PutPixel(0,479,14);</code> <code>PutPixel(639,479,14);</code> <code>readln; closegraph;</code> <code>end.</code></p>	<p><u>Natijaning ekrandagi ko'rinishi:</u></p> 

M-3. Nuqtalar yordamida ekranni o'rtasidan bo'luvchi gorizontaal chiziq hosil qiling.

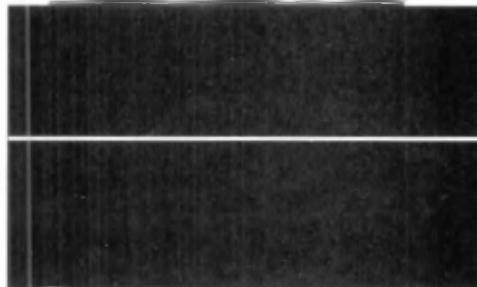
Yechim:

Ekran grafik holatga o'tkazilgach **PutPixel(X,Y,Rang)** protsedurasi va biror takrorlash operatori, masalan, parametrli takrorlash operatori yordamida nuqtalar joylashtiriladi. Nuqtalarni gorizontaal yo'nalishda ekranni o'rtasidan ($Y=[480/2]=240$) chapdan o'ngga qarab joylashtirish uchun **X** ni qiymatini **0** dan **639** gacha bittalab o'zgartirish kerak. Nuqtalar oq rangda aks etishi uchun rangni White (yoki 15) kabi tanlanadi. Dasturni **Enter** klavishi bosilguncha kuttirish uchun CloseGraph protsedurasidan avval Readln; operatori yoziladi.

Dasturi:

```
Uses graph;
var gd, gm, x: integer;
begin
  gd:=0; initgraph(gd, gm, "");
  for x:=0 to 639 do
    PutPixel(x, 240, 15);
  readln; closegraph;
end.
```

Natijaning ekrandagi ko'rinishi:



M-4. Random funksiyasidan foydalanib turli rangli nuqtalarni hosil qilish dasturini tuzing.

Yechim:

Ekran grafik holatga o'tkazilgach **PutPixel(X,Y,Rang)** protsedurasi va biror takrorlash operatori, masalan, parametrli takrorlash operatori yordamida **N** ta (masalan, 500 ta) nuqta joylashtiriladi. Nuqtalarni joylashtirish uchun **X** va **Y** o'zgaruvchilar qiymatini $X:=\text{Random}(640)$; $Y:=\text{Random}(480)$; formulalar yordamida o'zgartiriladi, chunki $\text{Random}(640)$ funksiyasi 0 dan 639 gacha, $\text{Random}(480)$ funksiyasi 0 dan 479 gacha bo'lgan butun sonlarni hosil qiladi. Random funksiyasi yordamida har bir (X, Y) juftlik hosil qilinayotganda qiymatlar turlicha bo'lishi uchun dasturning asosiy qismiga **Randomize** protsedurasi kiritiladi. Ekrandagi nuqtalar ham turli rangda bo'lishi uchun $\text{Rang}:=\text{Random}(16)$; formulasini qo'llash mumkin.

Dasturi:

```
Uses graph;
```

Natijaning ekrandagi ko'rinishi:


```

var gd, gm: integer;
  k, X, Y, Rang: integer;
begin Randomize; gd:=0;
  initgraph(gd, gm, "");
  for k:=1 to 500 do BEGIN
    X:=Random(640); Y:=Random(480);
    Rang:=Random(16);
    PutPixel(X, Y, Rang); END;
  readln; closegraph;
end.

```



M-5. $x \in [-10, 10]$ oraliqda $y=3x+5$ funksiyasining grafigini chizuvchi dastur tuzing.

Yechim:

Ekran grafik holatga o'tkazilgach **PutPixel(X,Y,Rang)** protsedurasi va shart bo'yicha takrorlash operatori (chunki o'zgarish qadami haqiqiy son), masalan, While yordamida nuqtalar joylashtiriladi. Nuqtalarni zichroq joylashtirish uchun **X** ni qiymatini **-10** dan **10** gacha **0.01** qadam bilan o'zgartirib, bu qiymatga mos **Y** ning qiymati $y:=3*x+5$; formula yordamida aniqlanadi. Nuqtalar oq rangda aks etishi uchun rangni White (yoki 15) kabi tanlanadi. Masshtabni kattalashtirish uchun x ni 10 ga, y ni 5 ga ko'paytirish kerak. Ma'lumki, grafik ekranda ordinata qiymati yuqoridan quyiga o'sadi. Shu sababli to'g'ri chiziq nuqtalari Dekart koordinatalar sistemasidagi kabi aks etishi uchun "-" ishora qo'yilgan. Grafikni ekran markazidan o'tishni ta'minlash uchun birinchi koordinataga $(640/2=)$ 320 ni, ikkinchi koordinataga $(480/2=)$ 240 ni qo'shish maqsadga muvofiq.

Dasturi:

```

Uses Graph;
var gd, gm: Integer; x, y, a, b: real;
Begin gd:= 0;
  InitGraph(gd, gm, ""); x:=-10;
  While x<=10 do BEGIN
    y:=3*x+5; a:=trunc(10*x+320);
    b:= trunc(-5*y+240);
    putpixel(a,b,15); x:=x+0.01; END;
  Readln; CloseGraph;
end.

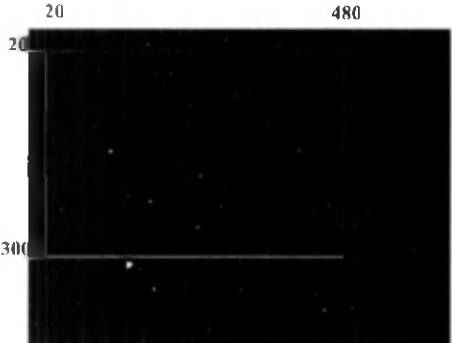
```

Natijaning ekrandagi ko'rinishi:



41 – dars. Paskalda ekranni grafik holatga o‘tkazish mavzusini takrorlash mavzusiga

T-1. Tomonlarini rangi turlicha bo‘lgan to‘g‘ri to‘rtburchak chizish dasturini tuzing.

<p><u>Yechim:</u> Ekran grafik holatga o‘tkazilgach PutPixel(X,Y,Rang) protsedurasi va parametrlilik takrorlash operatorlari yordamida kerakli nuqtalar joylashtiriladi.</p>	
<p><u>Dasturi:</u> Uses Graph; var gd, gm: Integer; x, y: integer; Begin gd:= 0; InitGraph(gd, gm, ""); For x:=20 to 480 do BEGIN PutPixel(x, 20, blue); PutPixel(x, 300, green); END; For y:=20 to 300 do BEGIN PutPixel(20, y, yellow); PutPixel(480, y, red); END; Readln; CloseGraph; end.</p>	<p><u>Natijaning ekrandagi ko‘rinishi:</u></p> 

T-2. Grafik koordinatasi bilan berilgan nuqta uchlarining grafik koordinatalari orqali berilgan to‘g‘ri chiziqqa tegishli yoki tegishli emasligini aniqlovchi dastur tuzing (yo‘llanma: nuqtani rangi to‘g‘ri chiziqning rangiga tengligini aniqlash uchun GetPixel funksiyasidan foydalaning).

<p><u>Yechim:</u> Ekran grafik holatga o‘tkazilgach, nuqtaning koordinatalari (x0,y0) va to‘g‘ri chiziq uchlarining koordinatalari (x1,y1) va (x2,y2) kiritiladi. Endi ikki nuqtadan o‘tuvchi to‘g‘ri chiziq tenglamasidan foydalanib PutPixel(X,Y,Rang) protsedurasi va parametrlilik takrorlash operatorlari yordamida kodi 0 bo‘lgan qora rangdan boshqa rangda, masalan, kodi 15 bo‘lgan oq rangda to‘g‘ri chiziq chiziladi. GetPixel(X,Y) funksiyasi yordamida (x0,y0) nuqtaning rangi aniqlanadi. Agar nuqta rangi kodi 15 (oq) bo‘lsa, nuqta to‘g‘ri chiziqqa tegishli, aks holda, ya‘ni nuqta rangi kodi 0 (qora) bo‘lsa, nuqta to‘g‘ri chiziqqa tegishli emas.</p> <p>Ikki nuqta (x1,y1) va (x2,y2) dan o‘tuvchi to‘g‘ri chiziq tenglamasi quyidagicha: $Y=y1+(X-x1)\cdot(y2-y1)/(x2-x1)$.</p> <p>Geometriya kursidan ma‘lumki, agar $x2=x1$ bo‘lsa, u holda to‘g‘ri vertikal bo‘lib, Y ning 0 dan 479 gacha bo‘lgan qiymatida $X=x1$ tenglama bilan</p>

aniqlanadi. Agar $x_2 \neq x_1$ bo'lsa, yuqoridagi formula o'z kuchini saqlab qoladi.

Dasturi:

```
Uses Graph;
var gd, gm: Integer; x, y: real;
x0, y0, x1, y1, x2, y2: integer;
Begin
gd:= 0; InitGraph(gd, gm, "");
Write('x0= '); Readln(x0);
Write('y0= '); Readln(y0);
Write('x1= '); Readln(x1);
Write('y1= '); Readln(y1);
Write('x2= '); Readln(x2);
Write('y2= '); Readln(y2);
IF x2=x1 THEN
For Y:=0 to 479 do
PutPixel(x1, trunc(Y), 15);
ELSE
BEGIN X:= 0;
While x<=639 do Begin
Y:=y1+(X-x1)*(y2-y1)/(x2-x1);
putpixel(trunc(X), trunc(Y),15);
X:=X+0.01; End;
END;
IF GetPixel(x0,y0)=15 THEN
Write('Tegishli') ELSE
Write('Tegishli emas');
Readln; CloseGraph;
end.
```

Natijaning ekrandagi ko'rinishi:

$$X=x_1$$



$$Y=y_1+(X-x_1)\cdot(y_2-y_1)/(x_2-x_1)$$



$$Y=y_1+(X-x_1)\cdot(y_2-y_1)/(x_2-x_1)$$



T-3. Turli rangda 15 ta parallel kesmalar chizuvchi dastur tuzing (yo'llanma: kesma koordinatalarini va rangni oshirish uchun takrorlash operatoridan foydalaning).

Yechim:

Ekran grafik holatga o'tkazilgach **PutPixel(X,Y,Rang)** protsedurasi va parametrli takrorlash operatorlari yordamida kerakli nuqtalar joylashtiriladi. Ranglarni 1 dan 15 gacha parametrli takrorlash operatori yordamida o'zgartirish qulay (kodi 0 bo'lgan rang olinsa kesma ko'rinmaydi, chunki grafik holatga o'tilganda avtomatik ravishda ekran foni rangi kodi 0 deb belgilanadi).

Dasturi:

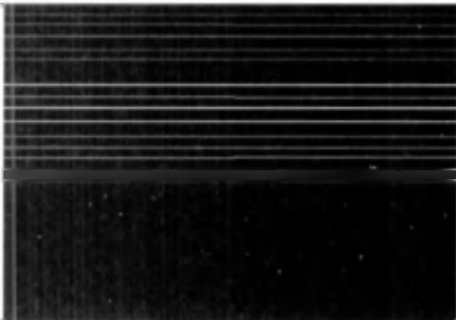
Uses Graph;

Natijaning ekrandagi ko'rinishi:

```

var gd, gm, x, y, rang: Integer;
Begin gd:= 0;
InitGraph(gd, gm, "");
For rang:=1 to 15 do
For y:=1 to 15 do
For x:=10 to 620 do
PutPixel(x, 10*y, rang);
ReadLn; CloseGraph;
end.

```



T-4. Random funksiyasi yordamida “yulduzli osmon” manzarasini hosil qilish dasturini tuzing.

Bu masalani yechimi avvalgi darsdagi 4-mashqning yechimi bilan bir xil.

T-5. Ekran markazidan o'tuvchi koordinalar o'qi, mos joyda koordinatalar o'qi nomini yozuvchi va $x \in [-7, 7]$ oraliqda $y=|x|$ funksiyasining grafigini chizuvchi dastur tuzing.

Yechim:

Ekran grafik holatga o'tkazilgach **PutPixel(X,Y,Rang)** protsedurasi va shart bo'yicha takrorlash operatori (chunki o'zgarish qadami haqiqiy son), masalan. While yordamida nuqtalar joylashtiriladi. Nuqtalarni zichroq joylashtirish uchun **X** ni qiymatini -7 dan 7 gacha **0.01** qadam bilan o'zgartirib, bu qiymatga mos **Y** ning qiymati $y=abs(x)$; formula yordamida aniqlanadi. Nuqtalar oq rangda aks etishi uchun rangni White (yoki 15) kabi tanlanadi. Masshtabni kattalashtirish uchun x ni 30 ga, y ni 20 ga ko'paytirish kerak. Ma'lumki, grafik ekranda ordinata qiymati yuqoridan quyiga o'sadi. Shu sababli to'g'ri chiziq nuqtalari Dekart koordinatalar sistemasidagi kabi aks etishi uchun “-” ishora qo'yilgan. Grafikni ekran markazidan o'tishni ta'minlash uchun birinchi koordinataga $(640/2=)$ 320 ni, ikkinchi koordinataga $(480/2=)$ 240 ni qo'shish maqsadga muvofiq.

Parametrli takrorlash operatorlari yordamida qizil rangdagi koordinata o'qlarini ekran markazi (320, 240) nuqtadan o'tkaziladi. Ekran piksellarda o'lchangani va hisob (0,0) nuqtadan boshlangani uchun **OuttextXY(A, B, 'matn')**: protsedurasi yordamida ekranning chap yuqori burchagidan gorizontaal bo'yicha A, vertikal bo'yicha B piksel uzoqlikda “matn” chiqariladi.

Dasturi:

```

Uses Graph;
var k, gd, gm: Integer; x, y, a, b: real;
Begin gd:=0; InitGraph(gd, gm, "");

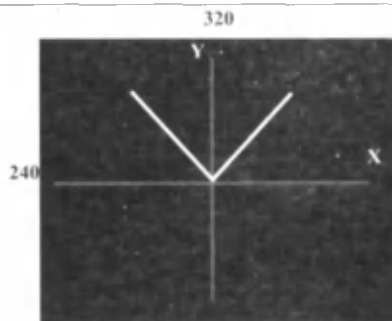
```

Natijaning ekrandagi ko'rinishi:

```

x:=-7;
While x<=7 do BEGIN y:=abs(x);
a:=30*x+320; b:=-20*y+240;
putpixel(trunc(a), trunc(b),15);
x:=x+0.01; END;
For k:=20 to 460 do PutPixel(320,k,4);
For k:=20 to 620 do PutPixel(k,240,4);
OuttextXY(300,10,'Y');
OuttextXY(620,220,'X');
Readln; CloseGraph;
End.

```



T-6*. Ichma-ich joylashgan 7 ta aylana chizuvchi dastur tuzing (yo'llanma: radiusni oshirish uchun takrorlash operatoridan foydalaning).

Yechim:

Ekran grafik holatga o'tkazilgach radiusi R ga teng aylana chizish dasturini darslikdagi kabi tuzib olinadi. Bunda **PutPixel(X,Y,Rang)** protsedurasi va shart bo'yicha takrorlash operatori (chunki o'zgarish qadami haqiqiy son), masalan, While yordamida nuqtalar joylashtiriladi. Nuqtalarni zichroq joylashtirish uchun **alfa** burchakni qiymati 0 dan 2π gacha 0.01 qadam bilan o'zgartiriladi. Nuqtalar sariq rangda aks etishi uchun rangni Yellow (yoki 14) kabi tanlanadi. Grafikni ekran markazidan o'tishni ta'minlash uchun birinchi koordinataga $(640/2=)$ 320 ni, ikkinchi koordinataga $(480/2=)$ 240 ni qo'shiladi.

Endi aylanalar sonini 7 taga yetkazish uchun shunday o'zgartirish qilinadi: dasturning aylana chizish qismi While yordamida 7 marta takrorlanishi uchun radius o'zgarish qadamini 20 va chegarani 140 kabi tanlanadi ($7\cdot 20=140$).

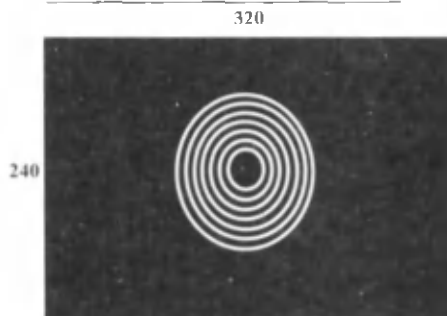
Dasturi:

```

Uses Graph;
Var gd, gm: integer;
x, y: Integer; R, alfa: real;
Begin Gd:=0;
InitGraph(Gd, Gm, ""); R:=20;
While R<=140 do BEGIN alfa:=0;
While alfa<=2*pi do begin
x:= 320 +trunc(R*cos(alfa));
y:= 240 +trunc(R*sin(alfa));
putpixel(x,y,14); alfa:=alfa+0.01;
end; R:=R+20; END;
Readln; CloseGraph;

```

Natijaning ekrandagi ko'rinishi:



T-7*. 7 marta o'chib-yonadigan aylana chizuvchi dastur tuzing (yo'llanma: aylana chizing va jarayonni sekinlashtirish uchun bo'sh takrorlash bajaring, avvalgi aylananing fon rangida chizing va jarayonni sekinlashtirish uchun bo'sh takrorlash bajaring, takrorlashni 7 marta bajartiring).

Yechim:

Ekran grafik holatga o'tkazilgach radiusi 50 ga teng aylana chizish dasturi tuzib olinadi. Bunda **PutPixel(X,Y,Rang)** protsedurasi va shart bo'yicha takrorlash operatori (chunki o'zgarish qadami haqiqiy son), masalan, While yordamida nuqtalar joylashtiriladi. Nuqtalarni zichroq joylashtirish uchun **alfa** burchakni qiymati 0 dan 2π gacha 0.01 qadam bilan o'zgartiriladi. Nuqtalar sariq rangda aks etishi uchun rangni Yellow (yoki 14) kabi tanlanadi. Grafikni ekran markazidan o'tishni ta'minlash uchun birinchi koordinataga ($640/2=$) 320 ni, ikkinchi koordinataga ($480/2=$) 240 ni qo'shiladi.

Endi yonib o'chishni tashkil etish uchun aylana chizish qismini nusxalab qora rang tanlanadi (ekran foni rangi tanlansa fon rangida aylana chiziladi, ya'ni aylana ko'rinmaydi). Paskal dasturlash tili juda katta tezlikda ishlagani uchun aylana chizilgani va o'chirilganini ko'rishga ulgurilmaydi. Shu sababli vaqtni cho'zish uchun parametrli takrorlash operatori yordamida bo'sh takrorlanishlar tashkil etiladi. O'chib yonish 7 marta takrorlanishi uchun parametrli takrorlash operatoridan foydalanish mumkin.

Paskal dasturlash tilida dasturni biror qismida vaqtni sekinlashtirish uchun Crt moduliga tegishli Delay(N) protsedurasi (N word turida) ishlatiladi. Delay(N) yozilgan joyda dastur N millisekund vaqt kutadi.

Dasturi:

```
uses graph;
var gd, gm, x, y, m, n: integer;
k: byte; alfa: real;
Begin gd:=0; Initgraph(gd, gm, "");
for k:=1 to 7 do BEGIN
for m:= 1 to 20000 do
for n:= 1 to 20000 do;
alfa:=0;
While alfa<=2*pi do begin
x:= 320 +trunc(50*cos(alfa));
y:= 240 +trunc(50*sin(alfa));
putpixel(x, y, 14);
alfa:= alfa+0.01; end;
```

Dasturi:

```
uses graph, crt;
var gd, gm, x, y: integer;
k: byte; alfa: real;
Begin gd:=0; initgraph(gd, gm, "");
for k:=1 to 7 do BEGIN
delay(65000); delay(65000);
delay(65000); delay(65000);
alfa:=0;
While alfa<=2*pi do begin
x:= 320 +trunc(50*cos(alfa));
y:= 240 +trunc(50*sin(alfa));
putpixel(x, y, 14);
alfa:= alfa+0.01; end;
```

<pre> for m:= 1 to 20000 do for n:= 1 to 30000 do; alfa:=0; While alfa<=2*pi do begin x:= 320 +trunc(50*cos(alfa)); y:= 240 +trunc(50*sin(alfa)); putpixel(x, y, 0); alfa:= alfa+0.01; end; END; readln; closegraph; end.</pre>	<pre> delay(65000); delay(65000); delay(65000); delay(65000); alfa:=0; While alfa<=2*pi do begin x:= 320 +trunc(50*cos(alfa)); y:= 240 +trunc(50*sin(alfa)); putpixel(x, y, 0); alfa:= alfa+0.01; end; END; readln; closegraph; end.</pre>
--	---

42 – dars. Paskalda shakllar chizish imkoniyatlari mavzusiga

M-1. Ekranni o'rtasidan o'tuvchi gorizontal va vertikal chiziq chizuvchi dastur tuzing.

Yechim:

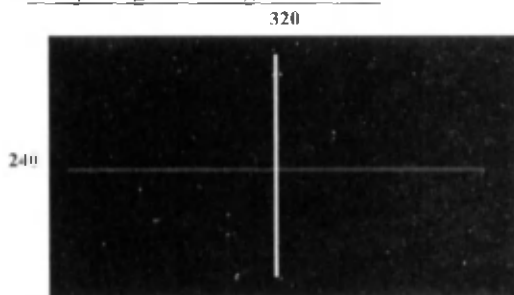
Gorizontal chiziq rangi qizil bo'lishi uchun **SetColor(red)** va vertikal chiziq rangi sariq bo'lishi uchun **SetColor(yellow)**: protseduralari qo'llaniladi. Ma'lumki, **Line(X1,Y1,X2,Y2)** protsedurasi ekranning (X1,Y1) koordinatali nuqtasi bilan (X2,Y2) koordinatali nuqtasini birlashtiruvchi joriy rangli kesma chizadi. Shunga ko'ra gorizontal chiziqni 20 pikseldan 600 pikselgacha chizdirish uchun X1=20, X3=600, ekran markazidan o'tishni ta'minlash uchun ikkinchi va to'rtinchi koordinataga (480/2=) 240 yoziladi. Vertikal chiziqni 20 pikseldan 460 pikselgacha chizdirish uchun X2=20, X4=460, ekran markazidan o'tishni ta'minlash uchun birinchi va uchinchi koordinataga (640/2=) 320 yoziladi.

Dasturi:

```

Uses Graph;
var gd, gm: Integer;
Begin gd:= 0;
InitGraph(gd, gm, "");
SetColor(red);
Line(20,240,600,240);
SetColor(yellow);
Line(320,20,320,460);
Readln; CloseGraph;
end.
```

Natijaning ekrandagi ko'rinishi:



M-2. Ekranning o'rtasida radiusi 100 dan kichik sariq rangli 4 ta aylana chizuvchi dastur tuzing.

Yechim:

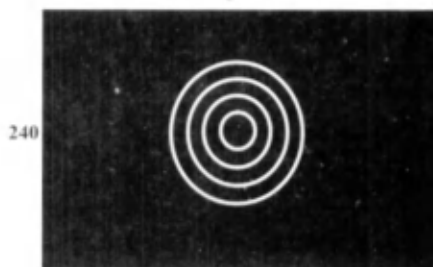
Aylanalarni chizg'i rangi sariq bo'lishi uchun barcha aylanaga bitta **Setcolor**(yellow); protsedurasi qo'llaniladi. Ma'lumki, **Circle**(X,Y,R) protsedurasi markazi (X,Y) nuqtada va radiusi R ga teng aylana chizadi. Shuning uchun barcha aylana uchun **Circle**(X,Y,R) protsedurasida markazni X=320, Y=240 kabi tanlanadi. Endi 4 ta radiusni 100 dan kichik bo'lgan sonlar, masalan, 40, 55, 70, 85 kabi tanlash mumkin. Radiusni takrorlash operatori yordamida ham o'zgartirish mumkin.

Dasturi:

```
Uses Graph;
var gd, gm: Integer;
Begin gd:=0; InitGraph(gd, gm, "");
Setcolor(yellow);
Circle(320, 240, 45);
Circle(320, 240, 60);
Circle(320, 240, 75);
Circle(320, 240, 90);
Readln; CloseGraph;
End.
```

Natijaning ekrandagi ko'rinishi:

320



Dasturi (For):

```
Uses Graph;
var gd, gm: Integer; k: byte;
Begin gd:=0; InitGraph(gd, gm, "");
Setcolor(yellow);
For k:=1 to 4 do
Circle(320, 240, k*15);
Readln; CloseGraph;
End.
```

Dasturi:

```
Uses Graph;
var gd, gm, k: byte;
Begin gd:= 0; InitGraph(gd, gm, "");
Setcolor(yellow); k:= 45;
While k<100 do begin
Circle(320, 240, k); k:=k+15; end;
Readln; CloseGraph;
End.
```

M-3. Ekranni sariq gorizontaal chiziqlar bilan to'ldiring.

Yechim:

Bu masalani turli xil usullarda hal etish mumkin. Masalan:

1) **Setcolor**(yellow); protsedurasi yordamida sariq rangli gorizontaal to'g'ri chiziqlar **Line**(0, 3-Y, 639; 3-Y) protsedurasida Y ni 0 dan 159 gacha bittalab oshirgan holda ($159 \cdot 3 = 477 < 479$);

2) To'g'ri to'rtburchak chizib, uning ichini bo'yash uchun rang va usulni **SetFillStyle**(usul, rang); protsedurasi yordamida tanlash orqali, bu yerda rang – tanlanayotgan rang kodi, usul – bo'yash usuli. **Bar**(X1.Y1.X2.Y2) protsedurasi ekranda yuqori chap burchagi (X1,Y1) va quyi o'ng burchagi (X2,Y2)

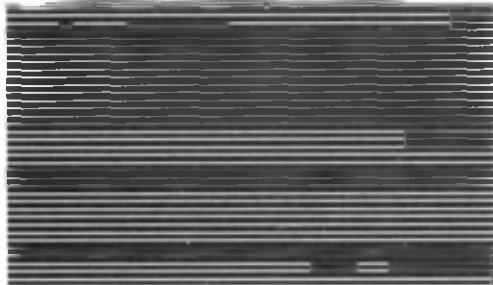
koordinatali nuqtalarda bo'lgan, ichi joriy rang va joriy usulda bo'yalgan to'g'ri to'rtburchak chizadi. Joriy rangni sariq, ya'ni Yellow (kodi 14), bo'yash usulini 2, ya'ni qalin gorizontal chiziqlar bilan to'ldirish, $X1=0$, $Y1=0$, $X2=639$, $Y2=479$ kabi tanlash masala yechimini beradi;

3) **Bar3D**($X1,Y1,X2,Y2,a,b$) protsedurasi joriy rang va joriy usulda bo'yalgan parallelepiped chizadi, bu yerda a – parallelepiped yon tomonining uzunligi, b esa mantiqiy ifoda bo'lib, uning qiymati "rost" bo'lsa parallelepipedning yuqori qirrasini chiziladi, "yolg'on" bo'lsa chizilmaydi. Shuning uchun **SetFillStyle**(2,14); $a=0$, $b=False$, $X1=0$, $Y1=0$, $X2=639$, $Y2=479$ kabi tanlash masala yechimini beradi.

Dasturi (For):

```
Uses Graph;
var gd, gm: Integer; Y:byte;
Begin
gd:= 0; InitGraph(gd, gm, "");
Setcolor(yellow);
For Y:=0 to 159 do
Line(0, 3*Y, 639; 3*Y);
Readln; CloseGraph;
End.
```

Natijaning ekrandagi ko'rinishi:



Dasturi (Bar):

```
uses graph;
Var gd, gm: integer;
begin
gd := 0; InitGraph(gd, gm, "");
setfillstyle(2, yellow);
Bar(0,0,639,479);
Readln; CloseGraph;
end.
```

Dasturi (Bar3D):



```
Uses Graph;
var gd, gm: integer;
Begin
gd:= 0; InitGraph(gd, gm, "");
SetFillStyle(2,14);
Bar3D(0,0,639,479,0,False);
Readln; CloseGraph;
end.
```

M-4. Svetofor rasmini chizuvchi dastur tuzing.

Yechim:

Bu masalani turli xil usullarda hal etish mumkin. Masalan:

1) **Bar**($X1,Y1,X2,Y2$) protsedurasi yordamida 2 ta to'g'ri to'rtburchak, **FillEllipse**(X,Y,XR,YR) protsedurasi yordamida 3 ta doira chizib (doira chizish uchun $XR=YR$ bo'lishi kerak), ularning ichini bo'yash uchun rang va usulni **SetFillStyle**(usul,rang); protsedurasi yordamida tanlash orqali. Bo'yash usuli 1 deb olinadi, chunki shakl ichi to'liq bo'yaladi. Ranglar esa doiralarda uchun 4 (qizil), 14 (sariq), 2 (yashil) tanlanadi. Dasturda doiralarga chegara chizig'i rangi **Setcolor**(rang) protsedurasi yordamida tanlanmagani uchun Paskal dasturi chegara rangini oq (kodi 15) rangda chizadi.

2) To'g'ri to'rtburchaklar chizish uchun **Rectangle(X1,Y1,X2,Y2)** protsedurasidan va doiralalar chizish uchun **Circle(X,Y,R)** protsedurasidan foydalanish mumkin. **SetFillStyle(usul,rang)**; protsedurasidan bu shakllarni ichini bo'yash usuli va rangini tanlashda, tanlangan rangni shakl ichiga "quyish" (MS Paint dasturidagi  kabi) uchun **FloodFill(A,B,D)** protsedurasidan foydalanish mumkin, bu yerda (A,B) shakl ichidagi biror nuqta, D chegara chizig'i rangi. Dasturda chegara chizig'i rangi **SetColor(rang)** protsedurasi yordamida tanlanmagan bo'lsa, u holda Paskal dasturi chegara rangini oq (kodi 15) rangda chizadi. Agar shakl ichini bo'yashda **FloodFill(A,B,D)** protsedurasidan foydalanayotganda (A,B) shakl ichiga tegishli bo'lmasa, u holda rang MS Paint dasturidagi  yordamida bo'yash kabi "yoyilib" ketadi.

Dasturi:

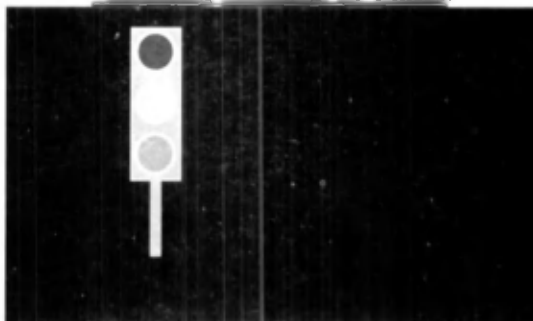
Uses graph;

Var gd, gm: Integer;

```
Begin gd := 0; InitGraph(gd, gm, ""); setfillstyle(1,6); bar(150,10,210,190);
  SetFillStyle(1,red); fillEllipse(180,40,30,30); SetFillStyle(1,14);
  fillEllipse(180,100,30,30); SetFillStyle(1,2); fillEllipse(180,160,30,30);
  setfillstyle(1,6); bar(175,190,185,300); Readln; CloseGraph;
```

End.

Natijaning ekrandagi ko'rinishi:



Dasturi (Floodfill):

Uses graph;

Var gd, gm: integer;

Begin gd:=0; InitGraph(gd, gm, "");

```
  Setfillstyle(1,6); rectangle(150,10,210,190); floodfill(160,100,15);
  SetFillStyle(1,4); circle(180,40,25); floodfill(180,40,15);
  SetFillStyle(1,14); circle(180,100,25); floodfill(180,100,15);
  SetFillStyle(1,2); circle(180,160,25); floodfill(180,160,15);
  Setfillstyle(1,6); rectangle(175,190,185,300); floodfill(180,200,15);
```

```
Readln; CloseGraph;
End.
```

M-5. Qizil rangli muntazam beshburchak chizuvchi dastur tuzing.

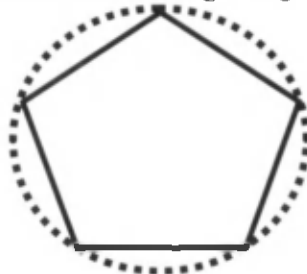
Yechim:

Masaladagi muntazam beshburchak uchlari koordinatalarini aniqlash uchun quyidagicha mulohaza o'rinli:

Muntazam beshburchakka tashqi aylana chizish mumkin. Aylanani bir "aylanib" chiqish 360^0 yoki $2\cdot\pi$ ga teng. Bu burchakni 5 ga bo'linsa, aylananing beshta teng yoyga ajratuvchi nuqtalar orasidagi burchak $2\cdot\pi/5$ aniqlanadi. Ekran markazini koordinata boshi hisoblab muntazam beshburchakning yuqori uchi koordinatalarini radiusi 100 ga teng aylananing nuqtasi sifatida

$x:= 320 +\text{trunc}(100\cdot\cos(-\pi/2)); y:= 240 +\text{trunc}(100\cdot\sin(-\pi/2));$

kabi aniqlash mumkin. Muntazam beshburchakning keyingi har bir nuqtasi shu formulada $-\pi/2$ ga, mos ravishda, $1\cdot(2\cdot\pi/5)=2\cdot\pi/5$, $2\cdot(2\cdot\pi/5)=4\cdot\pi/5$, $3\cdot(2\cdot\pi/5)=6\cdot\pi/5$, $4\cdot(2\cdot\pi/5)=8\cdot\pi/5$ ni qo'shish orqali hosil qilinadi.



Endi bu nuqtalarni **Line(X1,Y1,X2,Y2)** protsedurasi yoki **FillPoly(BS,KM)** protsedurasi yordamida tutashtirish mumkin. Chiziq rangini tanlash uchun **SetColor(4)**, fon rangini tanlash uchun esa **SetBkColor(14)** protseduralaridan foydalaniladi.

Dasturi (Line):

Uses Graph;

Var gd, gm: integer; X1,Y1, X2, Y2: Integer; X3,Y3, X4,Y4, X5, Y5: Integer;

Begin Gd:= 0; InitGraph(Gd, Gm, ""); Setbkcolor(14); Setcolor(4);

$X1:=320+\text{trunc}(100\cdot\cos(-\pi/2)); Y1:=240+\text{trunc}(100\cdot\sin(-\pi/2));$

$X2:=320+\text{trunc}(100\cdot\cos(-\pi/2+2\cdot\pi/5));$

$Y2:=240+\text{trunc}(100\cdot\sin(-\pi/2+2\cdot\pi/5));$

$X3:=320+\text{trunc}(100\cdot\cos(-\pi/2+4\cdot\pi/5));$

$Y3:=240+\text{trunc}(100\cdot\sin(-\pi/2+4\cdot\pi/5));$

$X4:=320+\text{trunc}(100\cdot\cos(-\pi/2+6\cdot\pi/5));$

```

Y4:=240+trunc(100*sin(-pi/2+6*pi/5));
X5:=320+trunc(100*cos(-pi/2+8*pi/5));
Y5:=240+trunc(100*sin(-pi/2+8*pi/5));
Line(X1,Y1,X2,Y2); Line(X2,Y2,X3,Y3); Line(X3,Y3,X4,Y4);
Line(X4,Y4,X5,Y5); Line(X5,Y5,X1,Y1); Readln; CloseGraph;
End.

```

Dasturi (FillPoly):

Uses Graph;

Var gd, gm : Integer; km : Array[1..5,1..2] of Integer;

Begin Gd:= 0; InitGraph(Gd, Gm, ""); Setbkcolor(14); Setcolor(4);

km[1,1]:=320+trunc(100*cos(-pi/2));

km[1,2]:=240+trunc(100*sin(-pi/2));

km[2,1]:=320+trunc(100*cos(-pi/2+2*pi/5));

km[2,2]:=240+trunc(100*sin(-pi/2+2*pi/5));

km[3,1]:=320+trunc(100*cos(-pi/2+4*pi/5));

km[3,2]:=240+trunc(100*sin(-pi/2+4*pi/5));

km[4,1]:=320+trunc(100*cos(-pi/2+6*pi/5));

km[4,2]:=240+trunc(100*sin(-pi/2+6*pi/5));

km[5,1]:=320+trunc(100*cos(-pi/2+8*pi/5));

km[5,2]:=240+trunc(100*sin(-pi/2+8*pi/5));

FillPoly(5,km); Readln; CloseGraph;

End.

43 – dars. Paskalda shakllar chizish imkoniyatlari mavzusini takrorlash mavzusiga

T-1. O'zbekiston bayrog'i rasmini chizuvchi dastur tuzing.

Yechim:

Bayroq rasmi to'g'ri to'rtburchaklar birlashmasi ko'rinishida tasvirlanadi.

To'g'ri to'rtburchaklar chizish uchun **Bar**(X1,Y1,X2,Y2); protsedurasidan, bo'yash uchun **SetFillStyle**(usul,rang); protsedurasidan foydalanish qulay. Bayroqdagi oy va yulduzlar chizish protseduralar mavzusidan keyin bajarilishi maqsadga nuvofiq.

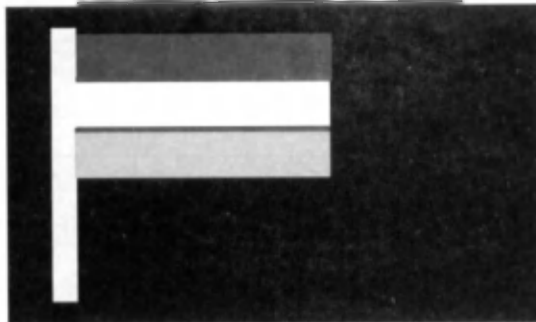
Dasturi:

```

Uses graph;
Var gd, gm:integer;
Begin gd := 0; InitGraph(gd, gm, ""); setfillstyle(1,LightBlue);
bar(50,50,380,100); setfillstyle(1,Red); bar(50,100,380,103);
setfillstyle(1,White); bar(50,103,380,153); setfillstyle(1,Red);
bar(50,153,380,156); setfillstyle(1,Green); bar(50,156,380,206);
setfillstyle(1,Yellow); bar(40,40,50,350); Readln; CloseGraph;
end.

```

Natijaning ekrandagi ko'rinishi:



T-2. Ekranning to'rtta burchagida eni 60 va bo'yi 40 ga teng qizil rangli to'rtburchaklar chizuvchi dastur tuzing.

Yechim:

Ekranning o'lchamlari 0..639x0..479 bo'lgani uchun to'g'ri to'rtburchaklar diagonallari uchlari koordinatalari quyidagicha aniqlanadi: chap yuqori burchakdagi to'g'ri to'rtburchakning diagonallari uchlari (0,0) va (60,40); o'ng yuqori burchakdagi to'g'ri to'rtburchakning diagonallari uchlari (639-60,0)=(579,0) va (639,40); chap quyi burchakdagi to'g'ri to'rtburchakning diagonallari uchlari (0,479-40)=(0,439) va (60,479); o'ng quyi burchakdagi to'g'ri to'rtburchakning diagonallari uchlari (639-60,479-40)=(579,439) va (639,479).

To'g'ri to'rtburchaklar chizish uchun **Bar(X1,Y1,X2,Y2)**; chegarasi rangini belgilash uchun **SetColor(rang)**, ichini bo'yash uchun **SetFillStyle(usul,rang)**; protsedurasidan foydalanish qulay.

Dasturi:

```

Uses graph;
Var gd, gm:integer;
Begin gd := 0; InitGraph(gd, gm, "");
setcolor(4); setfillstyle(1, red);
bar(0,0,60,40); bar(579,0,639,40);

```

Natijaning ekrandagi ko'rinishi:

```
bar(0,439,60,479);
bar(579,439,639,479);
Readln; CloseGraph;
end.
```



Izoh: Paskal dasturlash tilining ba'zi modullari zamonaviy monitorlarning grafik ekran imkoniyatlarini to'liq aks ettira olmasligi sababli ekranning gorizontal yo'nalishdagi piksellari soni $0..479 \times 0..639$ emas, masalan, $0..432 \times 0..632$ bo'lishi mumkin. Shuning uchun yuqoridagi kabi dasturlarni tahrirlash zarur bo'ladi.

T-3. Ekranni teng to'rt bo'lakka bo'lib, ularni mos ravishda qizil, sariq, yashil va ko'k ranglarga bo'yovchi dastur tuzing.

Yechim:

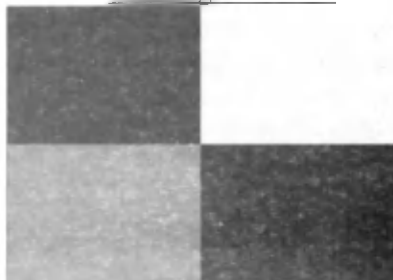
Ekranning o'lchamlari $0..639 \times 0..479$ bo'lgani uchun to'g'ri to'rtburchaklar yordamida 4 bo'lakka bo'lish mumkin. To'g'ri to'rtburchaklar diagonallari uchlari koordinatalari quyidagicha aniqlanadi: (0,0) va (320,240); (320,0) va (639,240); (0,240) va (320,479); (320,240) va (639,479).

To'g'ri to'rtburchaklar chizish uchun **Bar(X1,Y1,X2,Y2)**; chegarasi rangini belgilash uchun **SetColor(rang)**, ichini bo'yash uchun **SetFillStyle(usul,rang)**; protsedurasidan foydalanish qulay.

Dasturi:

```
Uses graph;
Var
gd, gm:integer;
Begin gd := 0; InitGraph(gd, gm, "");
setfillstyle(1,red); bar(0,0,320,240);
setfillstyle(1,yellow);
bar(320,0,639,240);
setfillstyle(1,green); bar(0,240,320,479);
setfillstyle(1,blue);
bar(320,240,639,479);
Readln; CloseGraph;
end.
```

Natijaning
ekrandagi ko'rinishi:



T-4. Ekranning o'rtasida radiusi 100 ga teng sariq rangli doira chizuvchi dastur tuzing.

Yechim:

Markazi ekran markazida va radiusi 100 piksel bo'lgan doira chizish uchun **FillEllipse**(320,240,100,100); chegara chizig'i rangi sariq bo'lishi uchun **SetColor**(yellow); ichini rangi sariq bo'lishi uchun **SetFillStyle**(1, yellow); protseduralari qo'llaniladi.

Dasturi:

Uses Graph;

var

gd, gm: Integer;

Begin

gd:= 0; InitGraph(gd, gm, "");

SetColor(yellow);

SetFillStyle(1, yellow);

FillEllipse(320,240,100,100);

Readln; CloseGraph;

end.

Natijaning ekrandagi ko'rinishi:



T-5. Oy va yulduzlar tasvirlangan kechki osmon manzarasini chizuvchi dastur tuzing.

Yechim:

Ekranda oq rangli oy ko'rinishini hosil qilish uchun ichi oqqa bo'yalgan doira bilan ichi fon rangi qoraga bo'yalgan doira kesishtiriladi. Shunda birinchi doiraning kerakmas qismi qora rangda aks etgani uchun ko'rinmaydi va oy ko'rinishi hosil bo'ladi. Doirani oq gardishi ko'rinmasligi uchun chegara rangini ham fon rangida olinishi muhim. Turli rangli yulduzlarni hosil qilish avval ko'rilgan edi. Yulduzlar sonini ko'paytirish uchun ichma-ich joylashgan parametrlil takrorlash operatoridan foydalanish qulay ($50 \cdot 15 = 750$ ta).

Dasturi:

Uses graph;

Var

gd, gm: integer; m, n :byte;

Begin

gd:=0; Initgraph(gd, gm, "");

setcolor(0); setfillstyle(1,15); fillellipse(200,200,50,50);

setfillstyle(1,0); fillellipse(225,200,45,45);

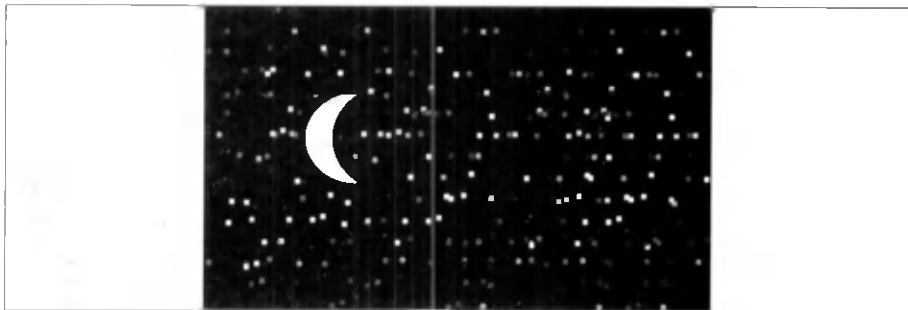
for m:= 1 to 50 do BEGIN for n:=1 to 15 do begin

putpixel(random(639), random(479), n); end; END;

Readln; CloseGraph;

End.

Natijaning ekrandagi ko'rinishi:



T-6. Dengiz uzra nur taratayotgan Quyosh rasmini chizuvchi dastur tuzing. Dengizni chizishda yoy chizish operatoridan foydalaning.

Yechim:

Ekran foniga och billur rang (kodi 11) tanlanib, radiusi 50 piksel bo'lgan chegarasi va ichi sariq rangli doira chiziladi: Setcolor(14); Setfillstyle(1,14); Fillellipse(510,100,50,50). Quyosh nurlarini chizish uchun quyosh markazidan chiquvchi chiziqlar olinishi hisoblashni osonlashtirgani uchun Line(500,100, X, Y); protsedurasi yordamida X va Y larni tanlash orqali chiziladi.

Mavjlanayotgan dengizni chizish uchun **Ellipse(X,Y,BB,OB,XR,YR)** protsedurasidan foydalaniladi, bu yerda markazi (X,Y) nuqtada, x va y o'qlari bo'yicha radiuslari mos ravishda XR va YR ga teng ellipsning BB burchagidan OB burchagigacha bo'lgan yoyini chizadi. Burchak gradus o'lehov birligida beriladi. Dengizga rang berish uchun **Floodfill** protsedurasidan foydalaniladi.

Dasturi:

Uses graph;

Var gd, gm: integer;

Begin gd:=0; Initgraph(gd, gm, "");

Setbkcolor(11); Setfillstyle(1,yellow);

Setcolor(yellow); Fillellipse(510,100,50,50); line(510,100,510,10);

line(510,100,510,190); line(510,100,420,100); line(510,100,600,100);

line(510,100,445,40); line(510,100,575,40); line(510,100,445,150);

line(510,100,575,150); Setcolor(blue); Setfillstyle(1, blue);

ellipse(40,300,22,145,70,45); ellipse(193,265,200,340,95,50);

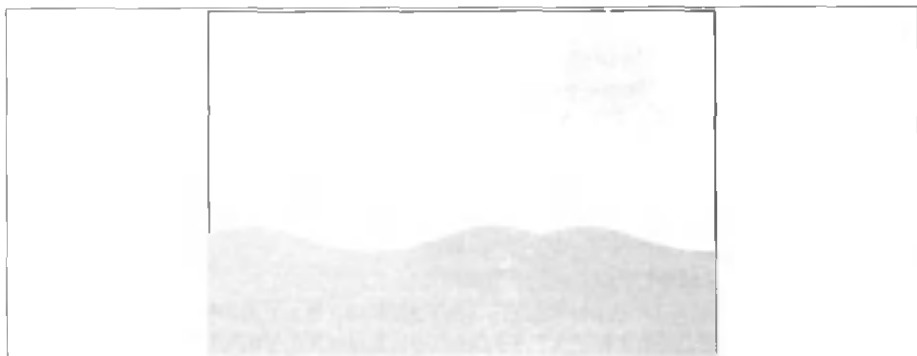
ellipse(340,307,40,146,70,45); ellipse(455,300,22,148,70,45);

ellipse(608,265,200,340,95,50); Floodfill(40,340,blue);

Readln; CloseGraph;

End.

Natijaning ekrandagi ko'rinishi:



T-7. Ichi siyrak qizil nuqtalar bilan to'ldirilgan muntazam oltiburchak chizuvchi dastur tuzing.

Yechim:

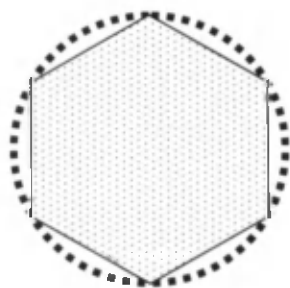
Masaladagi muntazam oltiburchak uchlari koordinatalarini aniqlash uchun quyidagicha mu-lohaza o'rinli:

Muntazam oltiburchakka tashqi aylana chizish mumkin. Aylanani bir "aylanib" chiqish 360^0 yoki $2\cdot\pi$ ga teng. Bu burchakni 6 ga bo'lsa, aylananani oltita teng yoyga ajratuvchi nuqtalar orasidagi bur-chak $2\cdot\pi/6=\pi/3$ aniqlanadi. Ekran markazini koor-dinata boshi hisoblanb muntazam oltiburchakning yuqori uchi koordinatalarini radiusi 100 ga teng aylananing nuqtasi sifatida

$$x:=320+\text{trunc}(100\cdot\cos(-\pi/2)); y:=240+\text{trunc}(100\cdot\sin(-\pi/2));$$

kabi aniqlash mumkin. Muntazam oltiburchakning keyingi har bir nuqtasi shu formulada $-\pi/2$ ga, mos ravishda, $1\cdot(\pi/3)=\pi/3$, $2\cdot(\pi/3)=2\cdot\pi/3$, $3\cdot(\pi/3)=\pi$, $4\cdot(\pi/3)=4\cdot\pi/3$, $5\cdot(\pi/3)=5\cdot\pi/3$ ni qo'shish orqali hosil qilinadi.

Chiziq rangini tanlash uchun **SetColor(4)**, fon rangini tanlash uchun **SetBkColor(14)**, ichini siyrak qizil nuqtalar bilan to'ldirish uchun **SetFillStyle(10,4)**; protseduralaridan foydalaniladi. Agar bu nuqtalar **FillPoly(BS,KM)** protsedurasi yordamida tutashtirilsa, ichi o'zi bo'yaladi. Agar nuqtalarni **Line(X1,Y1,X2,Y2)** protsedurasi yordamida tutashtirilsa ichiga bo'yash usulini qo'llash uchun **Floodfill** protsedurasidan foydalaniladi.



Dasturi (Line):

Uses Graph;

Var gd, gm, X1,Y1, X2,Y2: integer;

X3,Y3,X4,Y4,X5,Y5,X6,Y6: integer;

Begin Gd:= 0; InitGraph(Gd, Gm, ""); Setbkcolor(14); Setcolor(4);

SetFillStyle(10,4); X1:=320+trunc(100*cos(-pi/2));

```

Y1:=240+trunc(100*sin(-pi/2)); X2:=320+trunc(100*cos(-pi/2+pi/3));
Y2:=240+trunc(100*sin(-pi/2+pi/3));
X3:=320+trunc(100*cos(-pi/2+2*pi/3));
Y3:=240+trunc(100*sin(-pi/2+2*pi/3));
X4:=320+trunc(100*cos(-pi/2+pi)); Y4:=240+trunc(100*sin(-pi/2+pi));
X5:=320+trunc(100*cos(-pi/2+4*pi/3));
Y5:=240+trunc(100*sin(-pi/2+4*pi/3));
X6:=320+trunc(100*cos(-pi/2+5*pi/3));
Y6:=240+trunc(100*sin(-pi/2+5*pi/3));
Line(X1,Y1,X2,Y2); Line(X2,Y2,X3,Y3); Line(X3,Y3,X4,Y4);
Line(X4,Y4,X5,Y5); Line(X5,Y5,X6,Y6); Line(X6,Y6,X1,Y1);
Floodfill(320,240,red); Readln; CloseGraph;

```

End.

Dasturi (DrawPoly):

Uses Graph;

Var gd, gm : Integer; km : Array[1..6,1..2] of Integer;

Begin Gd:= 0; InitGraph(Gd, Gm, ""); Setbkcolor(14); Setcolor(4);

SetFillStyle(10,4); km[1,1]:=320+trunc(100*cos(-pi/2));

km[1,2]:=240+trunc(100*sin(-pi/2));

km[2,1]:=320+trunc(100*cos(-pi/2+pi/3));

km[2,2]:=240+trunc(100*sin(-pi/2+pi/3));

km[3,1]:=320+trunc(100*cos(-pi/2+2*pi/3));

km[3,2]:=240+trunc(100*sin(-pi/2+2*pi/3));

km[4,1]:=320+trunc(100*cos(-pi/2+pi));

km[4,2]:=240+trunc(100*sin(-pi/2+pi));

km[5,1]:=320+trunc(100*cos(-pi/2+4*pi/3));

km[5,2]:=240+trunc(100*sin(-pi/2+4*pi/3));

km[6,1]:=320+trunc(100*cos(-pi/2+5*pi/3));

km[6,2]:=240+trunc(100*sin(-pi/2+5*pi/3));

FillPoly(6,km); Readln; CloseGraph;

End.

T-8*. 12 xil bo'yash usulini namoyish etuvchi 40x40 o'lchamli 12 ta kvadrat chizuvchi dastur tuzing.

Yechim:

Ekran foniga (Setbkcolor(13);) va chegarasiga (Setcolor(13);) pushti rang (kodi 13) tanlanib, ichi va bo'yash (Setfillstyle(n,n);) usulini **n** parametr sifatida qaraladi va parametrlilik takrorlash operatori yordamida o'zgartiriladi. To'g'ri to'rtburchaklar Bar(x,y, x+40, y+40); protsedurasi yordamida X va Y larni 60 pikselga o'zgartirish orqali chiziladi.

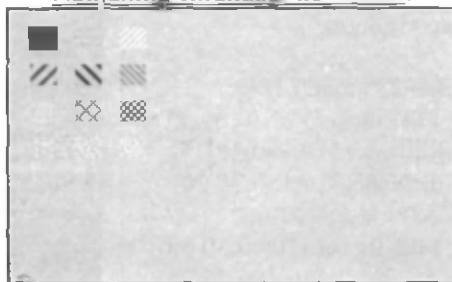
Dasturi:

```

Uses graph;
Var gd, gm, x, y: integer; n: byte;
Begin gd:=0; InitGraph(gd, gm, ""); Setbkcolor(13); Setcolor(13);
x:=60; y:=60;
For n:=1 to 3 do begin Setfillstyle(n,n); Bar(x, y, x+40, y+40); x:=x+60; end;
x:=60; y:=120;
For n:=4 to 6 do begin Setfillstyle(n,n); Bar(x, y, x+40, y+40); x:=x+60; end;
x:=60; y:=180;
For n:=7 to 9 do begin Setfillstyle(n,n); Bar(x, y, x+40, y+40); x:=x+60; end;
x:=60; y:=240;
For n:=10 to 12 do begin Setfillstyle(n,n); Bar(x,y,x+40,y+40);x:=x+60;end;
Readln; CloseGraph;
End.

```

Natijaning ekrandagi ko'rinishi:



T-9*. Svetofor chiroqlarini ketma-ket yondiradigan svetofor rasmini chizuvchi dastur tuzing.

Yechim:

Bu masalani quyidagich hal etish mumkin.

Har bir shakl uchun **SetFillStyle** protsedurasi yordamida rang turlicha tanlanadi, lekin bo'yash usuli kodi 1 (berilgan rang bilan bo'yash) deb olinadi: **SetFillStyle**(1, rang). Ekran foni chiroyli ko'rinishi uchun och siyohrang yoki pushti rang tanlanadi: **Setbkcolor**(13). Svetofor ramkasi uchun **Bar**(X1,Y1,X2,Y2) protsedurasi yordamida billur rangda to'g'ri to'rtburchak chiziladi.

Qizil, sariq va yashil rangli svetofor chiroqlari uchun **FillEllipse**(X,Y,XR,YR) protsedurasi yordamida (1 nishonli protsedura va operatorlar guruhi) 3 ta turli radiusli doira chizib (doira chizish uchun XR=YR bo'lishi kerak), birinchi doira ichidagi rang "yonib turgan", qolgan ikkitasini rangi "o'chib turgan" holga bo'yaladi. Xuddi shu doiralar nusxalanib ikkinchi (2 nishonli protsedura va operatorlar guruhi), keyin uchinchi (3 nishonli protsedura va operatorlar guruhi) doira ichidagi rang "yonib turgan", qolgan ikkita-

sini rangi "o'chib turgan" holga bo'yaladi.

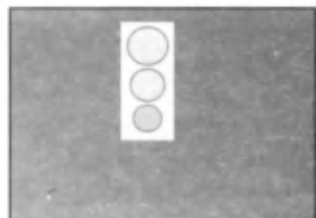
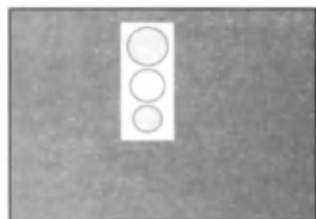
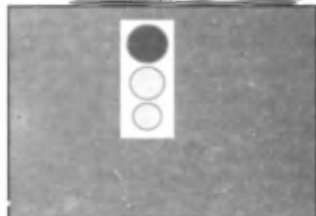
Svetofor yonib o'chishi qizil, sariq, yashil, sariq, qizil va shu kabi rangda bo'lganligi sababli yo'nalish tanlash uchun n o'zgaruvchi kiritiladi. Agar $n=1$ bo'lsa, yo'nalish qizildan yashilga (yuqoridan quyiga), $n=2$ bo'lsa yashildan qizilga (quyidan yuqoriga) tomon o'tiladi. Agar $n=3$ bo'lsa dastur o'z ishini yakunlaydi.

Dastur qismlarida vaqtni sekinlashtirish uchun **Crt** moduliga tegishli **Delay**(50000) protsedurasi parametrli takrorlash operatori bilan birgalikda foydalanildi. Dasturda ishni yakunlash uchun biror klavish bosilganini aniqlaydigan **Crt** moduliga tegishli **Keypress** mantiqiy funksiyasidan foydalanish qulay.

Dasturi:

```
Uses crt, graph;  
Label 1,2,3;  
Var gd, gm:integer; k, n:byte;  
Begin gd:=0; Initgraph (gd,gm,"");  
Setbkcolor(13);  
SetFillStyle(1,11); Bar(270,5,350,180);  
1: SetFillStyle(1,4); FillEllipse(310,40,30,30);  
SetFillStyle(1,7); FillEllipse(310,100,25,25);  
SetFillStyle(1,7); FillEllipse(310,150,20,20);  
n:=1; for k:=1 to 10 do delay(50000);  
2: SetFillStyle(1,7); FillEllipse(310,40,30,30);  
SetFillStyle(1,14); FillEllipse(310,100,25,25);  
SetFillStyle(1,7); FillEllipse(310,150,20,20);  
For k:=1 to 10 do delay(50000);  
If n=2 then goto 1;  
3: SetFillStyle(1,7); FillEllipse(310,40,30,30);  
SetFillStyle(1,7); FillEllipse(310,100,25,25);  
SetFillStyle(1,2); FillEllipse(310,150,20,20);  
For k:=1 to 10 do delay(50000);  
If keypressed then n:=3 else n:=2;  
If n=2 then goto 2 else CloseGraph;  
End.
```

Natijaning ekrandagi ko'rinishi:



44 – dars. Fayllar bilan ishlash mavzusiga

M-1. Hafta kunlarining nomlarini kiritib, ularni "HAFTA.TXT" faylida saqlab qo'yadigan dastur tuzing.

Yechim:

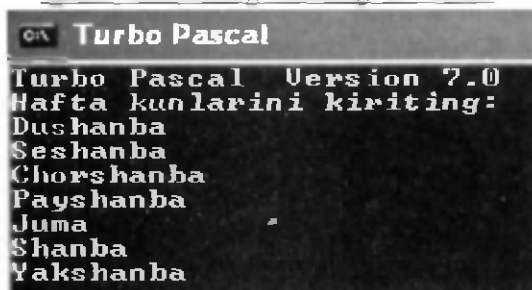
Fayl turidagi **f.** 10 ta belgili satr turidagi **kunlar** va butun turdagi **k**

o'zgaruvchilar tavsiflanadi. **Assign**(f, 'HAFTA.TXT'); protsedurasi fayl turidagi **f** o'zgaruvchi qiymati "HAFTA.TXT" ekanini belgilaydi. **Rewrite**(f); protsedurasi tashqi xotirada "HAFTA.TXT" faylini hosil qiladi va uni yozish uchun ochadi. **Writeln**('Hafta kunlarini kiriting: '); operatori ekranning joriy satrida "Hafta kunlarini kiriting:" yozuvini aks ettiradi va yurgichni yangi satrga o'tkazadi. **Writeln**(f, 'Hafta kunlari: '); operatori "HAFTA.TXT" faylining joriy satriga "Hafta kunlari:" yozuvini yozadi va yurgichni yangi satrga o'tkazadi. Parametrlil takrorlash operatori ishlaganda haftaning 7 ta kuni nomi **Readln**(kunlar); operatori orqali kiritiladi va **Writeln**(f, kunlar); operatori yordamida "HAFTA.TXT" fayliga yangi satrdan yozib boriladi.

Dasturi:

```
Var f: text; kunlar:string[10]; k: byte;
Begin Assign(f, 'HAFTA.TXT'); Rewrite(f);
Writeln('Hafta kunlarini kiriting: '); Writeln(f, 'Hafta kunlari: ');
For k:=1 to 7 do begin Readln(kunlar); Writeln(f, kunlar); end;
Close(f);
End.
```

Dastur ishlashining ekrandagi ko'rinishi:



"hafta.txt" faylining ekrandagi ko'rinishi:



M-2. 1-mashqdagi "HAFTA.TXT" faylini ochib, davomidan hafta kunlarining rus tilidagi nomlarini yozuvchi dastur tuzing.

Yechim:

Fayl turidagi **f** 11 ta belgili satr turidagi **kunRus** va butun turdagi **k** o'zgaruvchilar tavsiflanadi. **Assign**(f, 'HAFTA.TXT'); protsedurasi fayl turidagi **f** o'zgaruvchi qiymati "HAFTA.TXT" ekanini belgilaydi. Agar tashqi xotirada "HAFTA.TXT" fayli bo'lsa, u holda **Append**(f); protsedurasi bu faylni davomiga ma'lumot yozish uchun ochadi, aks holda xatolik haqida xabar chiqadi. Dasturning qolgan qismining ishi avvalgi mashqda ko'rilgan.

Dasturi:

```
Var f: text; kunRus: string[11]; k: byte;
Begin Assign(f, 'HAFTA.TXT');
Append(f); Writeln('Hafta kunlari nomini
```

"HAFTA.TXT" ning ekrandagi ko'rinishi:

```

rus tilida kiriting:');
Writeln(f, 'Hafta kunlari rus tilida: ');
For k:=1 to 7 do begin
Readln(kunRus); Writeln(f, kunRus); end;
Close(f);
End.

```

Dastur ishlashining ekrandagi ko'rinishi:

```

Turbo Pascal
Turbo Pascal Version 7.0 Copyright (c)
Hafta kunlari nomini rus tilida kiriting:
Ponedelnik
Vtornik
Sreda
Chetverg
Pyatnitsa
Subbota
Voskresenye_

```

```

HAFTA - Блокнот
Hafta kunlari:
Dushanba
Seshanba
Chorshanba
Payshanba
Juma
Shanba
Yakshanba
Hafta kunlari rus tilida:
Ponedelnik
Vtornik
Sreda
Chetverg
Pyatnitsa
Subbota
Voskresenye
Стр 1, 177

```

M-3. “HAFTA.TXT” faylida berilgan hafta kunlarining nomlarini ekranga chiqaruvchi dastur tuzing.

Yechim:

Fayl turidagi **f**, satr turidagi **kun** va butun turdagi **k** o'zgaruvchilar tavsiflanadi. Assign(f, 'HAFTA.TXT'); protsedurasi fayl turidagi **f** o'zgaruvchi qiymati “HAFTA.TXT” ekanini belgilaydi. Agar tashqi xotirada “HAFTA.TXT” fayli bo'lsa, u holda Reset(f); protsedurasi bu faylni ma'lumotlarini o'qish uchun ochadi, aks holda xatolik haqida xabar chiqadi. Readln(f, kun); operatori matnli “HAFTA.TXT” faylidagi 16 ta satrni navbati bilan **kun** o'zgaruvchisiga parametrlil takrorlash operatori yordamida o'qiydi.

Dasturi:

```

Var f: text; kun: string; k: byte;
Begin Assign(f, 'HAFTA.TXT'); Reset(f); Writeln('Hafta kunlari nomi: ');
For k:=1 to 16 do begin Readln(f, kun); Writeln(kun); end; Close(f);
End.

```

Natijaning ekrandagi ko'rinishi:

```

Turbo Pascal
Turbo Pascal Version 7.0
Hafta kunlari nomi:
Dushanba
Seshanba
Chorshanba
Payshanba
Juma
Shanba
Yakshanba
Hafta kunlari rus tilida:
Ponedelnik
Vtornik
Sreda
Chetverg
Pyatnitsa
Subbota
Voskresenye

```

45 – dars. Fayllar bilan ishlash mavzusini takrorlash mavzusiga

T-1. Sinfdozlaringizning familiya va ismlaridan tashkil topgan “SINF.TXT” nomli matnli fayl hosil qiluvchi dastur tuzing.

Yechim:

Fayl turidagi **f**, satr turidagi **Fam_ism** va butun turdagi **k** va **n** o‘zgaruvchilar tavsiflanadi. Assign(f, ‘SINF.TXT’); protsedurasi fayl turidagi **f** o‘zgaruvchi qiymati “SINF.TXT” ekanini belgilaydi. **Rewrite(f)**; protsedurasi tashqi xotirada “SINF.TXT” faylini hosil qiladi va uni yozish uchun ochadi. O‘zgaruvchi **n** o‘quvchilar sonini aniqlaydi, unga asosan parametrli takrorlash operatori Readln(Fam_ism); va Writeln(f, Fam_ism); operatorlarini takrorlashlar sonini belgilaydi. O‘quvchining familiyasi va ismi Readln(Fam_ism); operatori orqali kiritiladi hamda Writeln(f, Fam_ism); operatori yordamida “SINF.TXT” fayliga yangi satrdan yozib boriladi.

Dasturi:

Var

f: text; Fam_ism: string; k, n: byte;

Begin

Write(‘Sinfdozlaringiz sonini kiriting: ‘); Readln(n);

Assign(f, ‘SINF.TXT’); Rewrite (f);

Writeln(‘Sinfdozingiz familiyasi va ismini kiriting:‘);

For k:=1 to n do begin

Readln(Fam_ism); Writeln(f, Fam_ism); end;

Close(f);

End.

Dastur ishlashining ekrandagi ko‘rinishi:

```
Turbo Pascal
Turbo Pascal Version 7.0 Copyright (c) 198
Sinfdozlaringiz sonini kiriting: 5
Sinfdozingiz familiyasi va ismini kiriting:
Abdullayev Mirkomil
Begmatov Abduvali
Mahkamov Alisher
Mirobidov Mirkamol
Xurramov Bahodir
```

“SINF.TXT” ni ekrandagi ko‘rinishi:

```
SINF - Блокнот
Abdullayev Mirkomil
Mahkamov Alisher
Mirobidov Mirkamol
Xurramov Bahodir
Стр 1
```

T-2. “sinf.txt” faylida berilgan 9-sinf o‘quvchilarning familiyalari ichidan “M” harfi bilan boshlanadiganlarini ekranga chiqaruvchi dastur tuzing.

Yechim:

Fayl turidagi **f**, satr turidagi **Fam** va butun turdagi **k** va **n** o'zgaruvchilar tavsiflanadi. Assign(f, 'sinf.txt'); protsedurasi fayl turidagi **f** o'zgaruvchi qiymati "sinf.txt" ekanini belgilaydi. Agar tashqi xotirada "sinf.txt" fayli bo'lsa, u holda **Reset(f)**; protsedurasi bu faylni o'qish uchun ochadi, aks holda xatolik haqida xabar chiqadi.

Tashqi xotiradagi "sinf.txt" fayli necha satrdan iboratligi no'malum. Shuning uchun Paskalning **Eof(f)** funksiyasi shart bo'yicha takrorlash operatori bilan birga qo'llaniladi: **Eof** – mantiqiy funksiya bo'lib, faylda o'qish uchun ma'lumotlar qolmagan bo'lsa "Rost", aks holda "Yolg'on" qiymatini qabul qiladi. Readln(f, fam); operatori matnli "sinf.txt" faylidagi satrlarni navbati bilan **Fam** o'zgaruvchisiga o'qiydi. Tarmoqlanish operatori **Fam** o'zgaruvchisi qiymatining 1-belgisi "M" ga teng bo'lsa, ekranga chiqaradi.

Dasturi:

Var f: text; Fam: string;

Begin

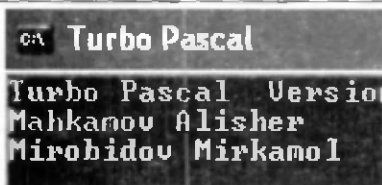
Assign(f, 'sinf.txt'); Reset(f);

Repeat ReadLn(f, fam); If Fam[1]='M' Then WriteLn(fam); Until Eof(f);

Close(f); Readln;

End.

Dastur ishlashining ekrandagi ko'rinishi:



```

Turbo Pascal
Turbo Pascal Version
Mahkanov Alisher
Mirobidov Mirkamol

```

T-3. "sinf.txt" faylida berilgan 9-sinf o'quvchilarning familiyalari ichidan "B" harfi bilan hoshlanadiganlarini ajratib olib, ulardan "bsinf.txt" faylini hosil qiluvchi dastur tuzing.

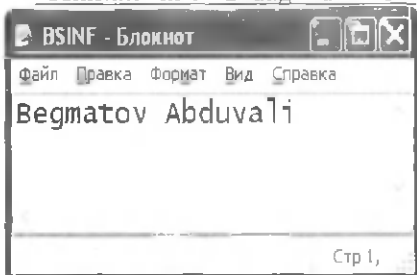
Yechim:

Fayl turidagi **f** va **fb**, satr turidagi **Fam** o'zgaruvchilar tavsiflanadi. Assign(f, 'sinf.txt'); protsedurasi fayl turidagi **f** o'zgaruvchi qiymati "sinf.txt" ekanini belgilaydi. Agar tashqi xotirada "sinf.txt" fayli bo'lsa, u holda **Reset(f)**; protsedurasi bu faylni o'qish uchun ochadi, aks holda xatolik haqida xabar chiqadi. Assign(fb, 'bsinf.txt'); protsedurasi fayl turidagi **fb** o'zgaruvchi qiymati "bsinf.txt" ekanini belgilaydi. **Rewrite(f)**; protsedurasi "bsinf.txt" faylini hosil qiladi va yozish uchun ochadi. Tarmoqlanish operatori **Fam** o'zgaruvchisi qiymatining 1-belgisi "B" ga teng bo'lsa, WriteLn(fb, Fam); yordamida "bsinf.txt" fayliga yozadi.

Dasturi:

```
Var f, fb: text; Fam: string;  
Begin Assign(f, 'sinf.txt'); Reset(f);  
Assign(fb, 'bsinf.txt'); Rewrite(fb);  
Repeat ReadLn(f, Fam);  
If Fam[1]= 'B' Then  
WriteLn(fb, Fam);  
Until Eof(f);  
Close(f); Close(fb); ReadLn;  
End.
```

"bsinf.txt" ni ekrandagi ko'rinishi:



T-4*. $y = \sin^2 x$ funksiyasining $[-\pi, \pi]$ oraligqdagi qiymatlarini 0,01 qadam bilan hisoblang. Natijalarni "sinus.out" faylida saqlab qo'ying.

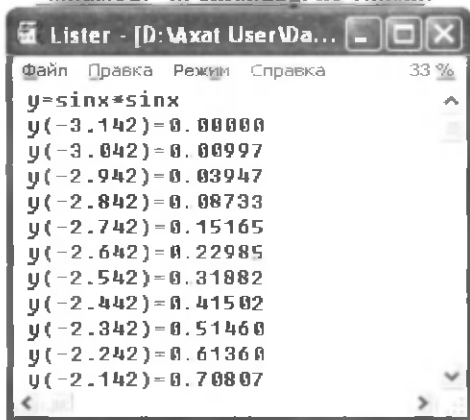
Yechim:

Fayl turidagi s , haqiqiy turdagi x, y o'zgaruvchilar tavsiflanadi. Assign(s, 'sinus.out'); protsedurasi fayl turidagi s o'zgaruvchi qiymati "sinus.out" ekanini belgilaydi. Rewrite(s); protsedurasi "sinus.out" faylini hosil qiladi va yozish uchun ochadi. Shart bo'yicha takrorlash operatori yordamida x o'zgaruvchi qiymatini 0,01 qadam bilan o'zgartirib, $y = \sin^2 x$ funksiyasi qiymati hisoblanadi va WriteLn(s, 'y('x, ')=' , y:0:5); yordamida "sinus.out" fayliga yoziladi.

Dasturi:

```
Var s: text; x, y: real;  
Begin  
Assign(s, 'sinus.out');  
Rewrite(s);  
WriteLn(s, 'y=sinx*sinx');  
x:= -pi;  
Repeat  
y:=sqr(sin(x));  
WriteLn(s, 'y('x, ')=' , y:0:5);  
x:=x+0.1;  
Until x>pi;  
Close(s);  
End.
```

"sinus.out" ni ekrandagi ko'rinishi:



T-5*. "sinus.out" fayliga izoh qo'shuvchi dastur tuzing.

Yechim:

Fayl turidagi s , haqiqiy turdagi x, y o'zgaruvchilar tavsiflanadi. Assign(s, 'sinus.out'); protsedurasi fayl turidagi s o'zgaruvchi qiymati "sinus.out" eka-

nini belgilaydi. Agar tashqi xotirada "sinus.out" fayli bo'lsa, u holda **Append(f)**; protsedurasi bu faylni davomiga ma'lumot yozish uchun ochadi, aks holda xatolik haqida xabar chiqadi. **WriteLn(s, 'funksiyani qiymati 0.01 qadam bilan hisoblandi')**; yordamida "sinus.out" fayliga izoh qo'shiladi

Dasturi:

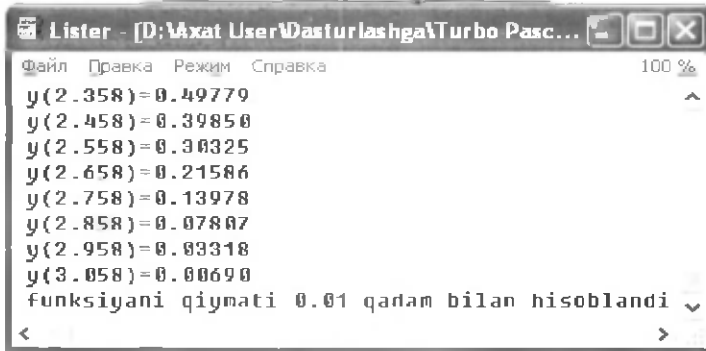
Var s: text;

Begin Assign(s, 'sinus.out'); Append(s);

WriteLn(s, 'funksiyani qiymati 0.01 qadam bilan hisoblandi'); Close(s);

End.

"sinus.out" ni ekrandagi ko'rinishi:



46 – dars. Protsedura va funksiyalar mavzusiga

M-1. Uchta to'g'ri to'rtburchakning har birining bittadan diagonali uchlarining koordinatalari berilgan: 1) 20,20 va 80,200; 2) 200,97 va 500,156; 3) 300,120 va 400,420. Shu to'g'ri to'rtburchaklarni mos ravishda qizil, sariq va yashil ranglarda chizish dasturini tuzing.

Yechim:

Berilgan qiymatlarda to'g'ri to'rtburchaklarni qizil, sariq va yashil ranglarda chizish uchun uch marta chegara rangini belgilovchi Paskalning Setcolor(rang); standart protsedurasidan foydalanishga to'g'ri keladi. Masalada berilgan qiymatlarni va rangni parametr sifatida olib, shu parametrlar ishtirokida **Tbch** foydalanuvchi protsedurasidan foydalanib ham yechish mumkin.

Protsedurali dasturda gd va gm global, x1, y1, x2, y2 va rang lokal o'zgaruvchilardir.

Dasturi (oddiy):

Uses Graph;

Dasturi (protsedurali):

Uses Graph;

<pre> Var gd, gm: Integer; Begin gd:=0; InitGraph(gd, gm, ''); Setcolor(4); Rectangle(20,20,80,200); Setcolor(14); Rectangle(200,97,500,156); Setcolor(2); Rectangle(20,20,80,200); ReadLn; CloseGraph; End.</pre>	<pre> Var gd, gm: Integer; Procedure Tbch(x1,y1,x2,y2,rang:Integer); begin Setcolor(rang); Rectangle(x1,y1,x2,y2); end; Begin gd:=0; InitGraph(gd, gm, ''); Tbch (20,20,80,200,4); Tbch (200,97,500,156,14); Tbch (300,120,400,420,2); ReadLn; CloseGraph; End.</pre>
--	---

M-2. Berilgan n natural son uchun $S=1\cdot 5+2\cdot 6+3\cdot 7+\dots+n\cdot(n+4)$ yig'indini hisoblash dasturini protsedura yordamida tuzing.

Yechim:

Yig'indidagi ko'paytmalarni hisoblash uchun foydalanuvchi protsedurasini tuzish mumkin. Protsedura parametri sifatida ko'paytuvchilarni olish mumkin. Protsedurali dasturda k , n , S va P gm global, $k1$ va $k2$ lokal o'zgaruvchilardir.

Dasturi (oddiy):

```

Var k, n, S, P: word;
Begin
Write('n ni kiriting: '); ReadLn(n);
S:=0; k:=1;
While k<=n do begin
S:=S+k*(k+4); k:=k+1; end;
Writeln('Javob S= ', S);
ReadLn;
End.
```

Dasturi (protsedurali):

```

Var k, n, S, P: word;
Procedure Kup(k1, k2:word);
Begin P:=k1*k2; end;
BEGIN Write('n ni kiriting: ');
ReadLn(n); S:=0; k:=1;
While k<=n do begin Kup(k, k+4);
S:=S+P; k:=k+1; end;
Writeln('Javob S= ', S); ReadLn;
END.
```

M-3. Uchta sondan kattasini topish dasturini tuzing. Buning uchun ikkita sondan kattasini topish funksiyasini tuzib, undan foydalaning.

Yechim:

Ikkita sondan kattasini topish uchun foydalanuvchi protsedurasini tuzish mumkin. Protsedurali dasturda A , B , D va Max global, n va m lokal o'zgaruvchilardir.

Dasturi (oddiy):

```

Var A, B, D, Max: real;
Begin Write('A ni kiriting: '); ReadLn(A); Write('B ni kiriting: '); ReadLn(B);
Write('D ni kiriting: '); ReadLn(D); If A>B Then Max:=A Else Max:=B;
If D>Max Then Max:=D; Writeln('Sondan kattasi= ', Max:0:5); ReadLn;
```

End.

Dasturi (protsedurali):

Var A, B, D, Max: real;

Function Katta(n, m:real):real;

Begin

If n>m Then Katta:=n Else Katta:=m;

End;

BEGIN Write('A ni kiriting: '); Readln(A); Write('B ni kiriting: '); Readln(B);

Write('D ni kiriting: '); Readln(D); Max:=Katta(A,B); Max:=Katta(Max,D);

Writeln('Sonlardan kattasi= ', Max:0:5); ReadLn;

END.

47 – dars. Protседura va funkşiyalar mavzusini takrorlash mavzusiga

T-1. Berilgan matndagi 'a' belgini 'g' belgiga, 'm' belgini 's' belgiga, 'f' belgini 'h' belgiga almashtiruvchi dasturni protsedura yordamida tuzing.

Yechim:

Satrdan birinchi belgini ikkinchi belgiga almashtirish uchun avval belgi o'rnini aniqlanib o'chiriladi, so'ng yangi belgi o'rniga joylashtiriladi. Bu jarayon birinchi belgi tugaguncha davom etadi, shuning uchun shart bo'yicha takrorlash operatoridan foydalanish qulay.

Dasturi (oddiy):

Var S: string; k: char;

Begin Write('S ni kiriting: '); Readln(S);

While Pos('a', S)>0 do begin

k:= Pos('a', S);

Delete(S, k, 1); Insert('b', S, k); end;

While Pos('m', S)>0 do begin

k:= Pos('m', S);

Delete(S, k, 1); Insert('s', S, k); end;

While Pos('f', S)>0 do begin

k:= Pos('f', S);

Delete(S, k, 1); Insert('h', S, k); end;

Writeln('Javob S= ', S); ReadLn;

End.

Dasturi (protsedurali):

Var S: string;

Procedure Alm(n, m: char);

Var k: byte;

Begin

While Pos(n, S)>0 do begin

k:= Pos(n, S); Delete(S, k, 1);

Insert(m, S, k); end;

End;

BEGIN

Write('Satrni kiriting: '); Readln(S);

Alm('a','b'); Alm('m','s'); Alm('f','h');

Writeln('Javob S= ', S); ReadLn;

END.

T-2. $y = x^5 + 3x$ funkşiya qiymatini x ning $-9, -5, -2, 2, 5, 7$ qiy-

matlarida hisoblash dasturini tuzing. Darajani ko'paytirish orqali hisoblash uchun funktsiya tuzib oling.

Yechim:

Darajani ko'paytirish yordamida hisoblash uchun **Dar** funksiyasini tuzganda avval daraja **d** o'zgaruvchi orqali hisoblab olinadi, so'ng funktsiyaga o'zlashtiriladi. Umuman, biror foydalanuvchi funktsiyasi o'zlashtirish operatoridagi tenglikni chap tomonida qatnashsa (masalan, Dar:=) uning nomi, o'ng tomonida qatnashsa funktsiyaga murojaati (masalan, y:= Dar(x)+5*Dar(x)+8) yozilishi mumkin.

Dasturi (oddiy):

```
Var
  x, y, d: real; k: byte;
BEGIN
  Write('x ni kiriting: ');
  Readln(x);
  d:=1;
  For k:=1 to 5 do d:=d*x;
  y:=d+3*x;
  Writeln('y('x:0:2, ')='y:0:5);
  ReadLn;
END.
```

Dasturi (protsedurali):

```
Var x, y: real;
Function Dar(z: real): real;
Var k: byte; d:real;
Begin
  d:=1; For k:=1 to 5 do d:=d*z; Dar:=d;
End;
BEGIN
  Write('x ni kiriting: '); Readln(x);
  y:= Dar(x)+3*x;
  Writeln('y('x:0:2, ')='y:0:5); ReadLn;
END.
```

T-3. Protsedura yordamida ekranning o'rtasida ichma-ich joylashgan har xil rangli 15 ta aylana chizuvchi dastur tuzing.

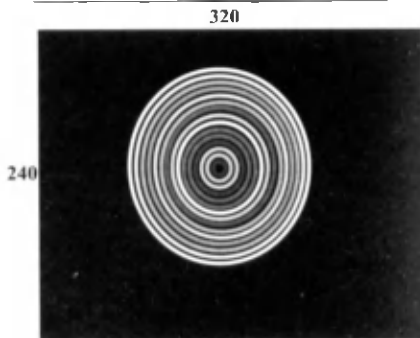
Yechim:

Protsedura parametrlari sifatida aylanalarning radiusi va rangini olish mumkin. Bu holda protsedura tanasida aylana chizig'i rangini aniqlovchi va aylana chizuvchi standart protsedura qatnashadi.

Dasturi:

```
Uses Graph;
Var n, rang, k, gd, gm: Integer;
Procedure aylana(r, rang: Integer);
begin
  Setcolor(rang); Circle(320,225, r);
end;
Begin
  gd:=0; InitGraph(gd, gm, '');
  For k:=1 to 15 do begin
    n:=k*15; Aylana(n,k); end;
  ReadLn; CloseGraph;
End.
```

Natijaning ekrandagi ko'rinishi:

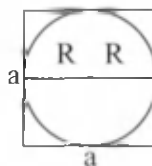


48 – 49 – dars. Takrorlashga doir topshiriqlar mavzusiga

T-1. Tomonining uzunligi a ga teng kvadrat va unga ichki chizilgan aylana chizish dasturi tuzilsin. a ning qiymati klaviaturadan kiritilsin.

Yechim:

Chizmaga asosan $a=2 \cdot R$ yoki $R=\frac{a}{2}$. Kvadratning chap yuqori qirrası koordinatasi (x, y) bo'lsa, u holda quyi o'ng qirrası $(x+a, y+a)$ bo'ladi. Demak, aylana markazi koordinatalari kesmani teng ikkiga bo'lish qoidasiga asosan quyidagicha bo'ladi:



$$\left(\frac{x+x+a}{2}, \frac{y+y+a}{2} \right) = \left(\frac{2 \cdot x+a}{2}, \frac{2 \cdot y+a}{2} \right) = \left(x+\frac{a}{2}, y+\frac{a}{2} \right).$$

Masalan, agar $a=100$, $x=100$ va $y=80$ deb olinsa, aylana markazi koordinatalari $(150, 130)$ kabi bo'ladi. Dasturda a o'zgeruvchini qiymatini klaviaturadan kiritish uchun `ReadLn(a)` operatori yoziladi. Shakllar chizish protseduralari piksellarning butun qiymatlarida ishlagani uchun a o'rniga `trunc(a)` va R o'rniga `trunc(R)` qaraladi.

Dasturi:

Uses Graph;

Var x, y, gd, gm : Integer; a, R : real;

Begin $gd:=0$; `InitGraph(gd, gm, '')`; `Write('a=')`; `ReadLn(a)`; $R:=a/2$;

`WriteLn('Kvadrat chap yuqori qirrası koordinatalarini kiriting:')`;

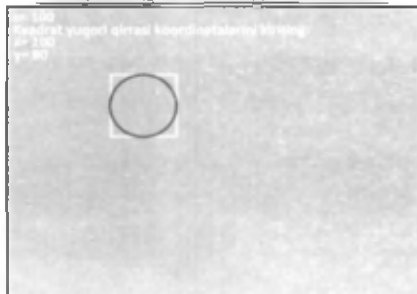
`Write('x=')`; `ReadLn(x)`; `Write('y=')`; `ReadLn(y)`; `Setbkcolor(2)`;

`SetColor(14)`; `Rectangle(x, y, x+trunc(a), y+trunc(a))`; `SetColor(4)`;

`Circle(x+trunc(a)/2, y+trunc(a)/2, trunc(R))`; `ReadLn`; `CloseGraph`;

End.

Natijaning ekrandagi ko'rinishi:



T-2. Radiusi R ga teng aylana va unga tashqi chizilgan kvadrat chizish dasturi tuzilsin. R ning qiymati klaviaturadan kiritilsin.

Yechim:

Avvalgi masaladagi chizmaga asosan $a=2\cdot R$. Aylana markazi koordinatasi (x, y) bo'lsa, u holda kvadratning chap yuqori qirrasini koordinatasi $(x-R, y-R)$ bo'lsa, u holda quyi o'ng qirrasini $(x+R, y+R)$ bo'ladi. Dasturda R o'zgaruvchisini qiymatini klaviaturadan kiritish uchun `ReadLn(R)` operatori yoziladi. Shakllar chizish protseduralari piksellarning butun qiymatlarida ishlagani uchun a o'rniga `trunc(a)` va R o'rniga `trunc(R)` qaraladi.

Dasturi:

Uses Graph;

Var x, y, gd, gm : Integer; a, R : real;

Begin `gd:=0; InitGraph(gd, gm, '');`

`Write('R= '); ReadLn(R); a:=2*R;`

`WriteLn('Aylana markazi koordinatasini kiriting:');`

`Write('x= '); ReadLn(x); Write('y= '); ReadLn(y);`

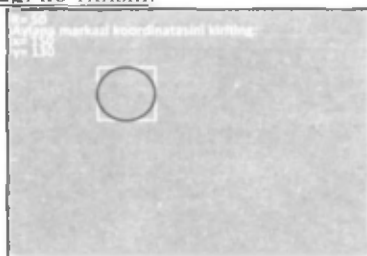
`Setbkcolor(2); Setcolor(14);`

`Rectangle(x-trunc(R), y-trunc(R), x+trunc(R), y+trunc(R));`

`Setcolor(4); Circle(x, y, trunc(R)); ReadLn; CloseGraph;`

End.

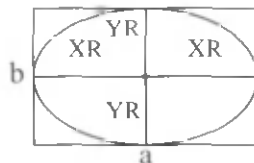
Natijaning ekrandagi ko'rinishi:



T-3. Tomonlarining uzunligi a va b bo'lgan to'g'ri to'rtburchak va unga ichki chizilgan ellips chizish dasturi tuzilsin. a va b ning qiymati klaviaturadan kiritilsin.

Yechim:

Ma'lumki, `FillEllipse(X,Y,XR,YR)` protsedurasi markazi (X,Y) koordinatali nuqtada, X va Y o'qlari bo'yicha radiuslari (eni va bo'yi) mos ravishda XR va YR ga teng, joriy rang va usulda bo'yalgan ellips chizadi. Chizmaga asosan



$a=2 \cdot XR$, $b=2 \cdot YR$ yoki $XR=\frac{a}{2}$, $YR=\frac{b}{2}$. Kvadratning chap yuqori qirrasini

koordinatasi (x, y) bo'lsa, u holda quyidagining qirrasini $(x+a, y+b)$ bo'ladi. Demak, ellips markazini koordinatalari kesmani teng ikkiga bo'lish qoidasiga asosan quyidagicha bo'ladi:

$$\left(\frac{x+x+a}{2}, \frac{y+y+b}{2} \right) = \left(\frac{2 \cdot x+a}{2}, \frac{2 \cdot y+b}{2} \right) = \left(x+\frac{a}{2}, y+\frac{b}{2} \right).$$

Dasturda a o'zgaruvchini qiymatini klaviaturadan kiritish uchun `ReadLn(a)` operatori yoziladi. Shakllar chizish protseduralari piksellarning butun qiymatlarida ishlagani uchun a o'rniga `trunc(a)`, b o'rniga `trunc(b)`, XR o'rniga `trunc(XR)` va YR o'rniga `trunc(YR)` qaraladi.

Dasturi:

Uses Graph;

Var x, y, gd, gm: Integer; a,b, XR, YR: real;

Begin gd:=0; InitGraph(gd, gm, "");

Write('a= '); ReadLn(a); XR:=a/2; Write('b= '); ReadLn(b); YR:=b/2;

WriteLn('To'rburchakni chap yuqori qirrasini koordinatalarini kiriting:');

Write('x= '); ReadLn(x); Write('y= '); ReadLn(y);

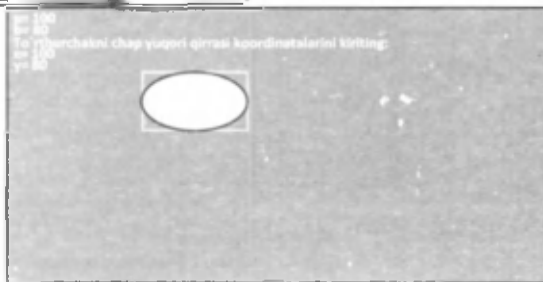
Setbkcolor(2); Setcolor(14); Rectangle(x, y, x+trunc(a), y+trunc(b));

Setcolor(4); Fillellipse(x+trunc(a/2), y+trunc(b/2), trunc(XR), trunc(YR));

ReadLn; CloseGraph;

End.

Natijaning ekrandagi ko'rinishi:



T-4. Ko'k fonli ekranni oralig'i 20 piksel bo'lgan vertikal oq chiziqlar bilan to'ldiring.

Yechim:

Ekran foni ko'k bo'lishi uchun `SetbkColor(blue)`, chiziq rangini belgilash uchun `SetColor(white)` tanlanadi. Ekran to'lguncha vertikal oq chiziqlar chizilishi uchun `Line(k, 0, k, 479)` protsedurasi shart bo'yicha takrorlash operatori yordamida k ni qiymati 0 dan 639 bo'lguncha 20 qadam bilan ishlatiladi.

Dasturi:

```
Uses Graph;  
Var k, gd, gm: Integer;  
Begin gd:=0;  
InitGraph(gd, gm, "");  
Setbkcolor(blue);  
Setcolor(white); k:=0;  
While k<=639 do begin  
Line(k, 0, k, 479); k:=k+20; end;  
ReadLn; CloseGraph;  
End.
```

Natijaning ekrandagi ko'rinishi:



T-5. Har xil rangli 15 ta doira chizing.

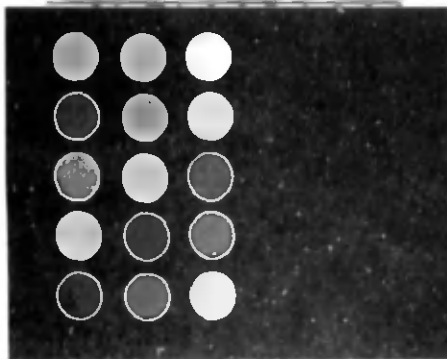
Yechim:

Doiralarga nar xil rang tanlash uchun m o'zgaruvchi kiritiladi. Har ta doirani markazini 100 dan boshlab 100 pikselga gorizontal bo'yicha surish uchun $x:=100*((m-1) \text{ div } 5+1)$ formula o'rinli. Bir vertikalda joylashgan 5 ta doirani vertikal bo'yicha 80 pikselga surish uchun $y:=((m \text{ mod } 5)+1)*80$ formula o'rinli.

Dasturi:

```
Uses Graph;  
Var m, x, y, gd, gm: Integer;  
Procedure doira(a, b, rang: Integer);  
Begin Setfillstyle(1, rang); Fillellipse(a, b, 35, 35); end;  
Begin gd:=0; InitGraph(gd, gm, "");  
for m:=1 to 15 do begin x:=100*((m-1) div 5+1); y:=((m mod 5)+1)*80;  
doira(x, y, m); end; ReadLn; CloseGraph;  
End.
```

Natijaning ekrandagi ko'rinishi:



T-6. Stol rasmini chizing

Yechim:

Stol oyoqlari va stol usti qoplamasini hajmli chizish uchun **Bar3D(...)** protsedurasidan foydalanish qulay.

Dasturi:

Uses Graph;

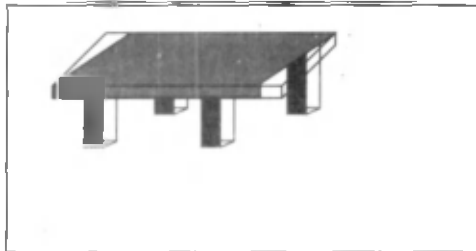
Var gd, gm: integer;

Begin

```
gd:=0; InitGraph(gd, gm, ""); setbkcolor(15); setcolor(1);  
setfillstyle(1, 1); bar3d(100, 130, 115, 220, 10, true);  
bar3d(150, 90, 165, 180, 10, true); bar3d(287, 130, 302, 220, 10, true);  
bar3d(327, 90, 342, 180, 10, true); setcolor(13); setfillstyle(1, 13);  
bar3d(80, 120, 320, 140, 60, true); floodfill(200, 80, 13); setcolor(4);  
bar3d(80, 120, 320, 140, 60, true); ReadLn; CloseGraph;
```

End.

Natijaning ekrandagi ko'rinishi:



T-7. Uy rasmini chizing

Dasturi:

Uses Graph;

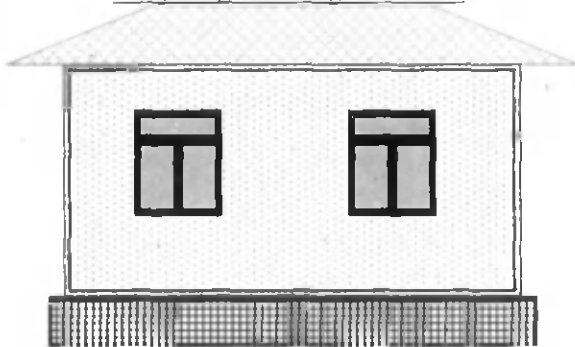
Var gd, gm: integer; m: array[1..4, 1..2] of integer;

Begin

```
gd:=0; InitGraph(gd, gm, ""); setbkcolor(15); setcolor(3);  
m[1,1]:=240; m[1,2]:=10; m[2,1]:=400; m[2,2]:=10;  
m[3,1]:=500; m[3,2]:=60; m[4,1]:=140; m[4,2]:=60;  
setfillstyle(8, 3); Fillpoly(4, m); setfillstyle(1, 7); bar(170, 61, 470, 250);  
setcolor(7); rectangle(170, 61, 470, 250); setfillstyle(1, 7);  
bar(175, 65, 465, 245); setfillstyle(7, 12); bar(160, 254, 480, 300);  
setcolor(12); rectangle(160, 252, 480, 300); setfillstyle(1, 4);  
bar(220, 100, 272, 180); bar(360, 100, 412, 180); setfillstyle(1, 7);  
bar(224, 104, 268, 176); bar(364, 104, 408, 176); setfillstyle(1, 4);  
bar(220, 120, 272, 130); bar(360, 120, 412, 130); bar(244, 120, 248, 176);  
bar(384, 120, 388, 176);
```

ReadLn; CloseGraph;
End.

Natijaning ekrandagi ko'rinishi:



УДК: 004.421
КБК 22.12
А45

Algoritmlash va Paskal dasturlash tili bo'yicha Berilgan misol va masalalarni yechish usullari / B. Boltayev [va boshq.]. – Toshkent: Nihol, 2012. – 164 б.

ISBN 978-9943-23-057-6

УДК: 004.421
КБК 22.12

**B.BOLTAYEV, A.AZAMATOV, SH.XIDIROV,
B.XURRAMOV VA K.ISKANDAROV**

**ALGORITMLASH VA PASKAL DASTURLASH TILI
BO'YICHA BERILGAN MISOL VA
MASALALARNI YECHISH
USULLARI**

Muharrir: A.Usmonov

Muqova dizayneri: *A. Ladina*

Texnik muharrir: *B. Nurmuhamedov*

Litsenziya raqami: AI № 129, sanasi: 12.11.2008 y.

Bosishga ruxsat etilgan sana: 14.12.2011. Ofset bosma.

Bichimi 60x84¹/₁₆. Bosma tabog'i 10,25. Shartli bosma tabog'i 9,5.

Ofset qog'oz. Times New Roman garniturası.

Kegl 11. Adadi: 3000 nusxa.

Buyurtma: № 131. Bahosi kelishilgan narxda.

«Ofset print» MChJ bosmaxonasida bosildi.

Bosmaxona manzili:

Toshkent shahar, Shayxontohur tumani, Maxtumquli 1,uy