

**O‘ZBEKISTON RESPUBLIKASI OLIY VA O‘RTA MAXSUS TA’LIM
VAZIRLIGI**

QARSHI MUHANDISLIK IQTISODIYOT INSTITUTI

Avtomatika va axborot texnologiyalari kafedrası

**“Informatika va axborot texnologiyalari”
fanidan**

REFERAT

Topshirdi: _____ Muxandis texnika fakulteti “Gidrotexnika inshootlari va nasos stansiyalaridan foydalanish” ta’lim yo‘nalishi 1-kurs talabasi Yusupov Serali

Qabul qildi: _____ Avtomatika va axborot texnologiyalari kafedrası o‘qituvchisi

Qarshi-2014

Mavzu: Vizual basic sikl operatorlari bilan ishlash

Reja:

1.Sikli strukturalar

1.1.Takrorlanishlar soni berilgan sikl (hisoblagichli sikl) For ... Next

2. Sharti oldinda va oxirida berilgan sikllar

2.1. Sharti oldinda berilgan Do While siklining sintaksisi

2.2. Sharti oxirida berilgan sikl

2.3. Sharti oldinda berilgan sikl

3. Foydalangan adabiyotlar ro'yxati

1. Siklli strukturalar

Dasturlash tillarida takrorlanuvchi jarayonlarni bajarish uchun siklli strukturalar ishlatiladi. Siklli strukturalarning uch xili mavjud:

1. Takrorlanishlar soni berilgan sikl (hisoblagichli sikllar)
2. Berilgan shart bajarilguncha bajariladigan sikl. Bunda shart sikl tanasi bajarilishidan oldin tekshiriladi (Yuqoridagi qator bilan boshqariladigan sikl)
3. Berilgan shart bajarilguncha bajariladigan sikl. Bunda shart sikl tanasi bajarilgandan so'ng tekshiriladi (Pastdagi qator bilan boshqariladigan sikl)

Sikl tanasi bo'yicha bir marta o'tish "iteratsiya" deyiladi.

1.1. Takrorlanishlar soni berilgan sikl (hisoblagichli sikl) For ... Next

For...Next strukturalari buyruqlar ketma-ketligini bir necha marta bajaradi. Bunaqangi konstrusiyani sikl deb ataladi va u yordamida bajariladigan dastur kodlarini – sikl tanasi deb ataladi..

For.. .Next strukturasining sintaksisi:

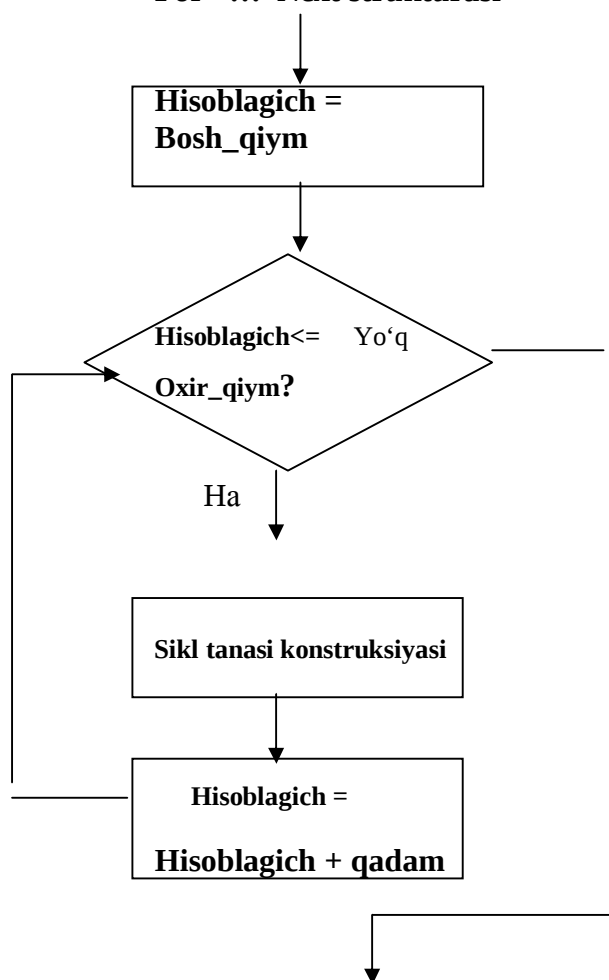
**For Hisoblagich = Boshlang'ich_qiymat To Oxirg'i_qiymat Step Qadam
konstruksiya**

Next Hisoblagich

Strukturaning birinchi argumenti - **Hisoblagich** siklning bajarilishlari sonini hisoblaydigan o'zgaruvchi nomini aniqlaydi. **Boshlang'ich_qiymat** parametri siklga birinchi marta kirishdan oldin **Hisoblagich** ga o'zlashtiriladigan boshlang'ich sonli qiymat bo'ladi. Sikl **Hisoblagich** ning qiymati **To** xizmatchi so'zidan keyingi **Oxirg'i_qiymat** ning qiymatidan oshib ketguncha bajariladi. Siklning har bir bajarilishidan so'ng **Hisoblagich** ning qiymati **Qadam** qiymatiga oshirilib boriladi. **Next** xizmatchi so'zi siklning oxirini ko'rsatadi. Siklning har bir bajarilishida **Hisoblagich** ning qiymati bilan **Oxirg'i_qiymat** argumentining qiymati solishtiriladi. Agar hisoblagichning qiymati **Oxirg'i_qiymat** ning o'rnatilgan qiymatidan oshib ketmasa, sikl tanasining konstruksiyasi bajariladi. Aks holda boshqarish **Next** dan keyingi operatorga o'tadi. **Hisoblagich** ning qiymatini o'zgartiruvchi **Qadam** ning qiymati manfiy ham bo'lishi mumkin. Bunday holatda sikl **Hisoblagich** ning qiymati **Oxirg'i_qiymat** ning qiymatidan kichik bo'lguncha bajariladi va hisoblagichning boshlang'ich qiymati oxirgi qiymatidan katta bo'lishi lozim. **Step** xizmatchi so'zini ishlatmasa ham bo'ladi. Bunday holatda esa qadamning qiymati avtomat ravishda 1 ga teng bo'ladi. Hisoblagich, Boshlang'ich_qiymat, Oxirg'i_qiymat, Qadam parametrlari – o'zgaruvchi, doimiy yoki butun turdagi ifoda bo'lishi mumkin.

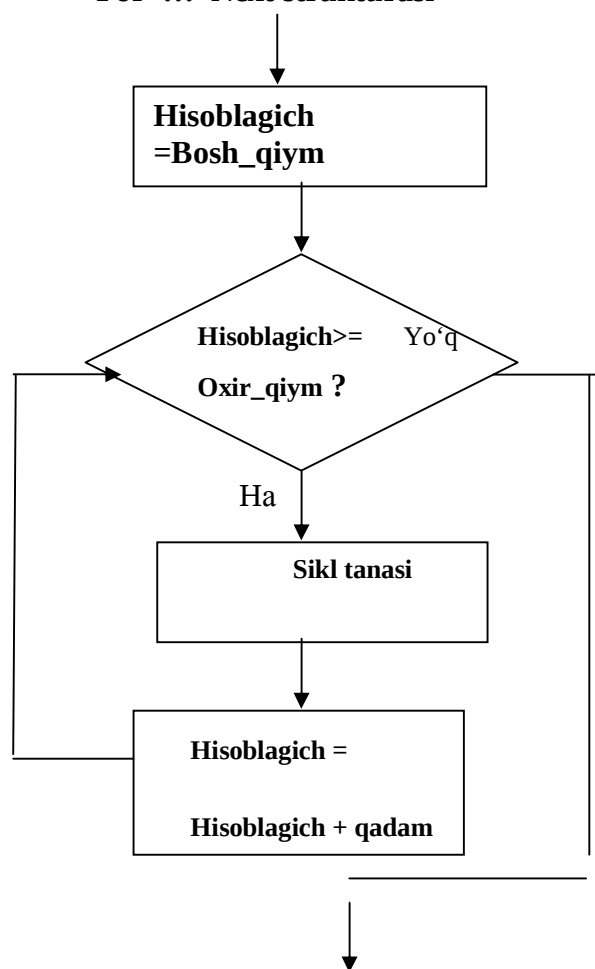
Musbat qadamli

For ... Next strukturasini



Manfiy qadamli

For ... Next strukturasini



Agar **Bosh_qiym** ning qiymati **Oxir_qiym** qiymatidan katta (musbat **qadam** da) yoki **Bosh_qiym** ning qiymati **Oxir_qiym** qiymatidan kichik (manfiy **qadam** da) bo'lsa, sikl tanasi bir marta ham bajarilmaydi.

Agar **Hisoblagich** ning qiymati sikl ichidagi shart yakunlanish holati bajarilmaydigan qilib o'zgartirilsa (masalan, iteratsiya oxirida sikl **Hisoblagichiga** **Bosh_qiym** qiymati o'zlashtiriladi), unda "sikllashuv" hosil bo'ladi, ya'ni sikl cheksiz bajariladi.

For ... Next strukturasini ishlatish. Arifmetik progressiyaning dastlabki 100 ta elementinining yig'indisini hisoblash.

Arifmetik progressiya o'zining a_0 nollik elementi va d ayirmasi bilan beriladi. Arifmetik progressiyaning har bir kelasi elementi formulasi $a_k = a_{k-1} + d$ bilan hisoblanadi.

Arifmetik progressiyaning dastlabki 100 ta elementinining yig'indisi quyidagicha:

$$S = \sum a_k, \text{ bu erda } k \text{ 0 dan 100 gacha bo'lgan qiymatlarni ketma-ket qabul qiladi.}$$

Algoritmi

1. a_0 va d larni kiritingiz
2. Arifmetik progressiyaning yig'indisi - S o'zgaruvchisiga a_0 ning qiymatini o'zlashtiringiz

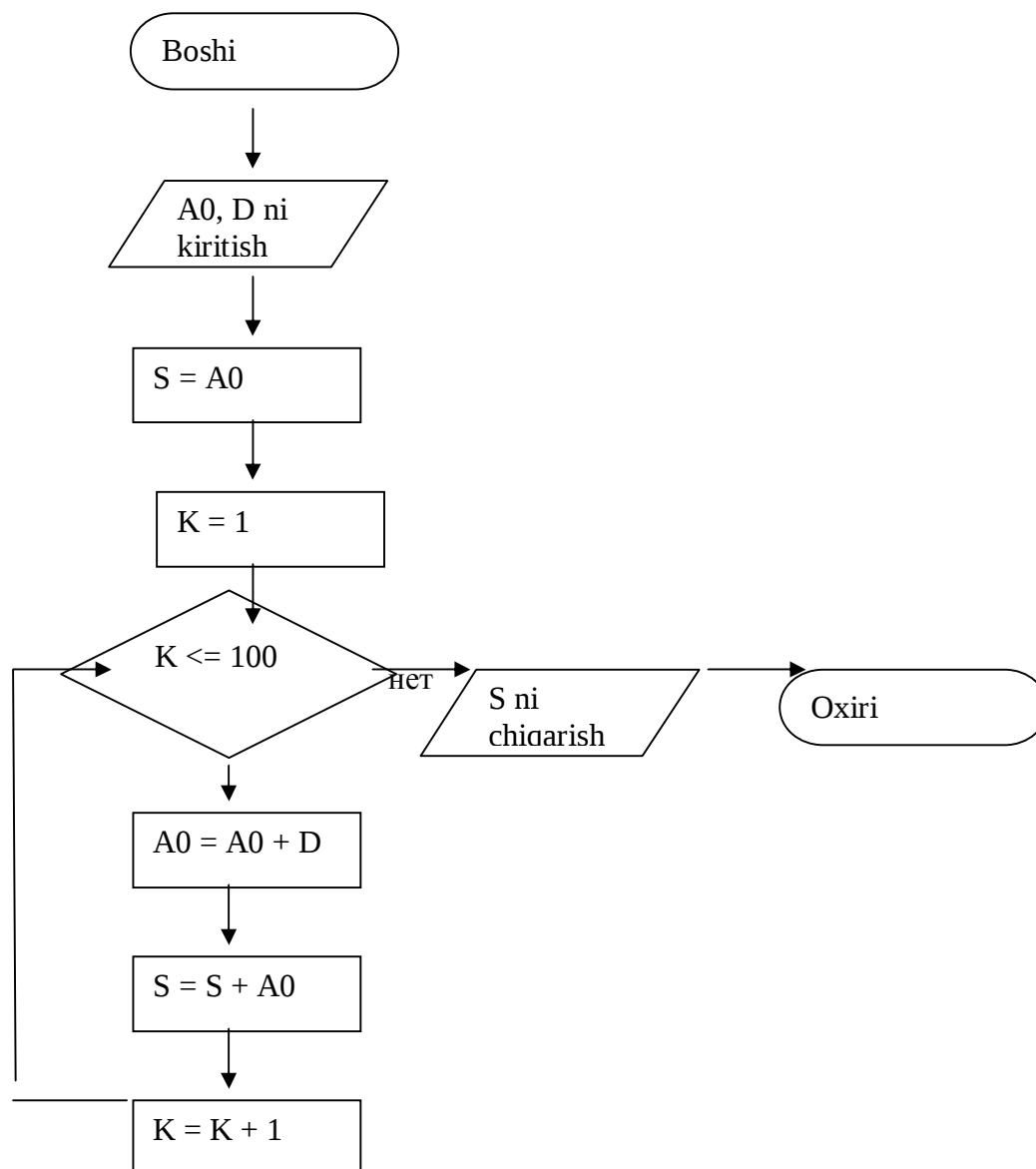
3. For K = 1 To 100 ... Next siklida har bir iteratsiyada arifmetik progressiyaning har bir elementini va yig'indining qiymatini quyidagi formula yordamida hisoblanadi

$$a_k = a_{k-1} + d \quad S = S + a_k$$

4. S ni chiqarish

Obyekt	Xossa	O'rnatilgan qiymatlari
Form1	Caption	For ... Next strukturasi
Label1	Caption	Arifmetik progresiyaning nolinchi elementini kiriting
Label2	Caption	Arifmetik progresiyaning ayirmasini kiriting
Label3	Caption	Arifmetik progressiyaning dastlabki 100 ta elementining yig'indisi
Command1	Caption	Hisoblash
Command2	Caption	Tugatish
Text1, Text2, Text3	Text	Text xossasi maydonini tozalash

Blok – sxemasi



The screenshot shows a window titled "Far ... Next strukturasi". It contains three input fields with labels and two buttons at the bottom.

Label	Value
Arifmetik progressiyaning nolinch elementini kiriting	1.5
Arifmetik progressiyaning ayirmasini kiriting	0.2
Arifmetik progressiyaning dastlabki 100 ta elementining yig'indisi	1161.5

Buttons: Hisoblash, Tugatish

For ... Next strukturasini ishlatish

M va A butun sonlarini kiritish. 1 dan M gacha bo'lgan natural sonlari kublarining yig'indisi S ni hisoblash. Agar yig'indi A dan oshib ketsa, hisoblash to'xtatiladi. Olingan yig'indi S ni va natural qatorning oxirgi sonining qiymati K ni chiqarish kerak.

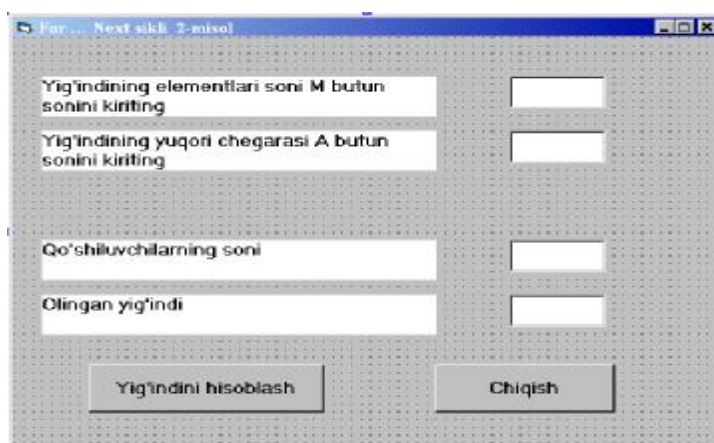
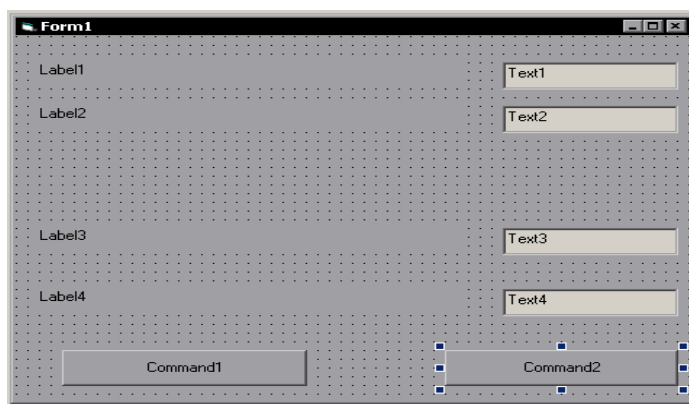
Algoritmi

1. M va A ni kiritish.
2. Kublar yig'indisi S o'zgaruvchisiga 0 qiymatini o'zlashtiringiz.
3. For K = 1 To M ... Next siklidagi har bir iteratsiyada K ning kubi qiymatini va quyidagi formula bo'yicha yig'indini hisoblangiz

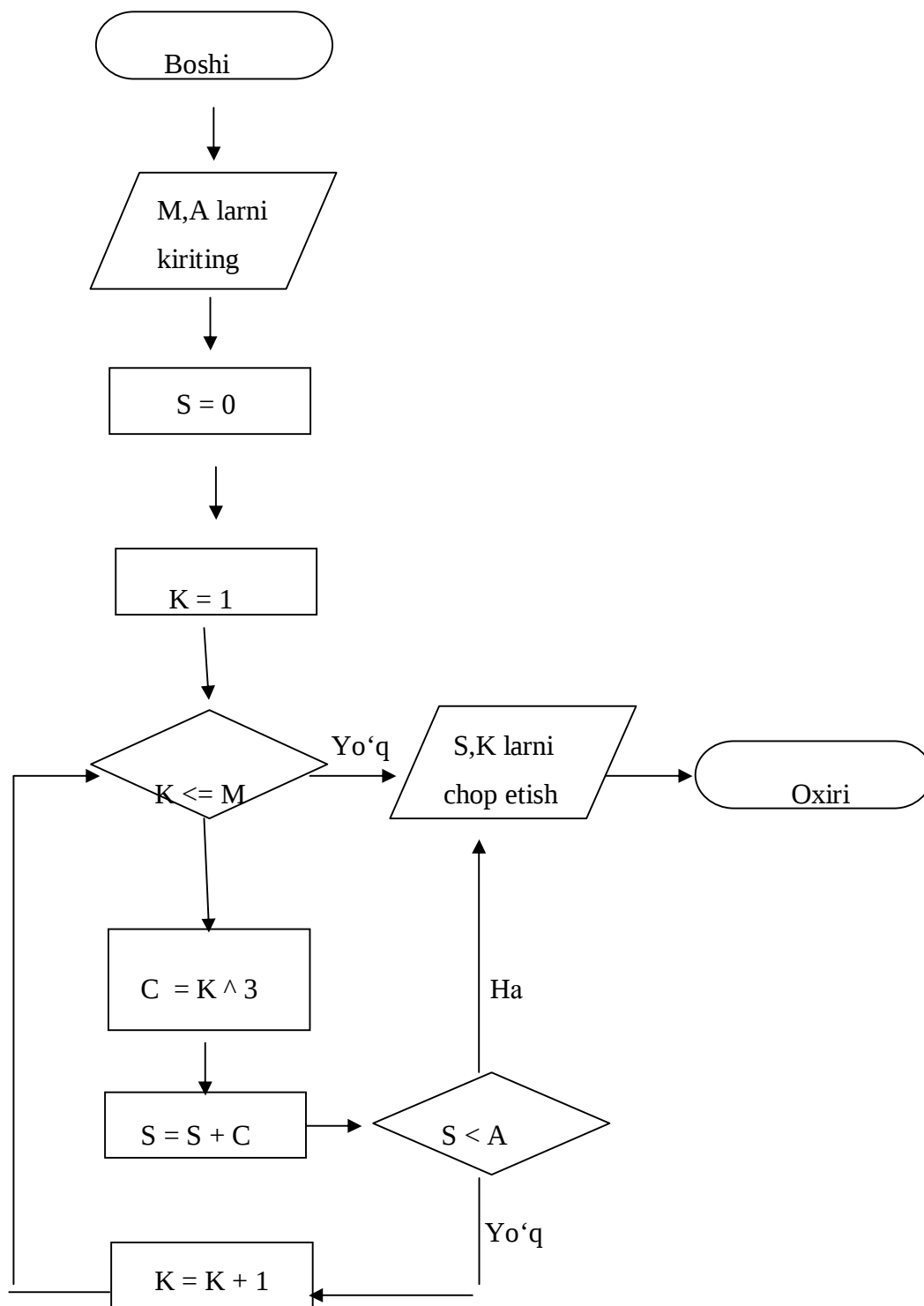
$$C = K^3 \qquad S = S + C$$

4. Har bir iteratsiyada S ning qiymati A ($S < A$) nazorat qiymatidan oshib ketmasligini tekshirish kerak. Agar S ni qiymati A dan oshib ketsa, siklni to'xtatish kerak, aks holda siklning keyingi iteratsiyasiga o'tadi.
5. S ni chiqarish

Sikl iteratsiyasining bajarilishini tugatmasdan For ... Next siklidan chiqish sikl ichiga joylashtiriladigan Exit For operatori yordamida amalgam oshiriladi



Blok – sxema



2. Sharti oldinda va oxirida berilgan sikllar

Bazi bir holatlarda sikl tanasining necha marta bajarilishini oldindan aniqlab bo'lmaydi. Bunday holatlarda sikldagi shartning qiymati True bo'lguncha bajariladi.

Bunaqangi sikllarning ikki holati mavjud:

Sikl sharti iteratsiyaga har bir kirishidan oldin tekshiriladi (sharti oldinda berilgan sikllar)

Sikl sharti har bir iteratsiyaning bajarilishidan so'ng tekshiriladi (sharti oxirida berilgan sikllar)

2.1. Sharti oldinda berilgan Do While siklining sintaksisi

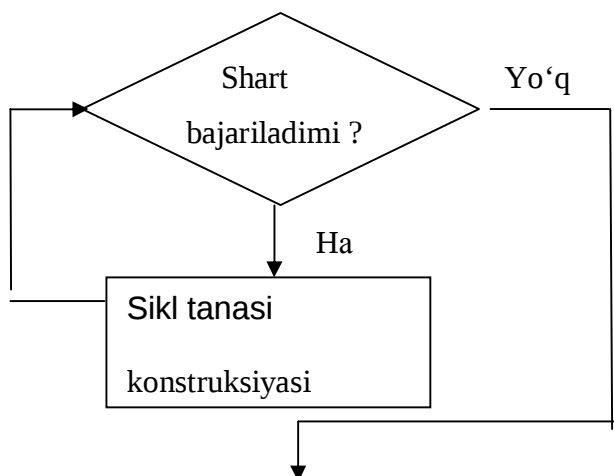
Do While Shart

Konstruksiya

Loop

Do While ... Loop konstruksiyasida berilgan sikl operatori undagi berilgan shartning qiymati True bo'lguncha davom etadi. Shart argumenti mantiqiy ifoda bo'lib, siklga har bir kirishda tekshiriladi. Agar uning qiymati True bo'lsa, Do While va Loop xizmatchi so'zlari orasidagi buyruqlar ketma-ketligi

bajariladi. Bu konstruksiya sikl tanasini tashkil qiladi. Agar siklga kirishda shartning qiymati False bo'lsa, sikldan chiqadi va boshqarish Loop dan keyingi konstruksiyaga beriladi. Shunaqangi holat mavjudki, ya'ni sikl ichidagi operator bir marta ham bajarilmasligi mumkin. Bu holat sikl sharti birinchi marta tekshirilganda uning qiymati False bo'lsa.



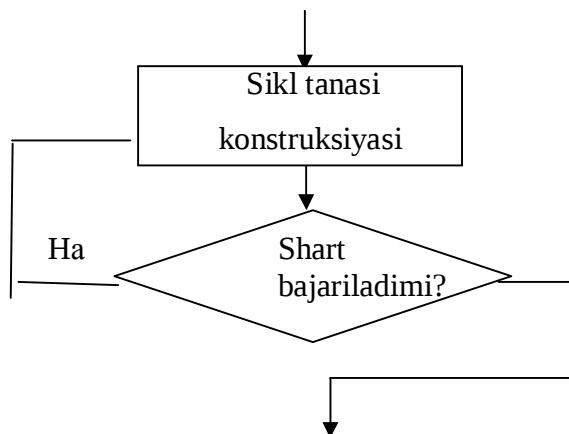
2.2. Sharti oxirida berilgan sikl

Do

Konstruksiyalar

Loop While Shart

Bu sikl strukturasini ishlatishda shart sikl oxirida tekshiriladi. Bu holatda sikl tanasi kamida bir marta bajariladi va undan keyin shart tekshiriladi.

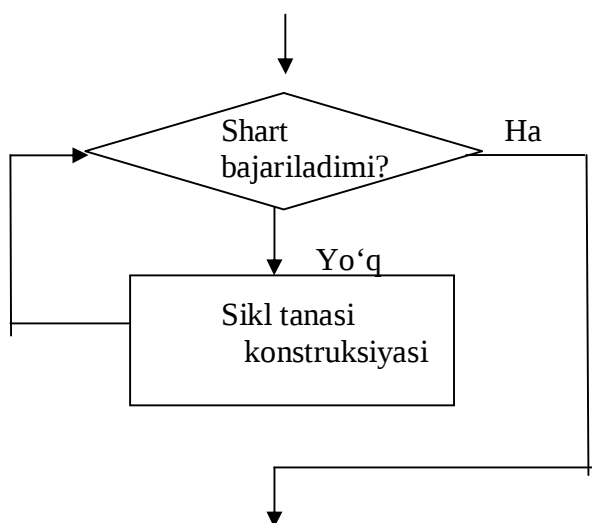


Do . . . Loop sikl konstruksiyasining yana bir ko'rinishi bor. Ular avvalgi ko'rilgan sikllar bilan bir xil, lekin sikl shartining qiymati False bo'lguncha bajariladi.

2.3. Sharti oldinda berilgan sikl

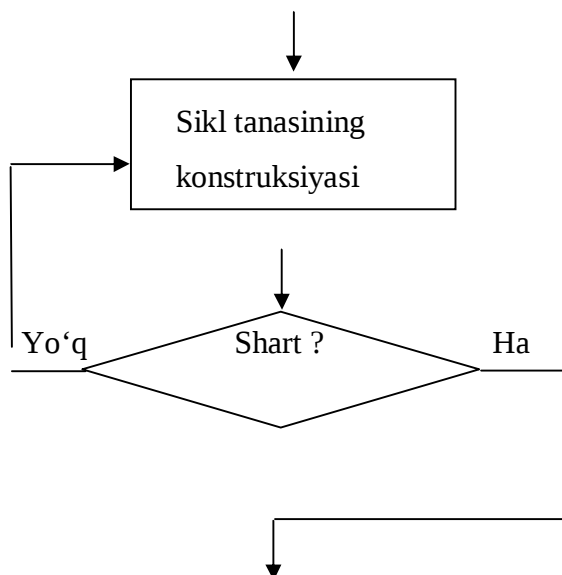
**Do Until Shart
konstruksiya**

Loop



2.4. Sharti oxirida berilgan sikl

**Do
Konstruksiya
Loop Until Shart**



Sikldan shartsiz chiqib ketish sharti oldinda va oxirida berilgan sikllarning strukturasi uchun **Exit** **Do** sintaksisiga ega. **Exit** buyrug'i siklning bajarilishini yakunlaydi va boshqarishni sikldan keyinga konstruksiyaga beradi.

Shartli sikl strukturasi ishlatish (shart sikl tanasi bajarilishidan oldin tekshiriladi)

Kvadrat berilgan. Kvadrat tomoni 2 ga teng. Kvadrat markazi dekart koordinatalari sistemasining (0,0) nuqtasida joylasgan. Siklda kvadrat ichidan tasodifiy nuqta olinadi. Nuqta markazlari (0,0) nuqtada

va radiuslari $R_1=0,5$; $R_2 =1,0$ bo'lgan ikkita konsentrik aylanadan tashkil topgan halqaga tegishlilikini tekshirish kerak. Quyidagi shartlarning biri bajarilsa, siklni to'xtatiladi.

Tasodifiy olingan nuqtalarning soni 300 ga teng bo'lsa.

Halqaga tegishli bo'lgan nuqtalar soni 50 ga teng bo'lsa.

Nuqtalarning umumiy soni va halqaga tegishli bo'lgan nuqtalarning soni chop etiladi.

Algoritmi

1. K_1 – aniqlanadigan nuqtalarning tartibi. Siklga kirishdan oldin unga nol qiymatini berish lozim.
2. K_2 – halqaga tushuvchi nuqtalar soni. Siklga kirishdan oldin unga nol qiymatini berish lozim.
3. **Sikl sarlavqasi.** $K_1 \leq 300$ va $K_2 \leq 50$ shartlarini tekshirish.
4. Agar tekshirish natijasi True bo'lsa, $K_1 = K_1 + 1$ iteratsiyasini amalga oshirish

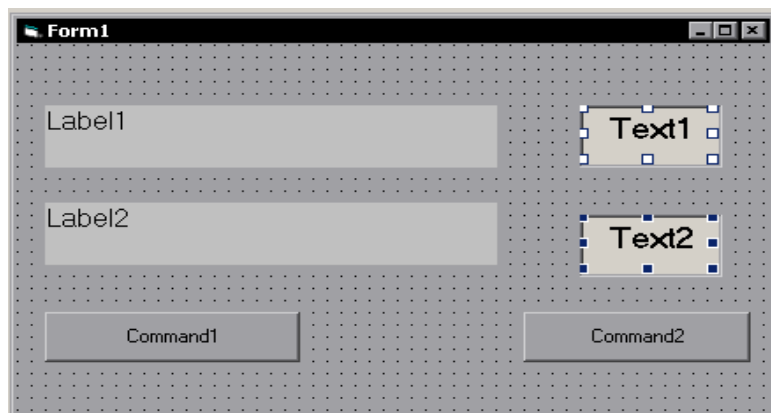
(x,y) nuxtasi koordinatalarini topish

$$-1.0 \leq x \leq +1.0, -1.0 \leq y \leq +1.0$$

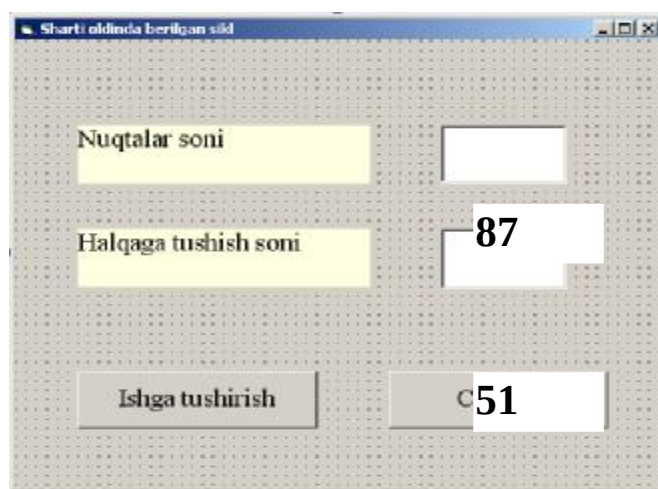
Nuqtaning $0.25 \leq x^2 + y^2 \leq 1.0$ halqasiga tegishlilikini tekshirish

Agar $K_2 = K_2 + 1$ sharti bajarilsa, 3 punktga o'tish

5. K_1 , K_2 larni chiqarish



Obyekt	Xossasi	O'rnatilgan qiymatlari
Form1	Caption	Sharti oldinda berilgan sikl
Label1	Caption	Nuqtalar soni
Label2	Caption	Halqaga tushishlar soni
Command1	Caption	Ishga tushirish
Command2	Caption	Dasturdan chiqish
Text1, Text2	Text	Text xossasi maydonini tozalash



Shartli sikllarning strukturasi ishlatish (sikl tanasi bajarilganidan so'ng shartni tekshirish)

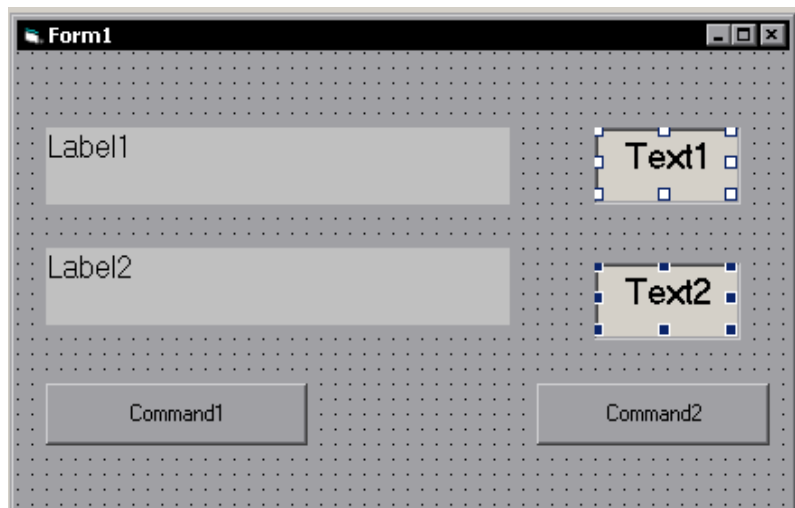
Tasodifiy sonlar datchigi yordamida 0 dan 1000 gacha bo'lgan intervaldan butun sonni aniqlash. Urinishlar 950 dan katta bo'lgan qiymatni olguncha davom etadi. Olingan natijani va urinishlar sonini chiqarish kerak.

Algoritmi

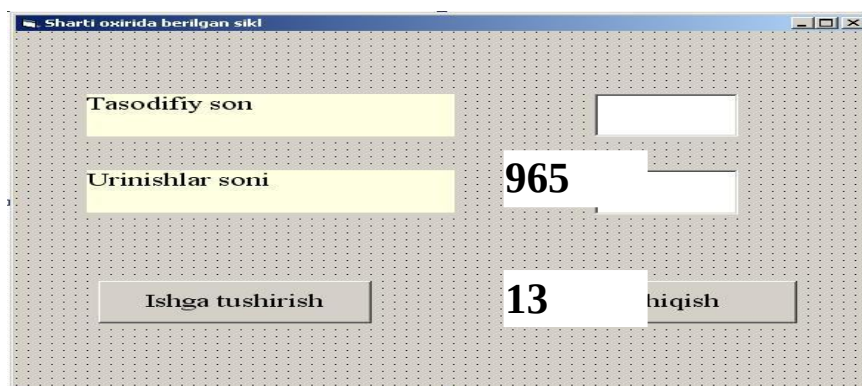
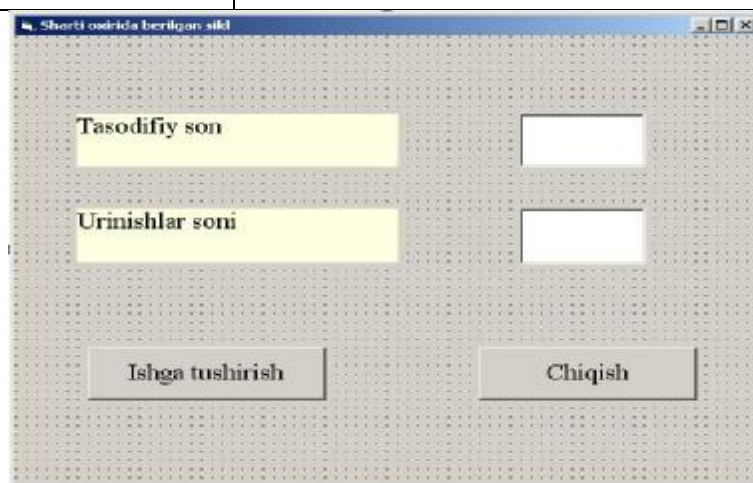
1. K – urinishlar soni - до входа в цикл обнулить
2. **Sikl sarlavqasi**
3. Sikl iteratsiyasi

$$K = K + 1$$

$$A = 1000 * \text{int}(\text{Rnd})$$
 tasodifiy sonini aniqlash
4. Agar $A \leq 950$ bo'lsa, 3-bosqichqa o'tish
5. A, K larni chiqarish



Obyekt	Xossa	O'rnatilgan qiymatlari
Form1	Caption	Sharti oxirida berilgan sikl
Label1	Caption	Tasodifiy son
Label2	Caption	Urinishlar soni
Command1	Caption	Ishga tushirish
Command2	Caption	Chiqish
Text1, Text2	Text	Text xossasi maydonini tozalash



3.Foydalangan adabiyotlar ruyxati

1. Delphi tilida dasturlash asoslari, G'afur G;ulom nomidagi nashriyot-matbaa ijodiy uyi Tosh-2007
- 2.Файсман А. Профессиональноэ программированиэ на Турбо Паскале. 1992.
- 3.Культин М.Б. Программированиэ в Турбо Пассал и Delphi, Санкт-Петер-бург, 2002 г.
- 4.Кондзюба С.П,ГромовВ.Н. Delphi . База даннх и приложенийЙ.М.-Санкт-Петер-бург- Киэв, 2002 г.
- 5.WWW.Интуит.ру. Интернет-Университет информасионнх технологий. Москва.
6. WWW.REF.uz Internet sayti.