

O‘ RESPUBLIKASI OLIY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI

**ISLOM KARIMOV NOMIDAGI
TOSHKENT DAVLAT TEXNIKA UNIVERSITETI**

TEXNIK TIZIMLARDA AXBOROT TEXNOLOGIYALARI

**Amaliy mashg‘ulotlarni bajarish uchun
o‘quv-uslubiy ko‘rsatmalar**



TOSHKENT – 2022

Tuzuvchilar: Kadirov M.M., To‘laganov Z.Ya., Zokirova F.R.

“Texnik tizimlarda axborot texnologiyalari” fanidan amaliy mashg‘ulotlar ishlarini bajarish uchun o‘quv-uslubiy ko‘rsatmalar. - Toshkent: ToshDTU, 2022 y. 50b.

Ushbu amaliy mashg‘ulotlar kechki ta‘lim talabalarining “60711400- Texnologik jarayonlar va ishlab chiqarishni avtomatlashtirish va boshqarish (tarmoqlar bo‘yicha)”, “60610200 - Axborot tizimlari va texnologiyalari (tarmoqlar va sohalar bo‘yicha)”, “60711300 - Metrologiya, standartlashtirish va mahsulot sifati menejmenti (tarmoqlar bo‘yicha)”, “60711500 - Mexatronika va robot texnika”, “60711600 - Intellektual muhandislik tizimlari (tarmoqlar va sohalar bo‘yicha)”, “60711200 - Elektronika va asbobsozlik (elektronika sanoatida)” ta‘lim yo‘nalishlariga mo‘ljallangan bo‘lib, u texnik tizimlarda ma‘lumotlarni qayta ishlash, MathCad tizimining ishlash texnologiyasini o‘rganish, MathCad integrallashgan sohasida muhandislik masalalarini yechish, MatLab muhitining integrallashgan sohasini o‘rganish va ob‘yektga yo‘naltirilgan dasturlash, shuningdek himoyalashning kriptografik usullaridan foydalanish kabi mavzular bo‘yicha amaliy ishlarni bajarish uchun mo‘ljallangan.

*Islom Karimov nomidagi Toshkent davlat texnika universiteti ilmiy-
uslubiy kengashi qaroriga muvofiq chop etildi
(11-sonli bayonnoma, 28.09.2022 y.)*

Taqrizchilar:

Abdurashidova K.T - TATU “Kompyuter tizimlari” kafedrası dotsenti;

Karimova N.O. - ToshDTU “Axborot texnologiyalari” kafedrası dotsenti.

©Toshkent davlat texnika universiteti, 2022

1 - AMALIY MASHG'ULOT

MATHCAD TIZIMI. TIZIMNING ISHLASH TEXNOLOGIYASINI O'RGANISH. MATHCAD INTEGRALLASHGAN SOHASIDA MUHANDISLIK MASALALARINI YECHISH

Ishning maqsadi: MatCAD tizimini interfeysi bilan ishlash va tizim yordamida texnik-muhandislik hisoblashlarni amalga oshirish ko'nikmalarini olish.

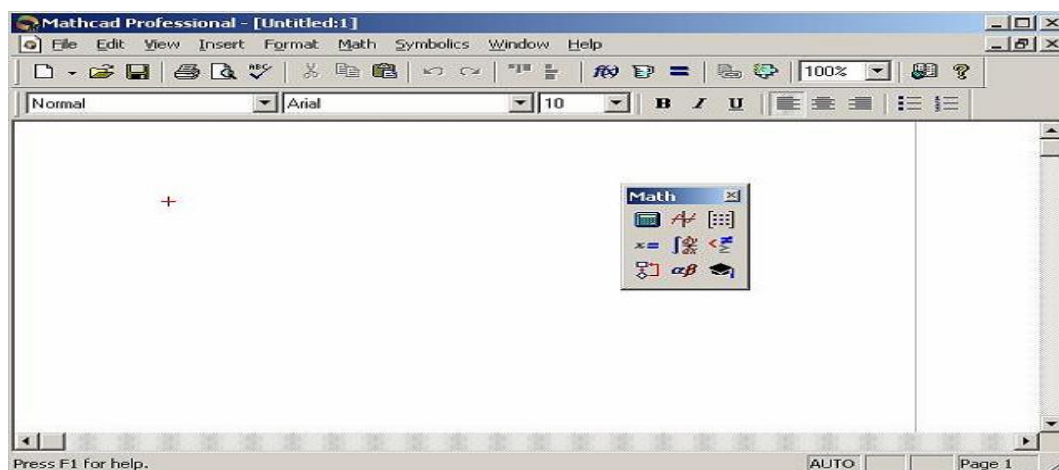
Reja:

1. MathCAD dasturiy sohasini o'rganish.
2. Ishchi paneli tarkibi.
3. Asosiy instrumentlar.
4. Ishchi sohani boshqarish.

Nazariy qism

MathCAD integrallashgan tizim bo'lib, matematik hamda texnik-muhandislik hisoblashlarni amalga oshirish uchun mo'ljallangan. Hisoblash vaqtida tushunarlik, aniqlik, oddiqlik kabi xususiyatlarni o'z ichiga oladi. Elektron jadvallarga xos foydalanishda oson.

MathCAD dasturini ishga tushirganda 1-rasmdagi oyna ochiladi.



1-rasm. MathCAD tizimining ishchi oynasi

MathCAD dasturining bosh menyusi boshqa MS Windows ilovalarning bosh menyulari kabi imkoniyatlarga ega bo'lgan buyruqlardan tashkil topgan.

File menyusi – fayllar bilan ishlash.

Edit menyusi – Hujjatlarni tahrirlash.

View menyusi – darcha elementlarni sozlash.

Insert menyusi – MathCAD hujjatga grafik ma'lumotlar, matritsalar, funksiyalar, gippersilkalar, komponentalarni qo'yish va obyektlarni so'zlash imkonini beradi.

Format menyusi – sonlar, formulalar, matnlar, abzaslar, kolontitullar va boshqalarga turli ko'inishdagi parametrlarni belgilovchi buyruqlarni o'z ichiga oladi.

Math menyusi– hisoblashlarni, rejim va parametrlarni o'rnatadi.

Symbolic menyusi – simvol malumotlarni hisoblaydi.

Window menyusi – bir necha darchalarni o'zaro ketma-ket va ulardan birini aktivlashtirishga xizmat qiladi.

Help menyusi – ma'lumot markazi va ma'lumotnomalar.

Math paneli tugmalari

MathCADning kuchli tomoni bu matematik simvollar, ularni ifodalash va kiritish insonga odatiy holatda berilgan. Ushbu instrumentlar paneleni bosh menyuning buyruqlari orasidagi **View** → **Toolbars** orqali ishga tushiriladi. **Math** panelida ishning qulayligi uchun ssilkalrning yig'indisi birlashtirilgan.

Math panelida 9 ta tugma joylashtirilgan bo'lib, har bir tugma o'z navbatida, maxsus vazifaga birlashtirilgan instrumentlar panelini ishga tushiradi.

Calculator. Bu panelda matematik topshiriqlar buyruqlari, hamda ko'pincha foydalaniladigan funksiyalar joylashtirilgan. Bu tugmadan kalkulyator sifatida foydalanish mumkin.

Boolean – taqqoslash operatorini va mantiqiy amallarni kiritish.

Evaluation – o'zgaruvchilar qiymatlarini va funksiyalarini o'zlashtirish operatori kirituvchi tugmasi bor.

Graph – grafika tuzish instrumenti.

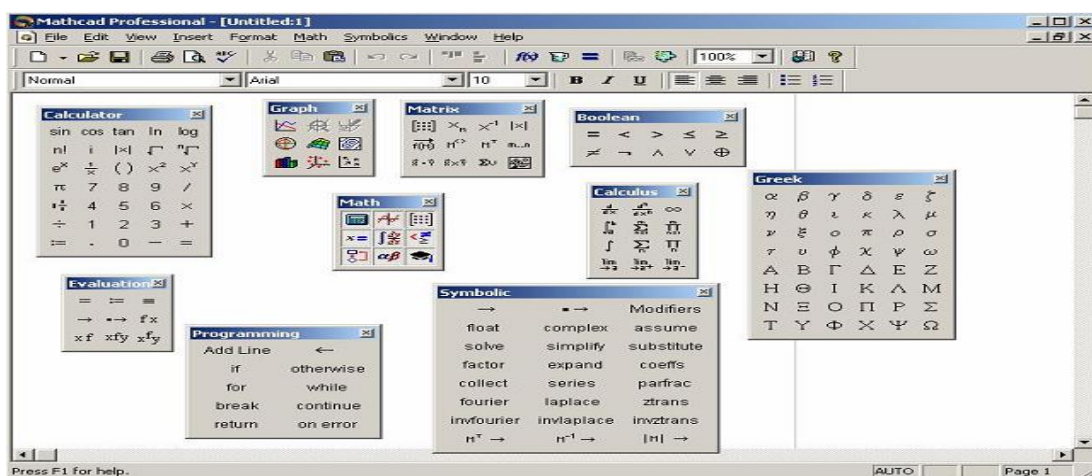
Vector and Matrix – vektorlar va matrisalar bilan ishluvchi instrumentlar.

Calculus – integrallashgan elementlarning matematik ifodalar ko'inishini differensiallashgan usulda ko'rsatadi. Bu paneldagi tugmalar qiymatlar summasi va ko'paytmalarini hisoblashga ijozat beradi.

Programming – dastur yozish instrumenti.

Greek Symbol – grek alifbo.

Symbol – Simvulli hisoblashlar uchun.



2-rasm. MathCad tizimining ishchi oynasi Math panelidagi ochilgan asboblari paneli

Matematik ifodalar

MathCAD ifodasining asosiy matematik elementlariga berilganlar toifasi, operatorlar, funksiyalar va boshqaruv tuzilmalari kiradi.

Operatorlar – MathCAD elementlari, ular yordamida matematik ifodalar yaratish mumkin. Ularga masalan arifmetik amallar simvoli, yig'indini hisoblash belgilari, ko'paytmalar, integrallar va boshqalar kiradi.

MathCAD tizimida ishchi hujjatga buyruqlar yozish

MathCAD tizimida buyruqlarni yozish qog'ozda yozib ishlaganga yaqin, va bu masalaning qo'yilishini va yechilishini osonlashtiradi. Natijada matematik vazifaning yechilishi dasturlashdan algoritmik tuzilishiga o'tadi.

Ma'lumotlar turiga sonli konstantalar, odatiy va tizimli o'zgaruvchilar, massivlar, (vektorlar va matrisalar) va fayl turdagi ma'lumotlar.

O'zgaruvchilarning foydalanish turlari

MathCAD tizimida quyidagi turdagi ma'lumotlar ko'rsatilgan:

1. Butun (2, -54,+43).
2. Haqiqiy (1.3,-2.23).
3. Notural($2.5+7i$). Kichik birlikni yozishda maxsus tugma **Calculus** panelidan foydalanish tavsiya etadi.
4. Satrli. Odatda u izoh turi: «hisoblangan natija».
5. Tizimli. Tizimli konstanta oldindan belgilangan o'zgaruvchi, berilishi tizimni yuklash vaqtida beriladi. Bu konstantalarga misol e yoki π .

Oddiy hisoblashlar

Arifmetik hisoblashning natijasi, undan keyin « \Rightarrow » yoki « \rightarrow » belgisi qo'yilsa. Birinchi holatda natija *sonli* ko'rinishda, ikkinchisida *simvulli* ko'rinadi.

Simvulli hisoblashga misol:

$$\frac{2.45}{6.178} + \frac{4}{52} - 76 - \frac{8}{87} \rightarrow -75.618462477305312281$$

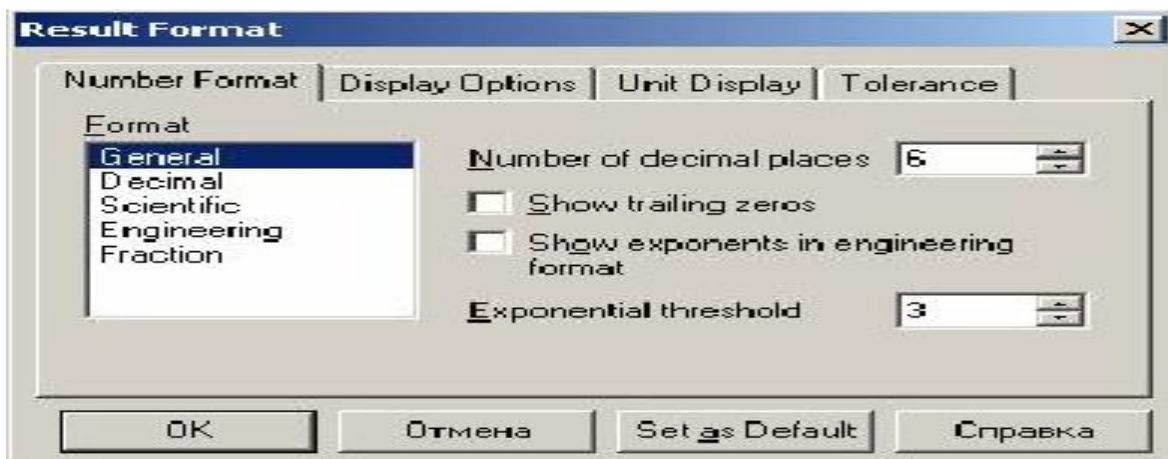
Hisoblash tizimlarida arifmetik amallar bilan ish olib borilganda matematikaning oddiy qonunlari amal qiladi. Hisoblashlar boshqa turdagi amallarni o'z ichiga olishi mumkin:

- ildiz chiqarish;
- darajasini kiritish;
- integrallash va differensiallash;
- faktorial va natija belgilari.

Amallarning ma'lum qismini **Calculator** panelidan olishimiz mumkin. Misol quyidagicha bo'lishi mumkin:

$$4.5 \cdot (\sqrt[5]{56.3} + \sqrt{14.356}) + 5.2^{1.8} - 4.89 + \frac{6.52}{4.78} = 43.046$$

Hisoblash vaqtida sonlarning natijasini, Format→Result orqali boshqarish mumkin. U holatda buyruq 3-rasmda ko'rsatilgan oynani hosil qiladi, qaysiki kirituvchi ma'lumotlarning parametrini qaytatan o'rnatishi kerak bo'ladi.



3-rasm. **Format** menyusining ishchi oynasi (format **Result**)

Quyida arifmetik berilishi simvolli hisoblash ko'rsatilgan:


$$\frac{25}{47} - 3^{-2} + \frac{7}{3} \cdot 2.5 + \pi \rightarrow 6.2541371158392434988 + \pi \text{ float},4 \rightarrow 9.396$$

«→» belgisidan keyin simvolli hisoblashning natijasi korsatilgan. Simvolli hisoblash natijasini sonli hisoblash natijasiga o'zgartirishi Symbolic panelidagi float buyrug'i orqali amalga oshiriladi. Bu buyruq shablon bo'lib, foydalanuvchi belgilarning (sonlar) miqdorini kiritish kerak.

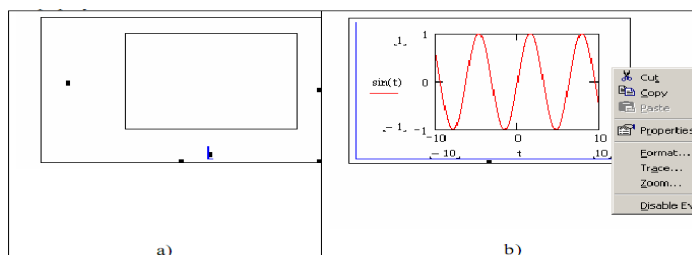
MathCAD tizimida grafika foydalanishda qulay va universal.

Grafik soha – 3 asosiy turga bo'linadi – ikki o'lchamli, uch o'lchamli va qo'yilgan grafik obrazlar. Ikki o'lchamli va uch o'lchamli grafika MathCAD tizimida o'zi qayta ishlangan ma'lumotlardan tuziladi.

Dekort grafikani yaratish:

1. Kursorni ishchi hujjatni bo'sh joyiga o'rnatib.
2. **Insert**⇒ **Graph** ⇒**X-Y Plot**, buyrug'ini tanlaymiz yoki **Shift+@** tugmalar yig'indisini tanlaymiz, **Graph** panelidagi  tugma orqali. Dekort grafikaning shabloni hosil boladi.
3. X yoyi ostining o'rta belgisining tagida birinchi mustaqil o'zgaruvchini kiriting. Vergul belgisidan so'ng-ikkinchisini va 10 gacha, misol uchun x1,x2,....
4. Y yoyining chap tomonidan birinchi mustaqil o'zgaruvchini vergul bilan kiriting va shu asosda qolganlarini ham kiritamiz va h.k. Misol uchun y1(x1), y2(x2),.....
5. Grafik muhitidan tashqariga sichqonchanning chap tugmasini bosib va siz grafikadan chiqasiz.

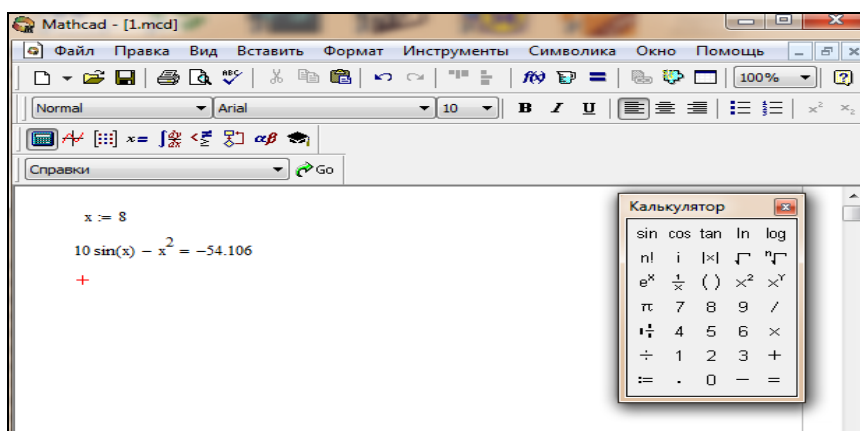
Ikki o'lchamli grafikani tuzish 4-rasmda ko'rsatilgan.



4-rasm. **Dekart koordinata** tizimida grafik muhit

MathCAD tizimida grafikani nuqtalar orqali tuzish mumkin. Bu holatda 2ta ustun qiymati kiritiladi x va y shu tekislikda shu ustunlar asnoda nuqtalar quriladi. Ustunlar qiymati **Matrix** panelidagi matritsa rasmi belgini bosib beriladi. Grafikaning o'zini olish uchun esa **Graph** panelidagi koordinata o'qi tugmasi bosiladi. Ramkada 2 ta qora to'ldirilmagan to'g'ri burchak marker hosil bo'ladi. Bitta markerga matritsa-ustunining nomi kiritiladi va u kordinatta o'qi, qaysiki y koordinata o'qiga qo'yiladi. Boshqa (Pastgi) markerga boshqa ustun nomi kiritiladi va keyinchalik **enter** tugmasi bosiladi.

1- Mashq. Misolni hisoblash: $10\sin x - x^2$



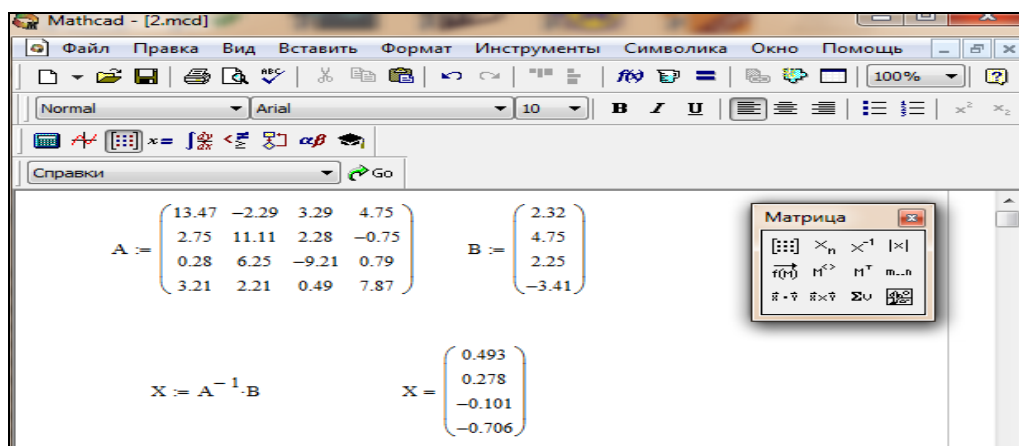
5-rasm. Misolni ishlash

2-Mashq. Matrisalar bilan ishlash (matrisa usulida tenglamani yechish)

1-jadval

A				B
13.47	-2.29	3.29	4.75	2.32
2.75	11.11	2.28	-0.75	4.75
0.28	6.25	-9.21	0.79	2.25
3.21	2.21	0.49	7.87	-3.41

Instrumentlar panelidan matritsa tanlab matritsaning qiymatini kiriting. Ishchi muhitda formulani kiriting. 6-rasmida ko'rsatilgan.



6-rasm. Matritsa

Nazorat savollari:

1. MathCad tizimida qaysi turdagi hisoblashlar bajariladi?
2. MathCad tizimida funksiya grafigi qaysi holatda amalga oshiriladi?
3. MathCad tizimida matrisalar bilan ishlash usullari ko'rsatilgan?

2-AMALIY MASHG'ULOT MATHCAD TIZIMIDA FUNKSIYA GRAFIKLARINI AMALGA OSHIRISH

Ishning maqsadi: MatCAD tizimida funksiya grafiglarni qurish ko'nikmalarini o'rganish.

Reja:

1. MathCAD dasturiy sohasini o'rganish.
2. Grafiglarni yaratish ma'lumotlarni kiritish
3. Ikki o'lchamli grafig qurishni o'rganish.
4. Uch o'lchamli grafig qurishni o'rganish.

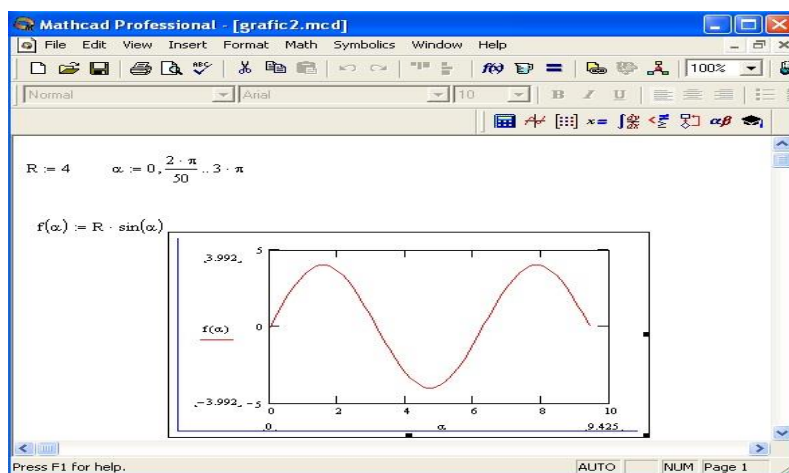
Nazariy qism

Ikki o'lchamli grafig qurish. Ikki o'lchamli funksiya grafigini qurish uchun quyidagi ketma-ketlikni bajarish kerak:

1. Qaysi joyga grafig qurish kerak bo'lsa, shu joyga krestli kursor qo'yiladi.
2. Matematik panelining Graph (Grafig) panelidan x-y Plot (Ikki o'lchovli grafig) tugmasi bosiladi.
3. Hosil bo'lgan ikki o'lchamli grafig shabloniga absiss o'qi argumenti nomi, koordinata o'qiga funksiya nomi kiritiladi.

4. Argumentning berilgan o'zgarish diapazonida grafikni qurish uchun grafik shabloni tashqarisi sichqonchada bosiladi. Agar argumentning diapazon qiymati berilmasa, u holda avtomatik ravishda argument diapazon qiymati 10 dan 10 gacha bo'ladi va shu diapazonda grafik quriladi (7-rasm). Grafik formatini qayta o'zgartirish uchun grafik maydoniga ikki marta sichqonchani bosish va ochilgan muloqot oynasidan kerakli o'zgarishlarni qilish kerak.

Agar bir necha funksiyalar grafigini qurish kerak bo'lsa va ular argumentlari har xil bo'lsa, u holda grafikda funksiyalar va argumentlar nomlari ketma-ket vergul qo'yilib kiritiladi. Bunda birinchi grafik birinchi argument bo'yicha birinchi funksiya grafigini va ikkinchisi esa mos ravishda ikkinchi argument bo'yicha ikkinchi funksiya grafigini tasvirlaydi va hokazo.



7-rasm. Funksiya grafigini qurish

Quyida grafik formati muloqot oynalarini ko'rib chiqamiz.

6. X-Y Axes – koordinata o'qini formatlash. Koordinata o'qiga setka, sonli qiymatlarni grafikga belgilarni qo'yish va quyidagilarni o'rnatish mumkin:

- ✓ LogScale – logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
- ✓ Grid Lines – chiziqqa setkalar qo'yish;
- ✓ Numbered – koordinata o'qi bo'yicha sonlarni qo'yish;
- ✓ Auto Scale – son qiymatlar chegarasini o'qda avtomatik tanlash;
- ✓ Show Markers – grafikka belgi kiritish;
- ✓ Autogrid – chiziq setkasi sonini avtomatik tanlash.

2. Trace – funksiya grafiklarini formatlash. Har bir funksiya grafigini alohida o'zgartirish mumkin:

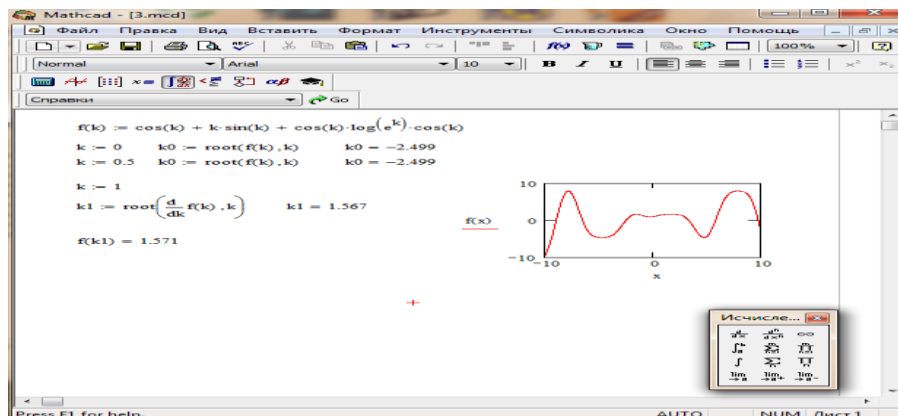
- ✓ chiziq ko'rinishi (Solid – uzliksiz, Dot – punktir, Dash – shtrixli, Dadot – shtrixli punktir);
- ✓ chiziq rangi (Color);
- ✓ grafik tipi (Type) (Lines – chiziq, Points – nuqtali, Bar yoki SolidBar – ustunli, Step – pog'onali grafik va boshqa);
- ✓ chiziq qalinligi (Weight);
- ✓ simvol (Symbol) – grafikda hisoblangan qiymatlar uchun (aylana, krestik, to'g'ri burchak, romb).

3. Label – grafik maydoni sarlavhasi. Title (Sarlovha) maydoniga sarlavha matni kiritiladi.

4. Defaults – bu yordamida grafik ko‘rinishga qaytish mumkin.

Mashq. Hisoblash: $y = \cos x + x \sin x + (\cos x)x \cos x$.

Funksiya grafigini tuzish (8-rasm).



8-rasm. Funksiya grafigi

Uch o‘lchamli grafik qurish. Uch o‘lchamli grafik qurish uchun quyidagi ketma-ketlikni bajarish kerak.

1. Ikki o‘zgaruvchili funksiya nomini, keyin ($:=$) yuborish operatori va funksiya ifodasini kiritish.

2. Grafik qurish kerak bo‘lgan joyga kursor qo‘yiladi.

3. Matematik panelining Graph (Grafik) panelidan Surface Plot (uch o‘lchamli grafik) tugmasi bosiladi. Shu joyda uch o‘lchamli grafik shabloni paydo bo‘ladi.

4. Shablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 9-rasm chap tomon.

Ikki o‘zgaruvchili funksiya bo‘yicha grafik sirtini qurishni tez qilish maqsadida boshqa usul ham mavjud va u ayrim hollarda funksiya sirtini tuzishda funksiya massiv sonli qiymatlarini ishlatadi, masalan, 9-rasm chap tomon. Bunday grafikni qurish uchun quyidagi ketma-ketlikni bajarish kerak.

1. Diskret o‘zgaruvchilar yordamida ikki funktsiyaning o‘zgaruvchisi uchun ham qiymatlarini kiritish.

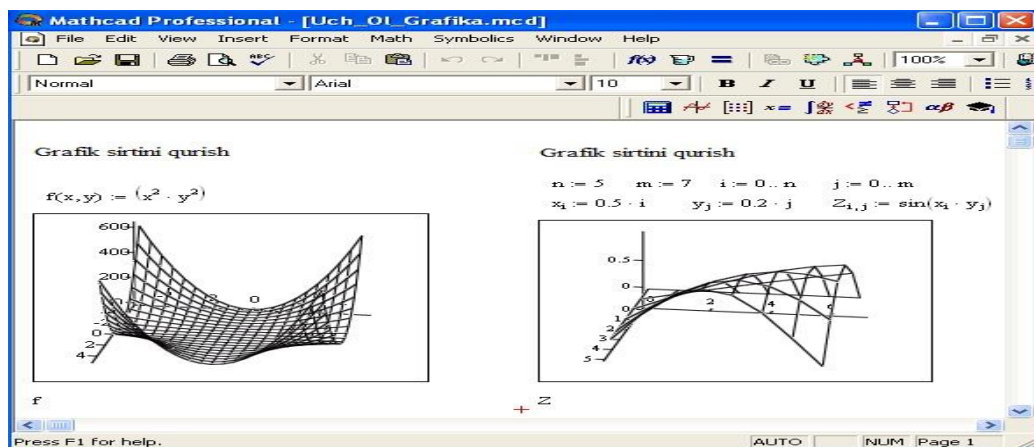
2. Massiv kiritish. Uning elementlari funksiya qiymatlari bo‘lib, ular berilgan funksiya argumentlari qiymatlaridan tashkil etiladi.

3. Kursor qaysi joyga grafik qurish kerak bo‘lsa shu joyga qo‘yiladi.

4. Grafik shabloniga funksiya nomi kiritiladi.

5. Shablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 9-rasm o‘ng tomon.

Grafik formatini qayta o‘zgartirish va unga ranglar berish uchun grafik maydonini ikki marta sichqonchani bosish va ochilgan muloqot oynasidan kerakli o‘zgarishlarni qilish kerak. Bu o‘zgartirishlar muloqot oynasi 10-rasmda berilgan.



9-rasm. Ikki o'zgaruvchili funktsiya grafigini qurish

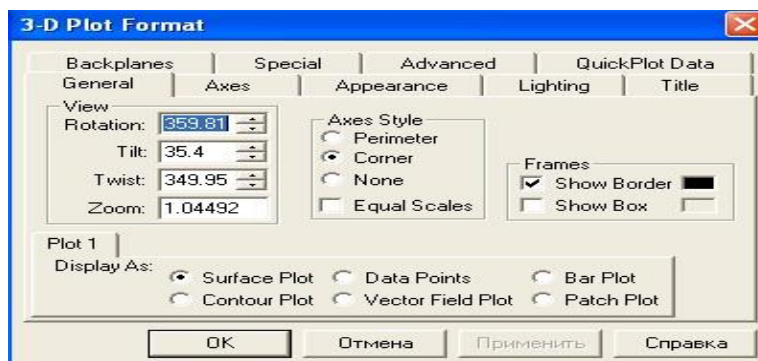
Bunda:

- ✓ Surface Plot – grafik sirti;
- ✓ Contour Plot – grafik chizig'i darajasi;
- ✓ Data Points – grafikda faqat hisob nuqtalarini tasvirlash;
- ✓ Vector Field Plot – vektor maydoni grafigi;
- ✓ Bar Plot – uch o'lchovli grafik gistogrammasi;
- ✓ Patch plot – hisob qiymatlari maydoni.

Bulardan tashqari yana bir qancha boshqarish elementlari mavjud. Ular grafikni formatlashda keng imkoniyatni beradi. Masalan, grafik masshtabini o'zgartirish, grafikni aylantirish, grafikga animatsiya berish va boshqa. 9-rasmda uch o'lchamli grafikni formatlash oynasi berilgan.

Grafikni boshqarishning boshqa usullari quyidagilar:

- ✓ *Grafikni aylantirish* uni ko'rsatib sichqonchanning o'ng tugmasini bosish bilan amalga oshiriladi.
- ✓ *Grafikni masshtablashtirish* Ctrl tugmasini bosib sichqoncha orqali bajariladi.
- ✓ *Grafikga animatsiya berish* Shift tugmasini bosish bilan sichqoncha orqali amalga oshiriladi.



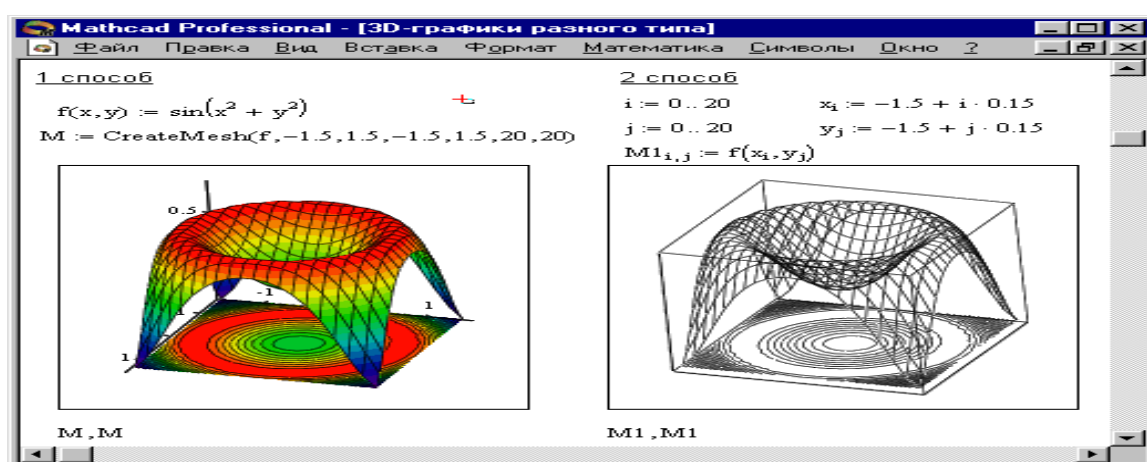
10-rasm. Grafikni formatlash oynasi

Uch o'lchamli yoki 3D grafika ikki o'zgaruvchan funktsiyani krsatadi $Z(X, Y)$.

MathCAD tizimida oldingilarda uch o'lchamli grafikani tuzishda yuzani matematik aniqlash kerak edi (11-rasm, 2 usul). Keyinchalik MathCAD *CreateMesh* funksiyasi qo'llaniladi.

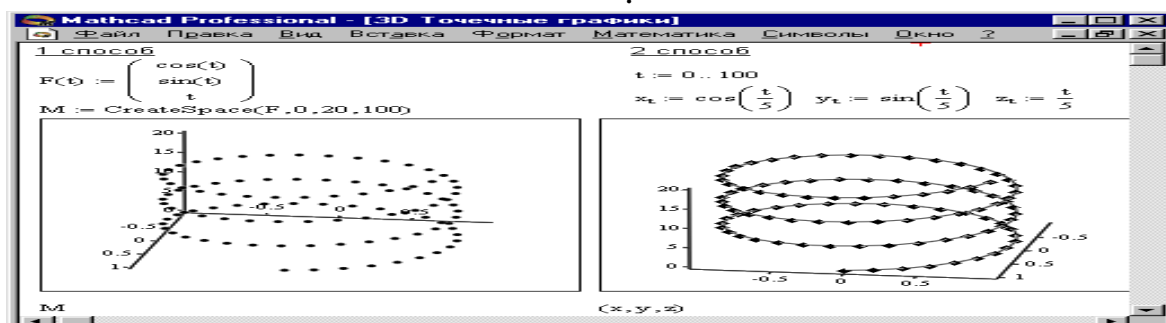
CreateMesh(*F* (yoki *G*, yoki *f1, f2, f3*), *x0, x1, y0, y1, xgrid, ygrid, fmap*) – Ma'lum funksiyani yuzasida to'r hosil qiladi, *F*. *X0, x1, y0, y1*– o'zgaruvchilar diapazonining o'zgarishi, *xgrid, ygrid* – o'zgaruvchilar to'ri hajmi, *fmap* – ko'rsatish funksiyasi. *CreateMesh* funksiyasi o'zgaruvchilar bilan yuzada to'r hosil qiladi -5dan 5gacha va 20×20 nuqtali to'r.

CreateMesh funksiyasidan foydalanish 3D grafika tuzish 11-rasm 1 usulda ko'rsatilgan. 11-rasmda bitta yuza har xil usul bilan tuzilgan, turli formatlash, yuza tegi va yuzaning o'zi ham kontur grafikada berilgan. Bu tuzilish rasmga ect taassurot beradi.



11-rasm. Bir rasmda 3D grafikani turli ko'rinishi

Bu grafika **Insert**⇒ **Graph** ⇒ **3D Scatter Plot** buyrug'i orqali tuziladi, yuzasi parametrli uch matrisalar yordamida kiritiladi (*X, Y, Z*) Boshlangich ma'lumotlarni aniqlash uchun *CreateSpace* funksiyasidan foydalaniladi



12-rasm. 3D nuqtali grafikani tuzish

CreateSpace (*F, t0, t1, tgrid, fmap*) – bunda uch o'lchamli vektor massiviga qaytiladi. *X*-, *y*-va *z*-koordinatalari, *F* aniq funksiyasi. *T0* va *t1* –

o'zgaruvchilarning o'zgarish diapazoni, *tgrid* – o'zgaruvchining o'lchov setkasi, *fmap* – aks etish funksiyasi.

Nazorat savollari:

1. Ikki o'lchamli grafik qurish qanday amalga oshiriladi?
2. Uch o'lchamli grafik qurish qanday amalga oshiriladi?
3. Funksiyalar bilan ishlash qanday amalga oshiriladi.

3 – AMALIY MASHG'ULOT MATLAB MUHITI, UNING INTEGRALLASHGAN SOHASINI O'RGANISH. SIMULINK PAKETI

Ishning maqsadi: Matlab dasturida matematik ifodalarni hisoblash. Simulink paketi bilan tanishish.

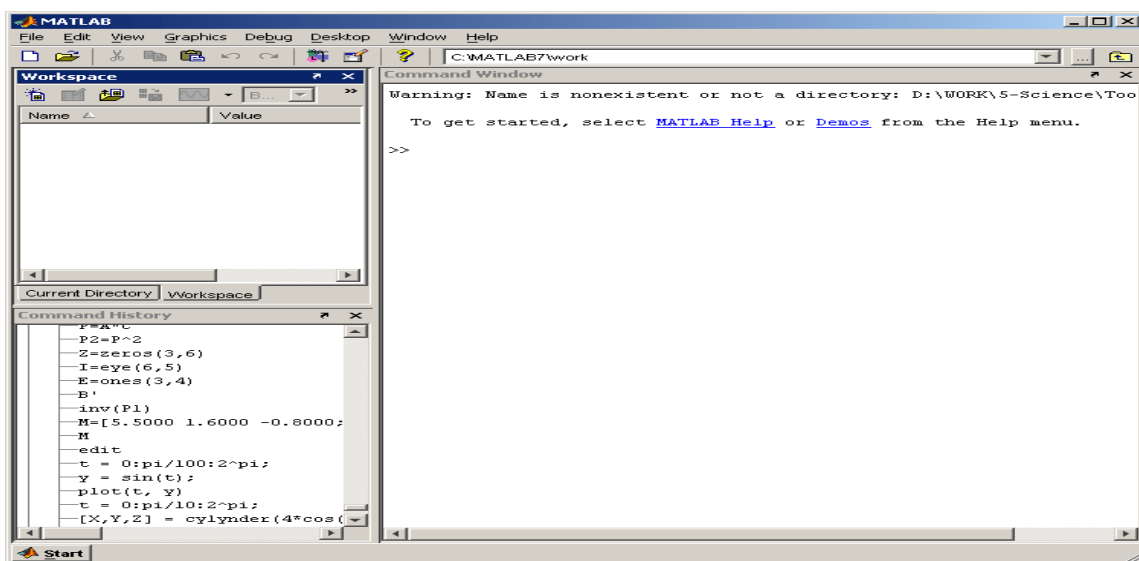
Reja:

1. Matlab muhiti bilan tanishib chiqing.
2. Matlab integrallashgan sohasini o'rganish.
3. Grafika hosil qilish usullarini o'rganish

Matlabni yuklash uchun “Пуск” tugmachasi yordamida “Приложения” bandiga kiriladi va dasturlar ro'yxatidan Matlab dasturi tanlanadi yoki ishchi stolda



Matlab belgisi **MATLAB** ustida sichqoncha ko'rsatkichini ikki marta bosish kerak. Natijada quyidagi darcha hosil bo'ladi.



13-rasm. Matlab dasturining umumiy ko'rinishi

Matlab dasturi quyidagi qismlardan iborat:

1. Sarlavha qatori;
2. Asosiy menyular qatori (File, Edit, View, Graphics, Debug, Desktop, Window, Help);
3. Qo'shimcha amallarni bajarish uchun mo'ljallangan maxsus uskunalar paneli (piktogrammalar);
4. "Current Directory" va "Workspace" darchalari ("Toolbox"da joylashgan modular bilan ishlash imkonini beradi);
5. "Command History" darchasi (oldin ishlatilgan operatorlarni ko'rish va ulardan foydalanish imkonini beradi);
6. Ishi maydon;
7. "Start" bo'limi.

MATLAB tizimi shunday ishlab chiqilganki, hisoblashlarni, foydalanuvchi dasturini tayyorlamasdan to'g'ridan-to'g'ri bajarish mumkin. Bunda Matlab superkalkulyator vazifasini bajarib, qatorli komanda rejimida ishlaydi. Masalan, $\gg 2+3$, $\text{ans}=5$; $\gg 2*3$, $\text{ans}=6$ va hokazo.

Tizimda ishlash muloqotli (dialogli) tavsifga ega bo'lib, "savol berildi – javob olindi" qoidasi bo'yicha ishlanadi. Ya'ni foydalanuvchi klaviatura yordamida hisoblanishi lozim bo'lgan ifodani kiritadi, tahrir qiladi (agar lozim bo'lsa) va kiritishni ENTER klaviaturasini bosish bilan yakunlaydi.

Umuman olganda, ma'lumotlarni kiritish va hisoblashlarni amalga oshirish quyidagicha amalga oshiriladi:

- Boshlang'ich ma'lumotlarni kiritishni ko'rsatish uchun \gg belgidan foydalaniladi;
- Ma'lumotlar oddiy yozuvli tahrir yordamida kiritiladi;
- Biror bir ifoda hisoblash natijasini blokirovka qilish uchun mazkur ifodadan keyin - ; (nuqta vergul) qo'yiladi;
- Hisoblashlar natijasini ko'rsatuvchi o'zgaruvchi aniqlanmagan bo'lsa, u holda Matlab tizimi bunday o'zgaruvchi deb *ans* oladi;
- O'zlashtirish amali sifatida juda ko'plab dasturlash tillari kabi : = belgi emas, balki matematikadagi oddiy = ni o'zi olinadi;
- Sozlangan funksiyalar (masalan, sin) yozma harflar bilan yoziladi hamda ularning argumentlari oddiy qavslar ichida yoziladi;
- Hisoblashlar natijasi yangi qatorda \gg belgisiz chiqadi;
- Muloqot "Savol berildi – javob olindi" ko'rinishida amalga oshadi.

Ma'lumki, juda ko'plab matematik tizimlarda, agar u son bo'lmasa, u holda $\sin(v)$ va $\text{exr}(v)$ ifodalarni hisoblab bo'lmaydi, ya'ni tizim bunday ifodalarni xato deb beradi. Matlabda esa agar berilgan o'zgaruvchi ektor bo'lsa, natija ham mazkur o'lchamdagi ektor bo'ladi, agar matritsa bo'lsa, natija ham matritsa bo'ladi.

Komandali rejimda bir qatordagi belgilarning maksimal soni – 4096, m – fayllarda esa chegaralanmagan. Barcha matematik tizimlarning markaziy tushunchasi bu matematik ifodalardir. Ma'lumki, ular ustida amallar bajarilayotganda, asosan ularning sonli qiymatlaridan foydalaniladi (kam holatlarda belgi ko'rinishlaridan ham foydalaniladi).

Matlab ham matematik tizim bo'lgani uchun bu yerda ham asosiy tushuncha matematik ifodalardir. Matlabda matematik ifodalarni ifodalashni qarab chiqaylik. Matlabda ifodalar bir qator ko'rinishida ifodalanib, sonlarni butun qismlarini ajratish uchun verguldan emas balki nuqtalardan foydalaniladi. Quyida ba'zi bir ifodalarni Matlab va oddiy matematikadagi ifodalanishini ko'rib chiqamiz:

2-jadval

Matlabda	Matematikada
2+3	2+3;
2^3*sqrt(y)/2	23√y/2
2.301*sin(x);	2,301sin(x)
4+exp(3)/5;	4+e3/5

Matematik ifodalar sonlar, konstantalar, o'zgaruvchilar, operatorlar, funksiyalar va turli xil maxsus belgilar ustiga quriladi. Ilgari aytib o'tganimizdek, nuqta vergul, ya'ni ; belgi natijani chiqishini blokirovka qiladi, ammo *ans* maxsus o'zgaruvchi yordamida natijani olishimiz mumkin. Son – Matlab tilining eng oddiy ob'yektlaridan biri bo'lib, u miqdoriy ma'lumotlarni ifodalab beradi. Sonlarni konstanta deb hisoblash mumkin. Sonlar butun, kasr, fiksirlangan va suzuvchi nuqtali bo'lishi mumkin. Ularni yaxshi ma'lum bo'lgan ilmiy shaklda, ya'ni mantissa va son tartibini ko'rsatgan holda ifodalash mumkin.

0
-3
2.301
123.456e-24
-234.456e10

Yuqoridan ko'rinib turibdiki, mantissadan sonning butun qismi kasr qismidan, juda ko'plab dasturlash tillarida qabul qilinganidek, vergul orqali emas, balki nuqta orqali ajratiladi. Son tartibini mantissadan ajratish uchun ular orasiga e belgisi qo'yiladi. "+" ishora sonlar oldiga qo'yilmaydi, "-" ishora esa qo'yiladi va uni unar minus deb nomlanadi. Sonlarda belgilar orasiga probel (bo'sh joy) qo'yish ruxsat etilmaydi.

Bundan tashqari sonlar kompleks bo'lishi mumkin: $z = \text{Re}(z) + \text{Im}(z) \cdot i$. Bunday sonlar $\text{Re}(z)$ haqiqiy va $\text{Im}(z)$ mavhum qismga ega bo'linadilar. mavhum qism kvadrat darajasi -1 ga teng bo'lgan, i va j ko'paytuvchilarga ega bo'ladi:

3i
2j
2+3i
-3.141i
-123.456+2.7e-3i

real (z) funksiya kompleks sonning butun qismini, $\text{image}(z)$ – esa mavhum qismini ajratib beradi. Kompleks sonning modulini (kattaligini) $\text{abs}(z)$ funksiya, fazasini $\text{angle}(z)$ funksiya hisoblab beradi. Masalan:

```

>> i
Ans=0+1.000i
>>z=2+3i
Z=2.000+3.000i
>>abs(z)
Ans=3.6056
>>real(z)
Ans=2
>>Imag(z)
Ans=3
>>angle(z)
Ans=0.9828

```

Matlab dasturlash tilida o'zgaruvchiga qiymat berish: <o'zgaruvchi nomi> = < ifoda qiymati > komandasi yordamida amalga oshiriladi. Bu erda (=) tayinlash (qiymat berish) operatori vazifasini bajaradi.

Masalan, >> x= 5+exr (3);

Matlabning yaxshi xususiyatlaridan biri shuki, unda avvaldan o'zgaruvchini turi e'lon qilinmaydi, balki uni qiymatlariga qarab aniqlanaveradi. Demak ifoda qiymati vektor yoki matritsa bo'lsa, u holda o'zgaruvchi shunga mos bo'ladi.

O'zgaruvchi nomi (identifikator) – boshlanishi harfdan iborat ixtiyoriy sondagi belgilardan tashkil topgan bo'lishi mumkin, ammo faqat boshidagi 31tasi orqali identifikatsiya qilinadi. O'zgaruvchi nomi boshqa o'zgaruvchilar bilan ustma-ust tushmasligi kerak, ya'ni nom noyob bo'lishi lozim. O'zgaruvchi nomi harfdan boshlangan bo'lsada, orasida raqamlar va belgidan (подчеркивания) iborat bo'lishi mumkin. Lekin ularning orasiga maxsus belgilar, masalan +, -, *, / va boshqalarni qo'yish mumkin emas.

Matlabda ma'lumotlar ustida bajariladigan ma'lum bir amalni bajarish uchun ishlatiladigan belgi operator deyiladi. Masalan, oddiy arifmetik amallar +, -, *, / - operatorlarga misol bo'ladi. Bu amallar (1*1) o'lchovlidan yuqori bo'lgan matritsalar ustida bajarilsa va natija ham matritsa bo'lsa, u holda amallar elementlararo bajariladi va * amali. *, / esa./, /. kabi belgilab amalga oshiriladi.

Masalan:

```

>> x= [2 4 6 8]
x= 2 4 6 8
>> u= [1 2 3 4]
u= 1 2 3 4
>> x/u
ans= 2
>> x.*u
ans= 2 8 18 32
>> x./u
ans= 2 2 2 2.

```

Matlabdagi barcha operatorlar ro'yxatini ko'rish uchun help ops komandasidan foydalaniladi.

Ikki o'lovli grafika. Matlab tizimining eng katta xususiyatlaridan biri, unda grafik chizish imkoniyatining mavjudligidir. Biz Matlabda ikki vektor grafigini chizishning eng sodda va umumiy komandalari bilan tanishamiz.

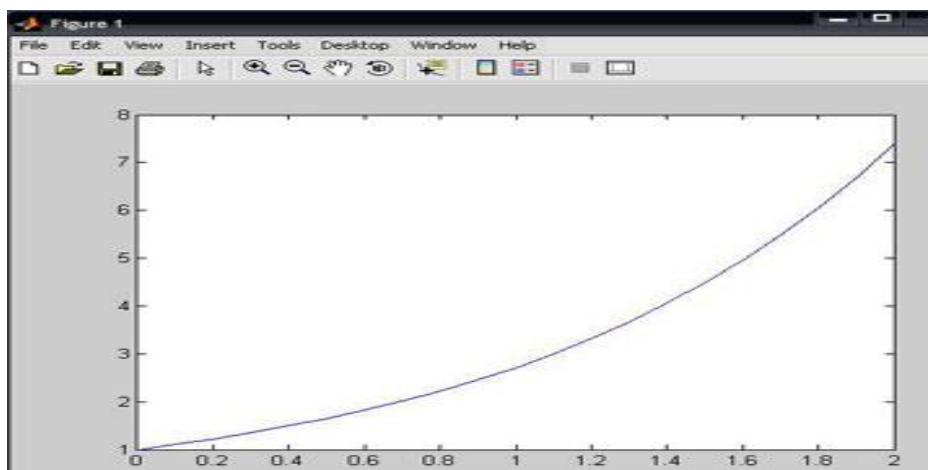
Matlabda grafiglarni har xil koordinata sistemalarida qurish mumkin. Bulardan to'g'ri burchakli dekart koordinatalari sistemasi, polyar koordinatalari, sferik va silindrik sistemalarni keltirish mumkin. Undan tashqari koordinatalarni bir sistemadagi ko'rinishidan boshqa ko'rinishga o'tkazish mumkin.

Biror bir sistemada grafik chizish uchun umumiy bo'lgan ba'zi grafik chizish komandalarini keltiramiz:

- `plot(x,y)`-x va y vektorlarning dekart tekisligidagi grafigini hosil qiladi;
- `plot(y)`-y ning y -vektor elementlari nomerlarga nisbatan grafigini yasaydi;
- `semilogx(x,y)`- "x"ning logarifmi grafigini "y" ga nisbatan yasaydi;
- `semilogy(x,y)`- "x"ning grafigini "y" ning logarifmiga nisbatan yasaydi;
- `loglog(x,y)`- "x"ning logarifmini "y" ning logarifmiga nisbatan grafigini yasaydi;
- `grid` -koordinatalar sistemasida to'rni hosil qiladi;
- `title ('matn')`- grafik tepasiga matn yozadi;
- `xlabel ('matn')`- "matn"ni "x" o'qi ostiga yozadi;
- `ylabel ('matn')`- "matn"ni "y" o'qining chap tomoniga yozadi;
- `text (x,y,'matn')`- "matn"ni (x, y) nuqtaga yozadi;
- `polar(theta, r)`- r va theta vektorlarning polyar koordinatalar sistemasida grafigini yasaydi (bu yerda theta faqat radianlarda beriladi);
- `bar(x)` yoki `stairs(x)`- "x" vektorning gistogrammasini yasaydi;
- `bar(x,y)` yoki `stairs(x,y)`- "u" vektor elementlarini gistogrammasini "x" vektorning elementlariga mos to'plamga joylashtirib chizadi;

Ma'lumki, dekart koordinatalar sistemasida grafik chizish (x, y) juftligining qiymatlarini aniqlab, hosil bo'lgan nuqtalarni kesmalar bilan tutashtirish orqali hosil qilinadi. Demak (x, y) juftliklar soni qanchalik ko'p bo'lsa, grafik ham shunchalik silliq va aniqroq bo'ladi. Juftliklar avvaldan berilgan bo'lishi yoki ma'lum funksiyaning argumenti va qiymatlaridan hisoblab hosil qilinishi yoki tajriba o'tkazish natijasida olingan bo'lishi mumkin. Masalan, $y=e^x$ funksiyaning $x \in [0,2]$ sigmentdagi grafigini chizish kerak bo'lsa, quyidagi matlab komandalari ketma-ketligi yetarli bo'ladi:

```
>> x=0:.1:2;  
>> y=exp(x);  
>> plot(x,y)
```



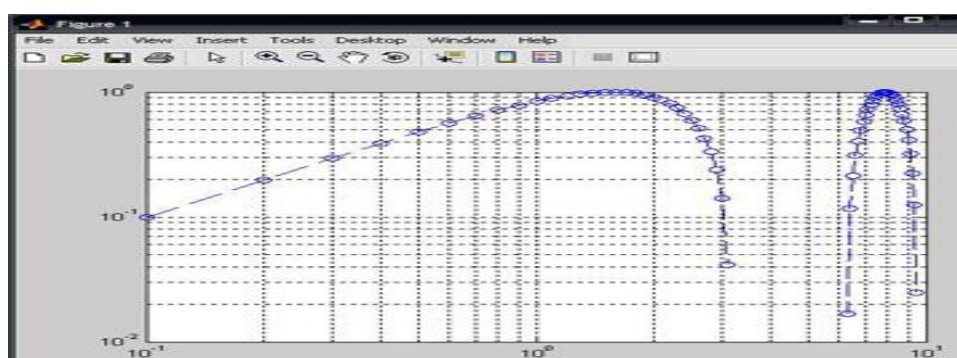
14-rasm. $y=e^x$ funksiya grafigi

plot(x,y) - komandasi grafik oynani ochadi va unda kerakli funksiya grafigini chizib beradi. Yangi komandani e'lon qilish uchun kursorni komandalar oynasiga o'tkazishimiz kerak. Grafik oyna qayta chizmaslik uchun har bir komandadan keyin uch nuqta (...) qatorning davomi belgisini ishlatish mumkin.

```
>> plot(x,y)...
>> grid,...
>> title('ko'rsatkichli funksiya'),...
>> xlabel('x'),...
>> ylabel('exp(x)'),...
```

Ko'pincha grafik komandalar M-faylga joylashtiriladi (Ishchi fayl yoki fayl funksiyalar). Bu usul xatoliklarni to'g'rilash uchun yaxshi imkoniyat beradi. Yana quyidagi misollarni ko'raylik:

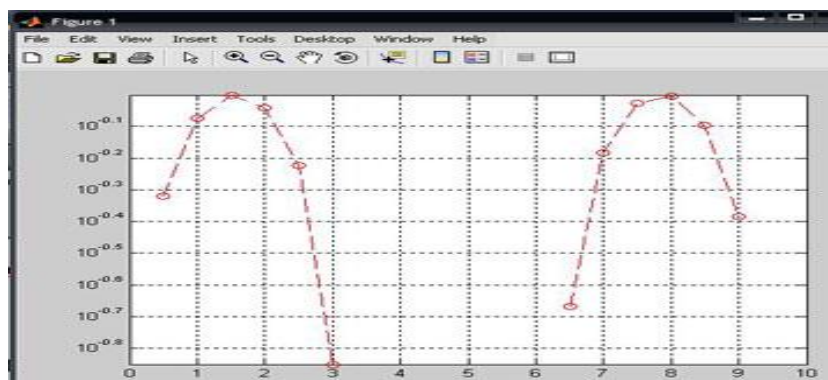
```
% x ning logarifmini sin(x) ning logarifmiga nisbatan chizilgan grafigi
x=0:1:10;log(x,sin(x),'--ob');grid on
```



15-rasm. Funksiya grafigi

Bu yerda '--' -liniya turi, 'o'-aylana tugun nuqta turi, 'b' - havorang liniya rangi. Endi boshqa grafik funksiyadan foydalanib ko'ramiz:

```
>> x=0:0.5:10;
>> semilogy(x,sin(x),'--or')
>> grid
```



16-rasm. Funksiya grafigi

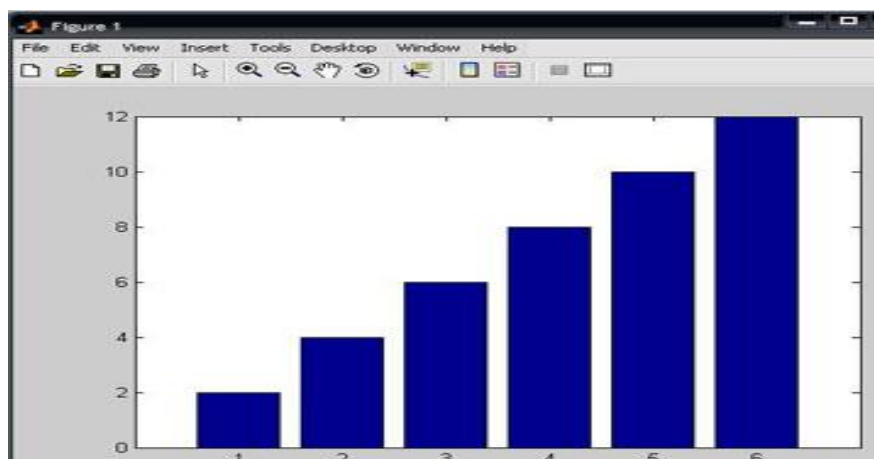
Bu misollardan ko‘rinib turibdiki, matlab tizimida grafik chiziqlarining rangini, turini, tugun nuqtalarini ko‘rsatish va boshqa imkoniyatlar mavjud.

2. Gistogrammalar. Polyar koordinatalarda grafika. Amaliy hisoblarda biror vektor tarkibini tasvirlaydigan ustunli diagrammalar deb ataluvchi gistogrammalar ko‘p uchraydi. Bunda vektorning har bir elementi balandligi uning qiymatiga mos bo‘lgan ustun shaklida ko‘rsatiladi. Ustunlar tartib raqamlariga va eng baland ustunning maksimal qiymatiga nisbatan ma‘lum masshtabga ega bo‘ladi. Bunday grafiklar masalan, iqtisodiy o‘zgarish va boshqa jarayonlarni ifodalashi mumkin. Ular `bar(a)` komandasi yordamida quriladi, masalan:

```
>> a=[2 4 6 8 10 12];
```

```
>> bar(a)
```

komandalari yordamida quyidagi gistogrammani olish mumkin:



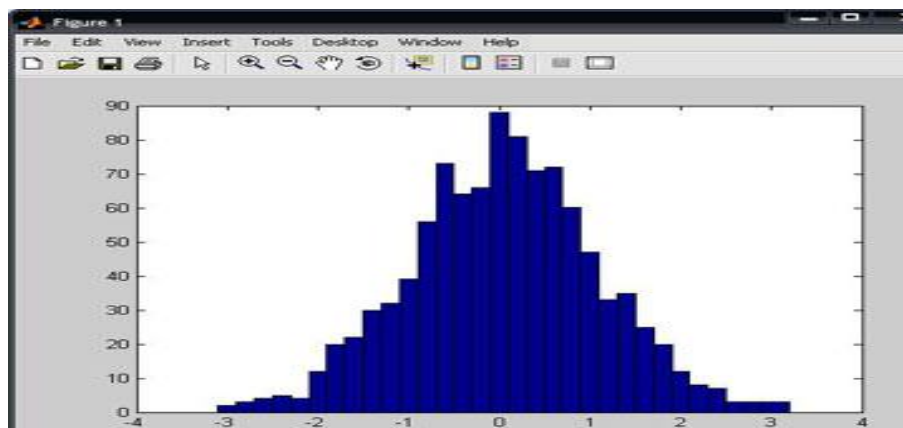
17-rasm. Gistogrammadan hosil qilish

Bundan tashqari gistogramma qurishning yana boshqa usuli ham mavjud bo‘lib, bu hist funksiyasi yordamida amalga oshiriladi:

- $N=\text{hist}(u)$ - avtomatik tanlangan 10 intervalli vektor qiymatini qaytaradi;
- $N=\text{hist}(u,m)$ -huddi yuqoridagi kabi, faqat M (M -skalyar) intarvalda qaytaradi;

Quyidagi misolni ko'ramiz:

```
>> x=-3:0.2:3; y=randn(1000,1);  
>> hist(y,x); h=hist(y,x)  
h =  
Columns 1 through 13  
2 3 4 5 4 12 20 22 30 32 39 56 73  
Columns 14 through 26  
64 66 88 81 71 72 60 47 33 35 25 20 12  
Columns 27 through 31  
8 7 3 3 3  
>>
```



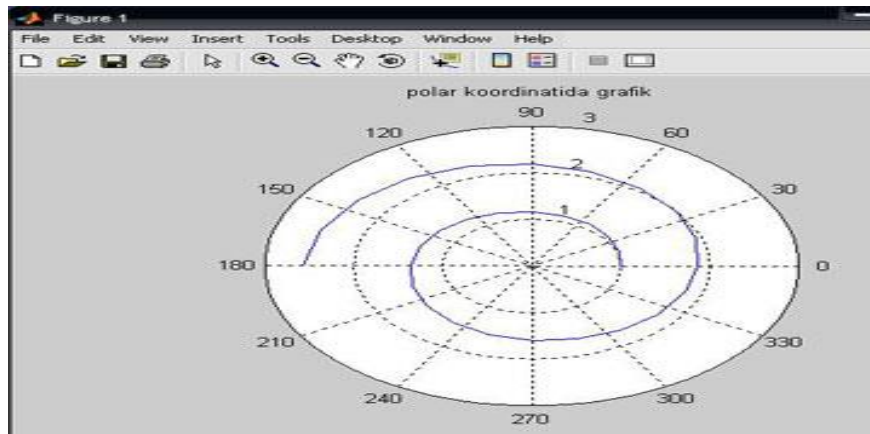
18-rasm. Gistogrammani misolda foydalanish

Qutbli koordinatalar tizimida ixtiyoriy nuqta xuddi radius vektor oxiri kabi, koordinatalar tizimining boshlang'ich nuqtasidan chiqib, RHO uzunlikka va THETA burchakka egaligini ko'rsatadi. RHO(THETA) funksiya grafigini qurish uchun quyida keltirilgan buyruqlardan foydalaniladi. THETA burchak odatda 0 dan $2 * \pi$ gacha o'zgaradi. Qutbli koordinatalar tizimida funksiya grafigini qurish uchun quyidagi buyruqlardan foydalaniladi :

- polar(THETA,RHO)- qutbli koordinatalar tizimida radius-vektor oxirining o'z holatidagi RHO uzunlik bilan va THETA burchakni ko'rsatuvchi grafikani quradi;
- polar(THETA,RHO, S)- analogli avvalgi buyruqda ishtirok etgan, lekin S qatorli konstanta yordamida qurish uslubini analogli plot buyrug'i asosida ruxsat beradi.

Quyidagi misolni ko'ramiz:

```
>> angle=0:.1*pi:3*pi;  
>> r=exp(angle/10);  
>> polar(angle,r),...  
>> polar(angle,r);  
>> title('polyar koordinatida grafik');  
>> grid on
```

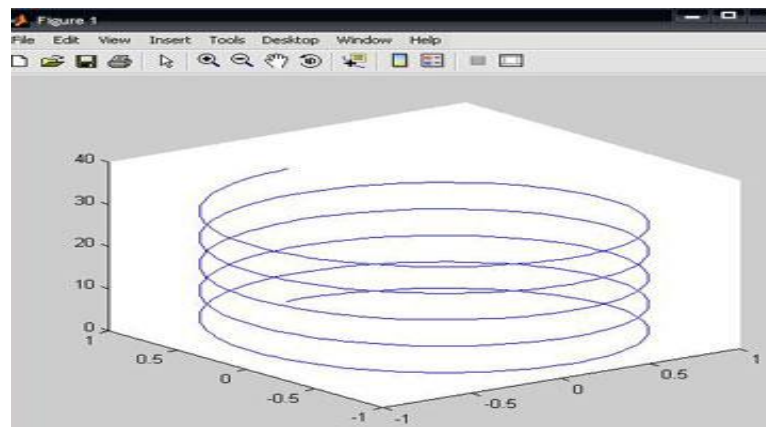


19-rasm. Polyar koordinatada grafik

Uch o‘lchovli grafika. Grafik chizishga doir misollar. Uch o‘lchovli fazoda grafik chizish uchun `plot3(x,y,z)` komandasidan foydalaniladi. Bunda x,y,z -vektorlar bir xil sondagi koordinatalarga ega bo‘lishi kerak, aks holda sistema xatolikni beradi.

Masalan,

```
>> t=0:pi/50:10*pi;
>> plot3(sin(t),cos(t),t)
```



20-rasm. Uch o‘lchovli grafika.

Demak, `plot3` komandasi yordamida uch o‘lchovli fazoda chiziqning grafigini xosil qilish mumkin.

Bundan tashqari uch o‘lchovli fazoda sirlarning grafigini hosil qiluvchi quyidagi komandalar mavjud:

- `mesh`-bu fazoda uch o‘lchovli “to‘r”ni chizadi;
- `surf`-fazoda uch o‘lchovli sirtini chizadi;
- `fill3`-fazoda uch o‘lchovli to‘ldirilgan ko‘pburchakni chizadi.

Nazorat savollari:

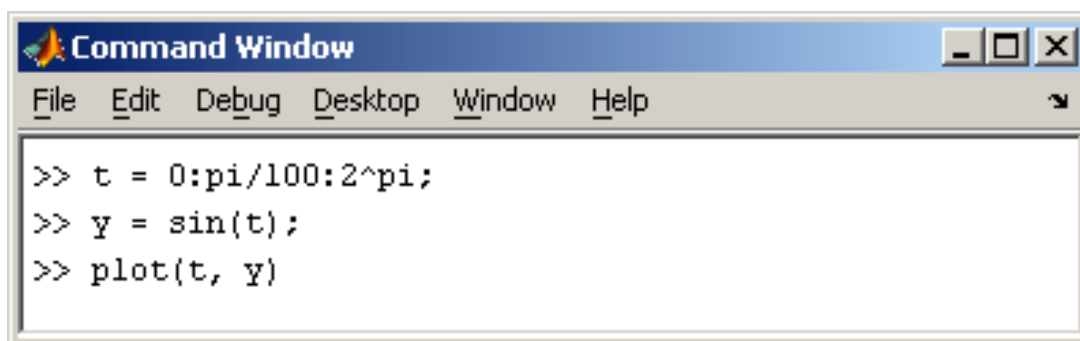
1. Matlab integrallashgan sohasi nimalardan tashkil topgan.
2. Matlab dasturida grafika bilan ishlash qanday amalga oshiriladi.

Matlab dasturi quyidagi qismlardan iborat:

1. Sarlavha qatori;
2. Asosiy menyular qatori (File, Edit, View, Graphics, Debug, Desktop, Window, Help);
3. Qo‘shimcha amallarni bajarish uchun mo‘ljallangan maxsus uskunalar paneli (piktogrammalar);
4. “Current Directory” va “Workspace” darchalari (“Toolbox”da joylashgan modular bilan ishlash imkonini beradi);
5. “Command History” darchasi (oldin ishlatilgan operatorlarni ko‘rish va ulardan foydalanish imkonini beradi);
6. Ishi maydon;
7. “Start” bo‘limi.

Matlab dasturida grafika bilan ishlash. Matlab sirtlar, chiziqlar va boshqa grafik obyektlarni o‘zlashtirish va yaratish imkonini beruvchi yuqori darajadagi funksiyalar majmuasini taqdim qiladi.

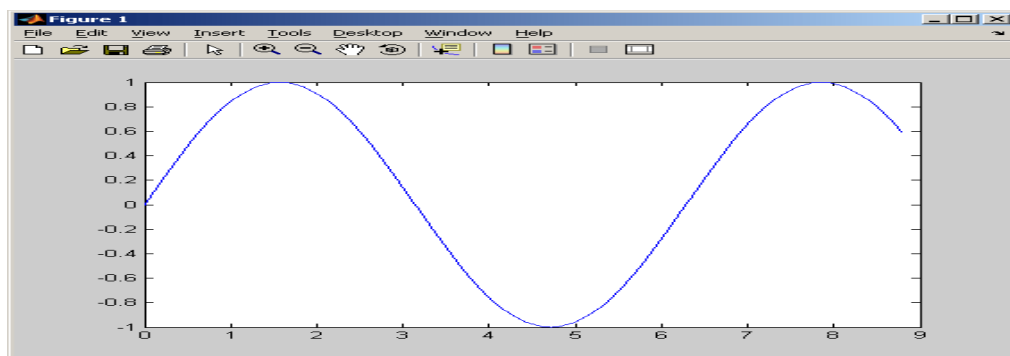
Grafikni chizish uchun *plot* ifodasidan foydalaniladi. Misol sifatida sin funksiyaning grafigini chizishni olamiz. t argument bo‘yicha 0 dan 2π gacha oraliqda 0.03 qadam o‘zgarib borishini hisoblash grafigini hosil qilamiz. Grafikni tasvirlash uchun quyidagi ifodani yozamiz:



```
>> t = 0:pi/100:2*pi;
>> y = sin(t);
>> plot(t, y)
```

22-rasm. Dastur matni

Operatorlar ketma – ket kiritilgandan so‘ng “enter” tugmasi bosiladi. Natija grafik alohida darchada (figure 1) paydo bo‘ladi va quyidagi ko‘rinishga ega bo‘ladi.



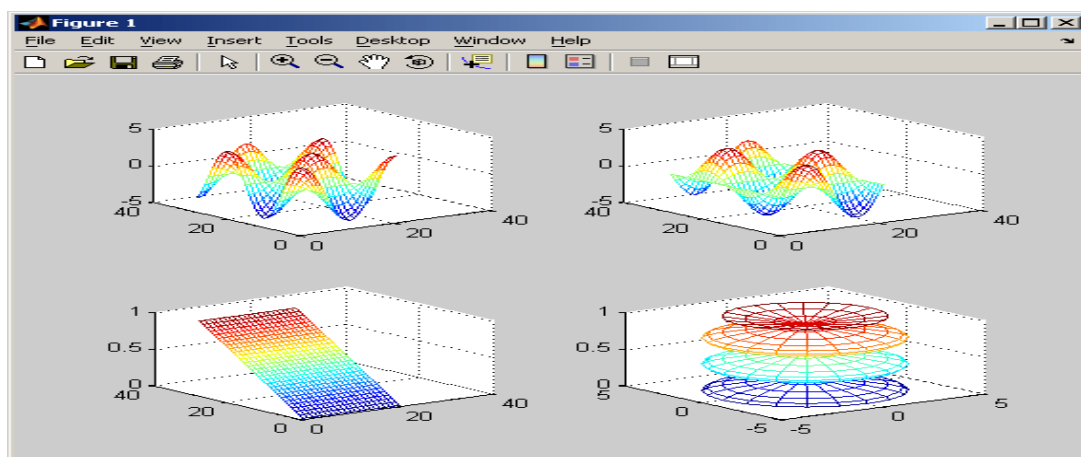
23-rasm. Sinus funksiyani grafigini hosil qilish

Matlab dasturida 3 o‘lchovli grafikni hosil qilish. Dasturda masalani yechish, ya’ni sitrlarni tasvirlashda 3 o‘lchovli grafiklarni hosil qilishga to‘g‘ri keladi. Buning uchun quyidagi buyrug‘larni kiritish kerak bo‘ladi:

```
Command Window
File Edit Debug Desktop Window Help
>> t = 0:pi/10:2*pi;
>> [X,Y,Z] = cylinder(4*cos(t));
>> subplot(2,2,1)
>> mesh(X)
>> subplot(2,2,2); mesh(Y)
>> subplot(2,2,3); mesh(Z)
>> subplot(2,2,4); mesh(X,Y,Z)
>> |
```

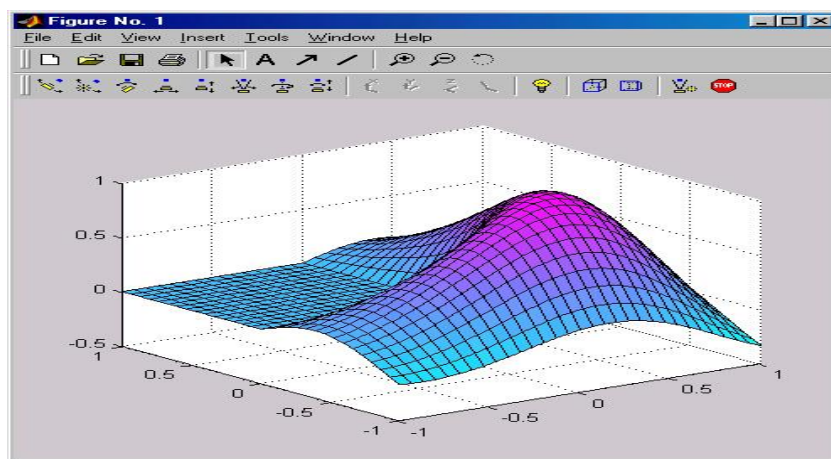
24-rasm. Dastur matnini kiritish darchasi

Natijada quyidagi 3 o‘lchovli grafik hosil bo‘ladi.




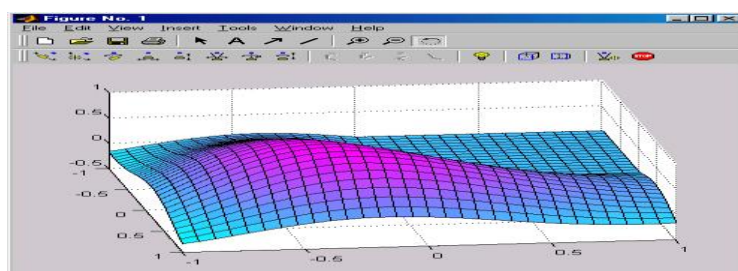
25-rasm. 3 o‘lchovli grafikni hosil qilish

Grafiklarni sichqoncha yordamida harakatlantirish. Figuralarni sichqoncha yordamida har xil burchaklarda harakatlantirish mumkin. Buning uchun Matlab dasturining logotip membranasidan foydalanamiz. Ishchi maydonga *membrane* buyrug‘ini kiritamiz. Natijada quyidagi darcha hosil bo‘ladi.



26-rasm. Matlab tizimi logotipining membranasasi

Grafikni harakatga keltirish uchun uskunalar panelidan  belgini aktivlashtirish kerak bo'ladi. Belgi aktivlashgandan so'ng sichqonchanning chap tugmasini bosib turgan holda kerakli tomonga aylantirish mumkin bo'ladi.



27-rasm. 3 o'lchovli grafikani sichqoncha yordamida aylantirish

Matlab tizimi operatorlari. Matlab tizimining operatorlari uchta kategoriyaga bo'linadi:

- arifmetik operatorlar - hisoblashlarni bajaradi va arifmetik ifodalarni tuzish imkonini beradi;
- munosabat operatorlari - sonli operandlarni solishtirish imkonini beradi;
- mantiqiy operatorlar - mantiqiy ifodalarni tuzish imkonini beradi.

Matlab tizimining arifmetik operatorlari bir xil o'lchovli massivlar bilan ishlaydi. Vektorlar va to'g'ri to'rtburchakli massivlar uchun ikkala operand ham bir xil o'lchovga ega bo'lishi kerak (skalyarlar bundan mutsasno).

Munosabat operatorlari: $<>$ $<=$ $>=$.

Munosabat operatorlari teng o'lchovli ikkita massivni elementlari bo'yicha solishtirishni bajaradi.

Oqimlarni boshqarish. Matlab oqimlarni boshqarish tarkiblarining 5 ta ko'rinishini o'z ichiga oladi:

- if operatori;
- switch operatori;
- for takrorlanishi;
- while takrorlanishi;

- break operatori.

if operatori - agar ifoda rost qiymat qabul qilsa, mantiqiy ifodani hisoblaydi va operatorlar guruhini bajaradi.

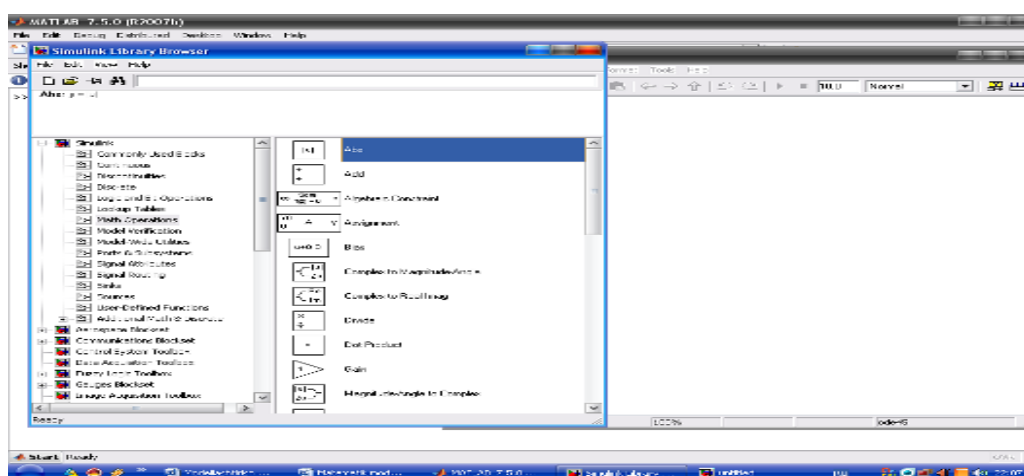
switch operatori - ifoda yoki o'zgaruvchilar qiymatida operatorlar guruhini bajaradi.

for takrorlanishi - operatorlar guruhini oldindan qayd qilingan son marta takrorlaydi.

while takrorlanishi - mantiqiy shart bajarilganga qadar operatorlar guruhini ma'lum son marta takrorlaydi.

break operatori - for yoki while takrorlanishidan chiqib ketishni ta'minlaydi.

Simulink bibliotekasini ishga tushirish. Uning ish stoli quyidagi ko'rinishda bo'ladi.



28-rasm. Simulink darchasi

Simulink bibliotekasi ba'zi asosiy bo'limlari:

- Continuous – chiziqli bloklar
- Discrete – diskret bloklar
- Math Operations – matematik amallar bloklari
- Sinks – registratsiya (ruyxat) qiluvchi qurilmalar
- Sources – signallar va ta'sirlar manbalari
- Nonlinear – chiziqli bo'lmagan bloklar
- Subsystems – ost tizim bloklari

Ushbu berilgan chiziqli tenglamalar sistemasini yechish blok-modeli quyidagi ketma-ketlikda amalga oshiriladi:

$$\begin{cases} 3x + y - 2 = 0 \\ x - 2y - 3 = 0 \end{cases}$$

Simulink Library Browser - **-File - -New - -Model** - yangi varak (model) ochiladi.

Simulink – qatoridan kerakli blok operatorlarni tanlab olish va yangi ochilgan varaqqa o'tkazish.

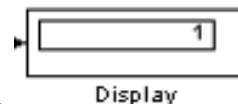
Dastlab *Math Operations* – blogidan  *Solve Algebraic Constraint* va *Gain* operatori tanlab olinadi. Undan so‘ng xuddi shu blokdan *Add*



operatori olinadi.

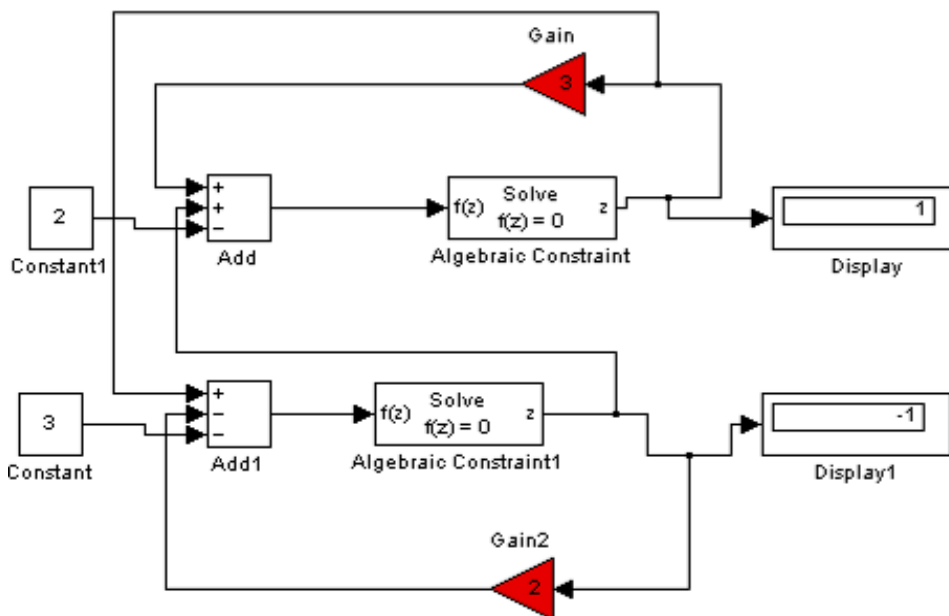
O‘zgarmas qiymatlarni *Sources* blogidan *Constant* operatori yordamida kiritiladi.

Signallarni kuchaytirish operatori *Gain* yordamida kerakli qiymatlarni kuchaytirish yoki kamaytirish mumkin.



So‘ngra *Skins* blogidan *Display* operatori tanlab olinadi .

Barcha operatorlar kerakli yo‘nalishdagi signal chiziqlar orqali bir biri bilan tutashtiriladi va yopiq kontur hosil qilinadi.



29-rasm. Natija darchasi

Barcha kontur yopilgandan so‘ng *Star Simulation* bosiladi va displeyda **x,u** qiymatlari paydo bo‘ladi.

Nazorat savollari:

1. Matlab integrallashgan sohasi nimalardan tashkil topgan.
2. Matlab dasturida grafika bilan ishlash qanday amalga oshiriladi.
3. Matlab dasturida 3 o‘lchovli grafikani hosil qilish xususiyatlari.

5-AMALIY MASHG'ULOT

KIBERXAVFSIZLIKDA KIPTOGRAFIK USULLARIDAN FOYDALANISH

Ishdan maqsad: Kriptografik usullar bilan tanishish va ularning amaliy tatbiq qilish texnologiyasini o'rganish.

Reja:

1. Kriptografiya asoslari bilan tanishish.
2. Shifrlash usullari.
3. Sezar usulida shifrlash.

Kriptografiya deb, maxfiy xabar mazmunini shifrlash, ya'ni ma'lumotlarni maxsus algoritm bo'yicha o'zgartirib, shifrlangan matnni yaratish yo'li bilan axborotga ruxsat etilmagan kirishga to'siq qo'yish usuliga aytiladi.

Kalit – kriptografiya o'zgartirishlar algoritmining ba'zi bir parametrlarining manfiy faoliyati bo'lib, barcha algoritmlardan yagona variantini tanlaydi. Kalitlarga nisbatan ishlatiladigan asosiy ko'rsatkich bo'lib kriptobardoshlilik hisoblanadi.

- Kriptografiya himoyasida shifrlarga nisbatan quyidagi talablar qo'yiladi:
- yetarli darajada kriptobardoshlilik;
- shifrlash va qaytarish jarayonining oddiyligi;
- axborotlarni shifrlash oqibatida ular hajmining ortib ketmasligi;
- shifrlashdagi kichik xatolarga ta'sirchan bo'lmasligi.
- Ushbu talablarga quyidagi tizimlar javob beradi:
- o'rinlarini almashtirish;
- almashtirish;
- gammalashtirish;
- analitik o'zgartirish.

O'rinlarini almashtirish shifrlash usuli bo'yicha boshlang'ich matn belgilarining matnning ma'lum bir qismi doirasida maxsus qoidalar yordamida o'rinlari almashtiriladi.

Almashtirish shifrlash usuli bo'yicha boshlang'ich matn belgilari foydalanilayotgan yoki boshqa bir alifbo belgilariga almashtiriladi.

Gammalashtirish usuli bo'yicha boshlang'ich matn belgilari shifrlash gammasi belgilari, ya'ni tasodifiy belgilar ketma-ketligi bilan birlashtiriladi.

Taxliliy o'zgartirish usuli bo'yicha boshlang'ich matn belgilari analitik formulalar yordamida o'zgartiriladi, masalan, vektorni matritsaga ko'paytirish yordamida. Bu erda vektor matndagi belgilar ketma-ketligi bo'lsa, matritsa esa kalit sifatida xizmat qiladi.

O'rinlarni almashtirish usullari. Ushbu usul eng oddiy va eng qadimiy usuldir. O'rinlarni almashtirish usullariga misol sifatida quyidagilarni keltirish mumkin:

- shifrovchi jadval;
- sehrlı kvadrat.
- Shifrovchi jadval usulida kalit sifatida quyidagilar qo'llaniladi:
- jadval o'lchovlari;
- so'z yoki so'zlar ketma-ketligi;
- jadval tarkibi xususiyatlari.

1-mashq. Shifrovchi jadval usuli yordamida matnni shifrlang.

Quyidagi matn berilgan bo'lsin:

АХБОРОТ_ТЕХНОЛОГИЯЛАРИНИ_РИВОЖЛАНТИРИШ

Ushbu axborot ustun bo'yicha ketma – ket jadvalga kiritiladi:

3-jadval

А	О	Е	О	А	_	Ж	И
Х	Т	Х	Г	Р	Р	Л	Р
Б	_	Н	И	И	И	А	И
О	Т	О	Я	Н	В	Н	Ш
Р	Е	Л	Л	И	О	Т	*

Natijada, 5x8 o'lchovli jadval tashkil qilinadi.

Endi shifrlangan matn qatorlar bo'yicha aniqlanadi, ya'ni o'zimiz uchun 4 tadan belgilarni ajratib yozamiz.

$T_1 = \text{АОЕОА_ЖИ ХТХГРРЛР Б_НИИИАИ ОТОЯНВНШ РЕЛЛИОТ*}$

Bu yerda kalit sifatida jadval o'lchovlari xizmat qiladi.

Hozirgi vaqtda kompyuter tarmoqlarida tijorat axborotlari bilan almashishda uchta asosiy algoritmlar, ya'ni DES, CLIPPER va PGP algoritmlari qo'llanilmoqda. DES va CLIPPER algoritmlari integral sxemalarda amalga oshiriladi. DES algoritmining kriptobardoshligini quyidagi misol orqali ham baholash mumkin: 10 mln. AQSh dollari xarajat qilinganda DES shifrlash ochish uchun 21 minut, 100 mln AQSh dollari xarajat qilinganda esa 2 minut sarflanadi. CLIPPER tizimi SKIPJACK shifrlash algoritmini o'z ichiga oladi va bu algoritm DES algoritmidan 16 mln marta kuchliroqdir.

Sehrlı kvadrat deb, katakchalariga 1 dan boshlab sonlar yozilgan, undagi har bir ustun, satr va diagonal bo'yicha sonlar yig'indisi bitga songa teng bo'lgan kvadrat shaklidagi jadvalga aytiladi.

Sehrlı kvadratga sonlar tartibi bo'yicha belgilar kiritiladi va bu belgilar satrlar bo'yicha o'qilganda matn hosil bo'ladi.

2-mashq. 4x4 o'lchovli sehrli kvadratni olamiz, bu yerda sonlarning 880 ta har xil kombinatsiyasi mavjud. quyidagicha ish yuritamiz:

4-jadval

4x4 o'lchovli sehrli kvadrat

4	14	15	1
9	7	6	12
5	11	10	8
16	2	3	13

Boshlang'ich matn sifatida quyidagi matnni olamiz:

DASTURIY TILLAR

va jadvalga joylashtiramiz:

DASTURIY TILLAR

T	A	P	Д
_	И	P	Л
Y	И	T	Й
*	A	C	Л

Shifrlangan matn jadval elementlarini satrlar bo'yicha o'qish natijasida tashkil topadi:

TARD _IRL UITY *ASL

Almashtirish usullari

Almashtirish usullari sifatida quyidagi usullarni keltirish mumkin:

- Sezar usuli;
- Affin tizimidagi Sezar usuli;
- Tayanch so'zli Sezar usuli va boshqalar.

Ssezar usulida almashtiruvchi harflar k va siljish bilan aniqlanadi. Yuliy Sezar bevosita k q 3 bo'lganda ushbu usuldan foydalangan.

k = 3 bo'lganda va alifbodagi harflar m = 26 ta bo'lganda quyudagi jadval hosil qilinadi:

5-jadval

k = 3 bo'lganda va alifbodagi harflar m = 26 ta bo'lganda quyidagi jadval hosil qilinadi:

A	→	D
B	→	E
C	→	F
D	→	G
E	→	H

F	→	I
G	→	J
H	→	K
I	→	L
J	→	M
K	→	N
L	→	O
M	→	P
N	→	Q
O	→	R
P	→	S
Q	→	T
R	→	U
S	→	V
T	→	W
U	→	X
V	→	Y
W	→	Z
X	→	A
Y	→	B
Z	→	C

3-mashq. Matn sifatida KOMPYUTER so'zini oladigan bo'lsak, Sezar usuli natijasida quyidagi shifrlangan yozuv hosil bo'ladi: NRPSBXWHU.

Sezar usulining kamchiligi bu bir xil harflarning o'z navbatida bir xil harflarga almashishidir.

Hozirgi vaqtda kompyuter tarmoqlarida tijorat axborotlari bilan almashishda uchta asosiy algoritmlar, ya'ni DES, CLIPPER va PGP algoritmlari qo'llanilmoqda. DES va CLIPPER algoritmlari integral sxemalarda amalga oshiriladi. DES algoritmining kriptobardoshligini quyidagi misol orqali ham baholash mumkin: 10 mln. AQSH dollari xarajat qilinganda DES shifrlash ochish uchun 21 minut, 100 mln AQSH dollari xarajat qilinganda esa 2 minut sarflanadi. CLIPPER tizimi SKIPJACK shifrlash algoritmini o'z ichiga oladi va bu algoritm DES algoritmidan 16 mln marta kuchliroqdir.

PGP algoritmi esa 1991 yilda Filipp Simmerman (AQSH) tomonidan yozilgan va elektron pochta orqali kuzatiladigan xabarlarni shifrlash uchun ishlatiladigan PGP dasturlar paketi yordamida amalga oshiriladi. FGP dasturiy vositalari Internet tarmog'ida elektron pochta orqali axborot jo'natuvchi foydalanuvchilar tomonidan shifrlash maqsadida keng foydalanilmoqda.

PGP (Pretty Good Privacy) kriptografiya dasturining algoritmi kalitli, ochiq va yopiq bo'ladi.

O'rin almashtirish usullari. O'rin almashtirish usullariga binoan dastlabki matn belgilangan uzunlikdagi bloklarga ajratilib har bir blok ichidagi simvollar o'rni ma'lum algoritm bo'yicha almashtiriladi.

Eng oson o‘rin almashtirishga misol tariqasida dastlabki axborot blokini matritsaga qator bo‘yicha yozishni, o‘qishni esa ustun bo‘yicha amalga oshirishni ko‘rsatish mumkin. Matritsa qatorlarini to‘ldirish va shifrlangan axborotni ustun bo‘yicha o‘qish ketma-ketligi kalit yordamida berilishi mumkin. Usulning kriptobardoshligi blok uzunligiga (matritsa ulchamiga) bog‘liq. Masalan uzunligi 64 simvolga teng bo‘lgan blok (matritsa o‘lchami 8x8) uchun kalitning $1,6 \cdot 10^9$ kombinatsiyasi bo‘lishi mumkin. Uzunligi 256 simvolga teng bo‘lgan blok (matritsa o‘lchami 16x16) kalitning mumkin bo‘lgan kombinatsiyasi $1,4 \cdot 10^{26}$ ga yetishi mumkin. Bu holda kalitni saralash masalasi zamonaviy EHMLar uchun ham murakkab hisoblanadi.

Gamilton marshrutlariga asoslangan usulda ham o‘rin almashtirishlardan foydalaniladi. Ushbu usul quyidagi qadamlarni bajarish orqali amalga oshiriladi.

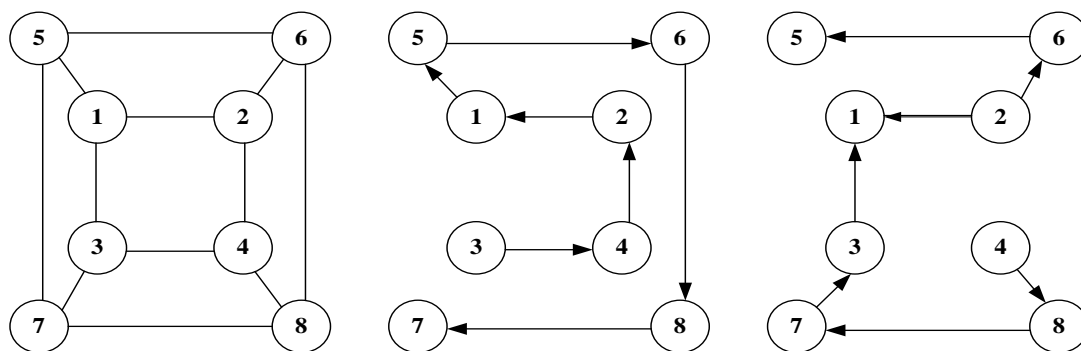
1-qadam. Dastlabki axborot bloklarga ajratiladi. Agar shifrlanuvchi axborot uzunligi blok uzunligiga karrali bo‘lmasa, oxirgi blokda bo‘sh o‘rinlarga maxsus xizmatchi simvollar-to‘ldiruvchilar joylashtiriladi (masalan, *).

2-qadam. Blok simvollarini yordamida jadval to‘ldiriladi va bu jadvalda simvolning tartib raqami uchun ma’lum joy ajratiladi (5.5-rasm).

3-qadam. Jadvaldagi simvollarni o‘qish marshrutlarning biri bo‘yicha amalga oshiriladi. Marshrutlar sonining oshishi shifr kriptobardoshligini oshiradi. Marshrutlar ketma-ket tanlanadi yoki ularning navbatlanishi kalit yordamida beriladi.

4-qadam. Simvollarning shifrlangan ketma-ketligi belgilangan L uzunlikdagi bloklarga ajratiladi. L kattalik 1-qadamda dastlabki axborot bo‘linadigan bloklar uzunligidan farqlanishi mumkin.

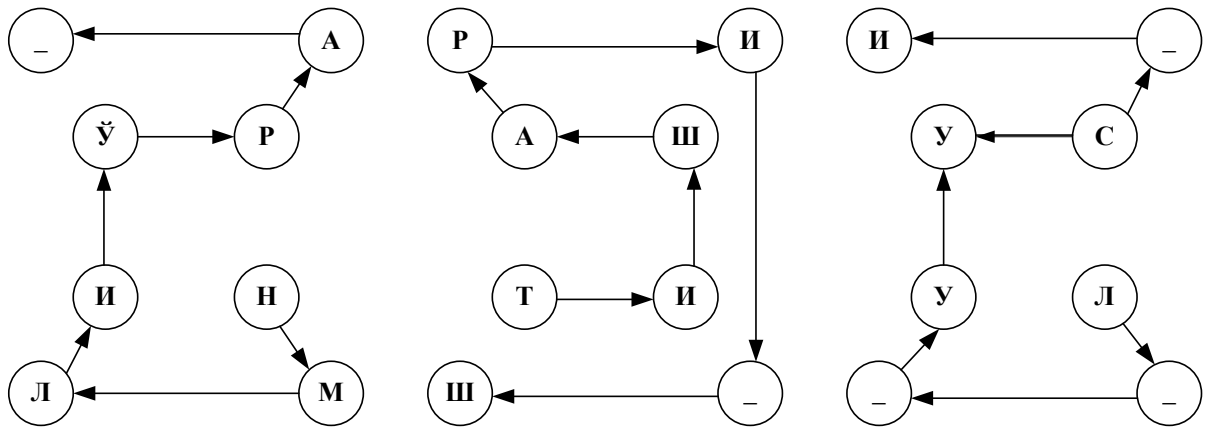
Rasshifrovka qilish teskari tartibda amalga oshiriladi. Kalitga mos holda marshrut tanlanadi va bu marshrutga binoan jadval to‘ldiriladi.



30-rasm. 8-elementli jadval va Gamilton marshrutlari variantlari

Jadvaldan simvollar element nomerlari kelishi tartibida o‘qiladi.

Misol. Dastlabki matn T_0 «O‘RIN ALMASHTIRISH USULI»ni shifrlash talab etilsin. Kalit va shifrlangan bloklar uzunligi mos holda quyidagilarga teng: $K=\langle 2,1,1 \rangle$, $L=4$. Shifrlash uchun 5.6-rasmدا keltirilgan jadval va ikkita marshrutdan foydalaniladi. Berilgan shartlar uchun matritsalarini to‘ldirilgan marshrutlar 5.10-rasmدا keltirilgan ko‘rinishga ega.



31-rasm. Gamilton marshruti yordamida shifrlash misoli

1-qadam. Dastlabki matn uchta blokka ajratiladi.

$B1 = \langle O'RIN_ALM \rangle$, $B2 = \langle ASHTIRISH \rangle$, $B3 = \langle USULI^{**} \rangle$;

2-qadam. 2,1,1 marshrutli uchta matritsa to'ldiriladi;

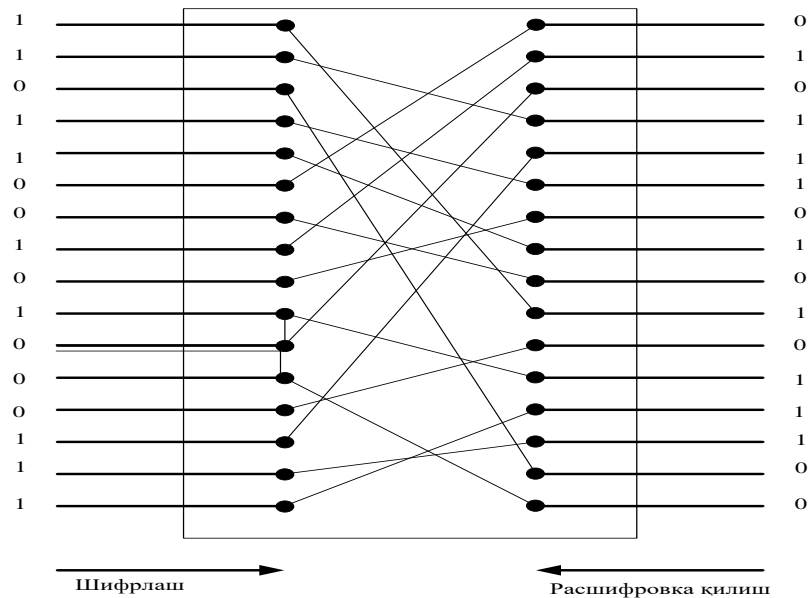
3-qadam. Marshrutlarga binoan simvollarini joy-joyiga qo'yish orqali shifratmatni hosil qilish.

$T_1 = \langle NMLIO'RA_TISHARI_ShL_UUS_I \rangle$

4-qadam. Shifr matni bloklarga ajratish.

$T_1 = \langle NMLI O'RA_TISHA RI_Sh L_U US_I \rangle$

Amaliyotda o'rin almashtirish usulini amalga oshiruvchi maxsus apparat vositalar katta ahamiyatga ega (5.7-rasm).



32-rasm. O'rin almashtirish sxemasi

Dastlabki axborot blokining parallel ikkili kodi (masalan, ikki bayt) sxemaga beriladi. Ichki kommutatsiya hisobiga sxemada bitlarning bloklardagi o'rinlari

almashtiriladi. Rasshifrovka qilish uchun esa sxemaning kirish va chiqish yo‘llari o‘zaro almashtiriladi.

O‘rin almashtirish usullarining amalga oshirilishi sodda bo‘lsada, ular ikkita jiddiy kamchiliklarga ega. Birinchidan, bu usullarni statistik ishlash orqali fosh qilish mumkin. Ikkinchidan, agar dastlabki matn uzunligi K simvoldan tashkil topgan bloklarga ajratilsa, shifrnı fosh etish uchun shifrlash tizimiga bittasidan boshqa barcha simvolları bir xil bo‘lgan test axborotining $K-1$ blokini yuborish kifoya.

Nazorat savollari:

1. Kriptografiya deb nimaga aytiladi?
2. Kriptografiyada qanday ko‘p qo‘llaniladigan usullarnı bilasiz?
3. Sesar usuli qanday usul?
4. Ochiq va yopiq kalitlar haqida aytib bering.

6-AMALIY MASHG‘ULOT OBYEKTGA YO‘NALTIRILGAN DASTURLASH TEXNOLOGIYALARI

Ishdan maqsad: Vizual dasturlash bilan tanishish va obyekt xossalarini tahrirlash va ularni boshqarish usullarini o‘rganish.

Reja

1. Nazariy qism bilan tanishish.
2. Dastur tuzish asoslarini o‘rganish.
3. Ilovalar yaratish.

Nazariy qism

Borland C++ Builder 6 dasturida dasturlash ikkita o‘zaro ta’sir etuvchi bir-biri bilan bog‘liq jarayon asosida tashkil qilinadi:

- dasturnı vizual loyihalash jarayoni;
- dastur kodlarini konsol muhitida kiritish jarayoni.

Vizual loyihalash jarayonida dasturda yaratilayotgan ilovaning dizayni shakllanadi. Dasturchi vizual loyihalash jarayonini bajarganda Borland C++ Builder 6 avtomatik ravishda dastur kodini yaratishni boshlaydi. Dasturchi loyihasini ishlashi mobaynida dastur kodini C++ tilining maxsus operatorlari bilan to‘ldiradi.

Borland C++ Builder 6 dastur ham Windows amaliyot tizimining boshqa dasturlari kabi ishga tushiriladi:

Пуск => Все программы => Borland C++ Builder 6 => C++ Builder 6.

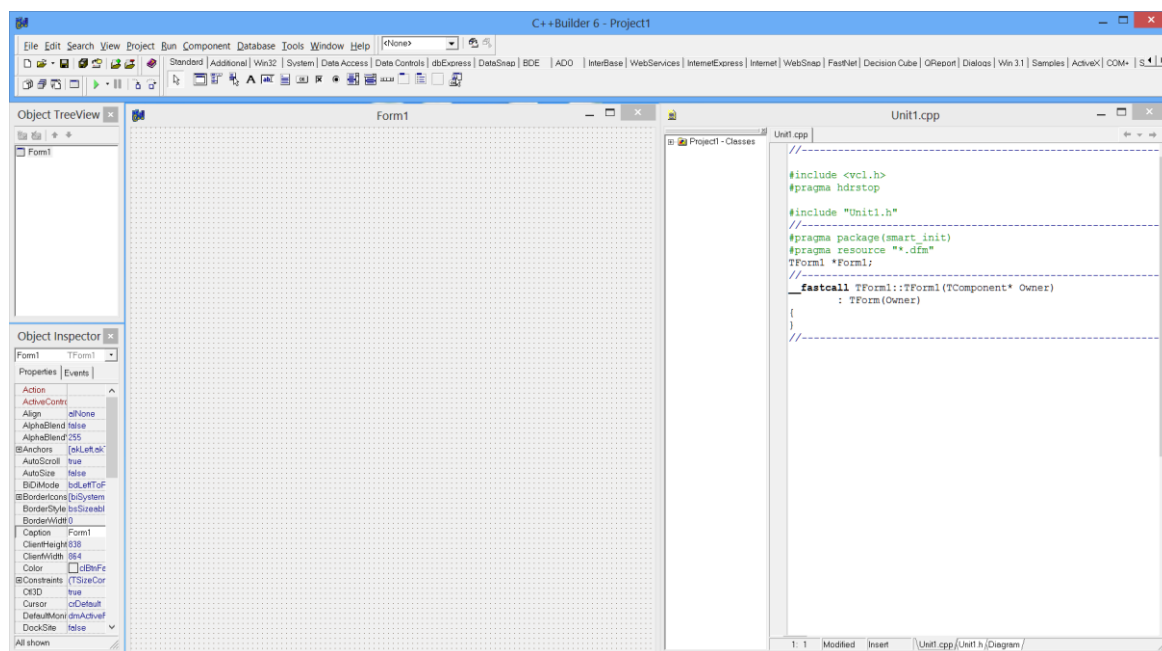
Dastur yuklangandan keyin ekranda darcha paydo bo‘ladi. Borland C++ Builder 6 dasturi quyidagi beshta asosiy darchani o‘z ichiga oladi:

1. Bosh oyna – C++ Builder 6 (Project1);
2. Forma oynasi (Form1);
3. Obyekt xossalarini tahrirlash oynasi (Object Inspector);
4. Obyektlar ro‘yxatini ko‘rish oynasi (Object tree View);
5. Dastur kodlarini kiritish oynasi (Unit.cpp).

Bosh oyna (Project1) ekranning yuqori qismida joylashgan bo‘lib, uning birinchi qatorida sarlavha, ya‘ni loyihaning nomi (C++ Builder 6 – Project1) joylashgan.

Ikkinchi qatorda asosiy menyular qatori gorizontol ko‘rinishda joylashgan. Asosiy menyular qatori dasturni yaratish uchun kerak bo‘ladigan barcha buyrug‘lar va funksiyalarga murojaat qilish imkonini yaratadi.

Keyingi qatorning chap tarafida tezkor murojaat etish tugmalari mavjud. Bajaradigan vazifasiga qarab ketma-ketlikda birlashtirilgan. Ular tez-tez ishlatiladigan buyruqlarga tezkor murojaat etish imkonini beradi. O‘ng tarafida vizual komponentalar palitrasi VCL (Visual Component Library, vizual komponentalar kutubxonasi) keltirilgan. Windows operatsion tizimi ilovalarni yaratish uchun vizual komponentalarni o‘z ichiga oladi. Vizual komponentalar palitrasi bir nechta qismlardan iborat guruhlariga bo‘lingan. Bu vizual komponentalar palitrasi yordamida tezkor va oson usulda dasturlarni yaratish mumkin.

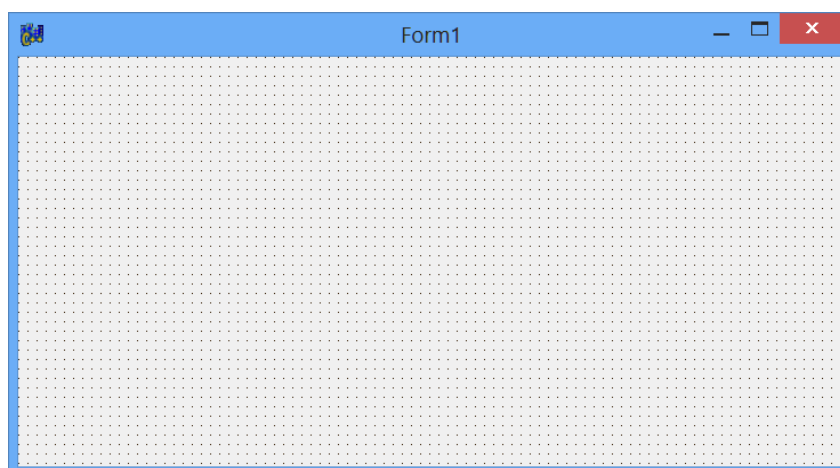


33-rasm. Borland C++ Builder 6 integrallashgan sohasi



34-rasm. Bosh oyna

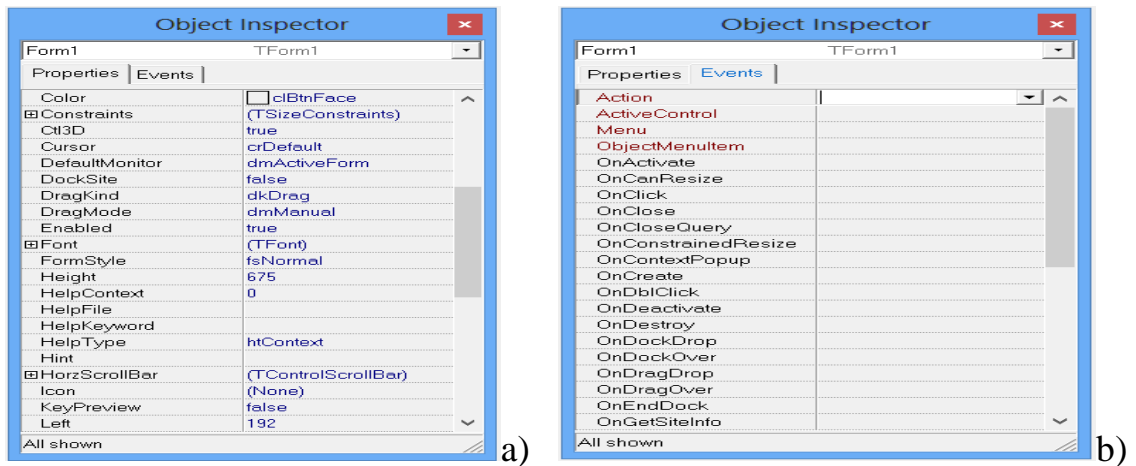
Forma oynasi (Form1) yangi yaratilishi kerak bo‘ladigan dasturning ko‘rinishi hisoblanadi. Forma oynasi C++ Builder 6 ilovalari uchun asos bo‘lib, unda yaratilayotgan dasturga komponentalarni joylashtirish mumkin. U dasturning sarlavhasidan boshlanadi.



35-rasm. Forma oynasi

Object Inspector oynasi (Object Inspector) obyekt xossalari va hodisalarini tahrirlash uchun xizmat qiladi. Obyektga yo‘naltirilgan dasturlashda dastur bu obyektlar tizimi bo‘lib, har bir obyekt bir qator xossalarga ega bo‘lishi mumkin. Xossa esa ma’lumotlar va ularni boshqarish usullaridan iborat. Obyekt xossalari bu – obyektga berilgan xarakteristika bo‘lib, uning ko‘rinishi, joylashishi va holatidir. Bundan tashqari obyekt turli hodisalarni ham o‘rnatishi mumkin. Hodisa deb bajarish, boshqarish usuliga aytiladi, masalan, sichqonni bosish, kursorni siljitish va hokazo amallarga aytiladi.

Object Inspector oynasi xossa va hodisalar parametrlarini o‘rnatish uchun mo‘ljallangan bo‘lib, u ikkita sahifadan iborat: Properties (xossalar: a-rasm) va Events (hodisalar: b-rasm).

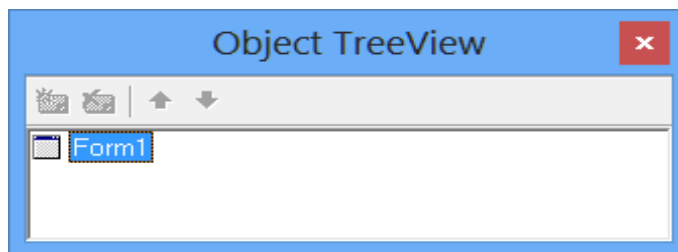


36-rasm. Object Inspector oynasi.
 a) “Properties” darchasi; b) “Events” darchasi

Properties sahifasi ajratilgan obyekt yoki komponentaning xossalarini o‘rnatadi. Masalan, Caption (yozuv) yordamida obyekt yozuvini o‘rnatish mumkin, Color (rang) xossasi obyektning rangini o‘rnatadi.

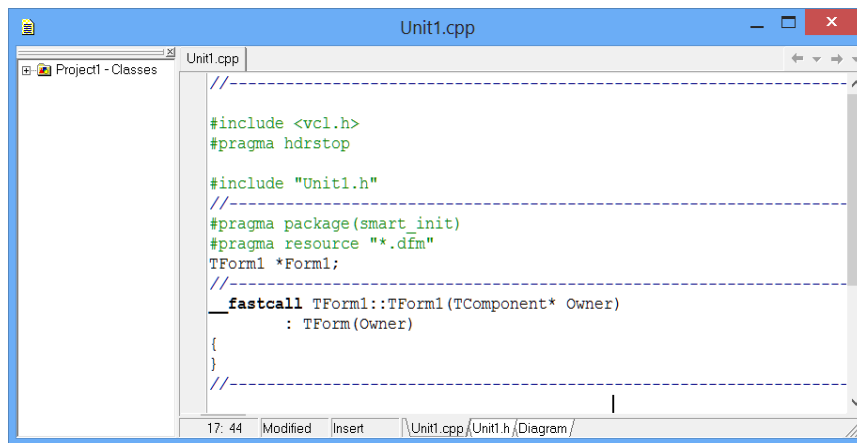
Events (hodisalar) sahifasida Forma ilovasida tanlangan obyekt uchun dastur bajarilishi jarayonida hodisa, ya’ni uni ishga tushirish holati belgilanadi. Har bir holatning standart nomi belgilangan. Masalan, OnClick – sichqonchaning chap tugmasini bir marta bosish, OnDbClick – sichqonchaning chap tugmasini bir marta bosish.

Obyektlar ro‘yxatini ko‘rish oynasi (Object Tree View) dasturda ishlatilayotgan komponentalarni daraxt ko‘rinishida tasvirlab beradi. Komponentalarni joylashuvini va ularning holatini ko‘rsatib turadi.



37-rasm. Obyektlar ro‘yxatini ko‘rish oynasi

Dastur kodlarini kiritish oynasi (Unit.cpp) yaratilayotgan yangi dasturning kodini (matni) kiritish va uni tahrirlash uchun foydalaniladi. Dasturda ma’lum jarayonini amalga oshirishi uchun kerak bo‘ladigan operatorlar ketma-ketligi kiritiladi.



38-rasm. Dastur kodlarini kiritish oynasi

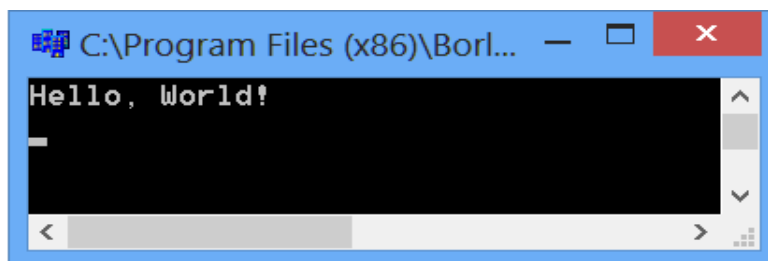
Konsol muhitida “Hello world” soʻzini chiqaruvchi dasturni koʻrib chiqamiz. Buning uchun yangi konsol muhitini ochamiz. Ochilgan muhitda quyidagi dastur matni kiritiladi:

```

#include <vcl.h>
#include <iostream.h>
#include <conio.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{
    cout << "Hello, World!" << endl;
    getch();
    return 0;
}
//-----

```

Natijada quyidagi dastur koʻrinishi paydo boʻladi:



39-rasm. Natija oynasi

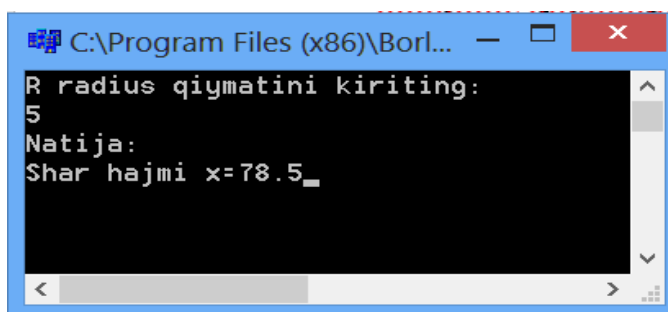
Misol. R radiusli sharning hajmi quyidagi formula boʻyicha hisoblansin:

$$V = \frac{4}{3} \pi R^2$$

Dastur quyidagi ko‘rinishda bo‘ladi.

```
#include <stdio.h>
#include <iostream.h>
#include <conio.h>
#include <math.h>
#include <vcl.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{ float pi=3.14;
  float r,v;
  cout << "R radius qiymatini kiriting:"<<"\n";
  cin >>r;
  v=4/3*(pi*pow(r,2));
  cout << "Natija: "<<"\n";
  cout <<"Shar hajmi x="<<v;
  getch();
  return 0;
}
//-----
```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:



40-rasm. Natija oynasi

1-misol. y funksiyaning qiymatini aniqlash dasturi tuzilsin. Konsol muhitida amalga oshirilsin.

$$y = \begin{cases} \frac{4r + 3m^2}{r - m}, & \text{agar } r \geq m + 1 \\ |r - m|, & \text{agar } r < m + 1 \end{cases}$$

Dastur matni quyidagi ko‘rinishda bo‘ladi:

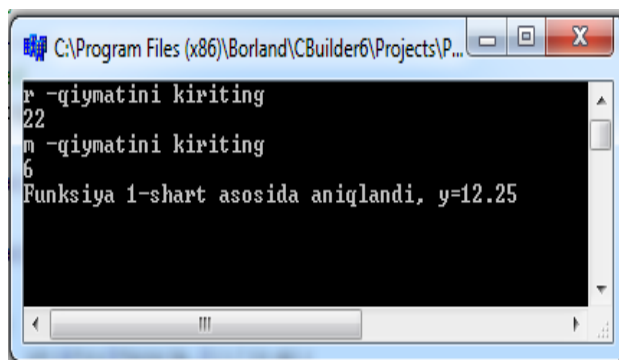
```
//-----
#include <stdio.h>
```

```

#include<iostream.h>
#include<conio.h>
#include <math.h>
#include <vcl.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{ float y, r, m;
cout<<"r -qiymatini kiriting"<<"\n"; cin>>r;
cout<<"m -qiymatini kiriting"<<"\n"; cin>>m;
    if (r>=m+1)
        {y=(4*r+3*pow(m,2))/(r-m);
        cout<<"Funksiya 1-shart asosida aniqlandi, y="<<y;} else
{y=fabs(r-m);
    cout<<"Funksiya 2-shart asosida aniqlandi, y="<<y;
} getch();
return 0;
}
//-----

```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:



41-rasm. Natija oynasi

Nazorat savollari:

1. Borland C++ Builder tizimi qanday dasturlash texnologiyalarini o‘zida birlashtirgan?
2. Borland C++ Builder tizimi ishchi oynasi qanday tarkibiy qismlardan iborat?

7-AMALIY MASHG‘ULOT MANTIQUIY DASTURLASH TEXNOLOGIYASI

Ishdan maqsad: Mantiqiy dasturlash bo‘yicha ko‘nikmalarni olish. Murakkab tarmoqlanuvchi hisoblash jarayonlarini amalga oshirishni o‘rganish.

Reja

1. Nazariy qism bilan tanishish.
2. Tarmoqlanuvchi jarayonlarni dasturlash.
3. Mantiqiy dasturlashni o‘rganish.
4. Hisobot tayyorlash.

Nazariy qism

Ko‘pgina masalalarni yechishda ba‘zi bir jarayonlar ma‘lum shart yoki shartlarning qo‘yilishiga nisbatan bajariladi, ya‘ni shartning bajarilishi yoki bajarilmasligiga ko‘ra boshqa jarayonlar, amallar tanlanadi. Bunday jarayonlar “Tarmoqlanuvchi jarayonlar” deb yuritiladi.

Tarmoqlanuvchi hisoblash jarayonlari oddiy va murakkab bo‘lishi mumkin. Bu esa jarayondagi tarmoqlar soniga bog‘liq. Ma‘lum bir tarmoqlanuvchi jarayon tarkibida yana tarmoqlanishlar bo‘lishi mumkin. Bunday tarmoqlanishlari bor bo‘lgan hisoblash jarayonlari “Murakkab tarmoqlanuvchi hisoblash jarayonlari” deb ataladi.

C++ tilida tarmoqlanuvchi jarayonlarni dasturlash uchun shartsiz, shartli o‘tish va tanlash operatorlaridan foydalaniladi.

Dasturda boshqaruvni ma‘lum shart asosida u yoki bu tarmoqqa (ma‘lum jarayonlar ketma-ketligi) uzatish shartli o‘tish operatori yordamida amalga oshiriladi. Shartli o‘tish operatori ikki xil: to‘liq va qisqa ko‘rinishda ishlatilishi mumkin.

Shartli o‘tish operatorining to‘liq ko‘rinishini ko‘rib chiqamiz. Uning dasturda ko‘rinishi quyidagicha yoziladi:

if (<mantiqiy ifoda>) <operator –1>; else <operator–2>;

bu yerda: if – agar, else– aks holda ma‘nosini anglatuvchi xizmatchi so‘zlar, operator –1 va operator –2 ixtiyoriy operatorlar.

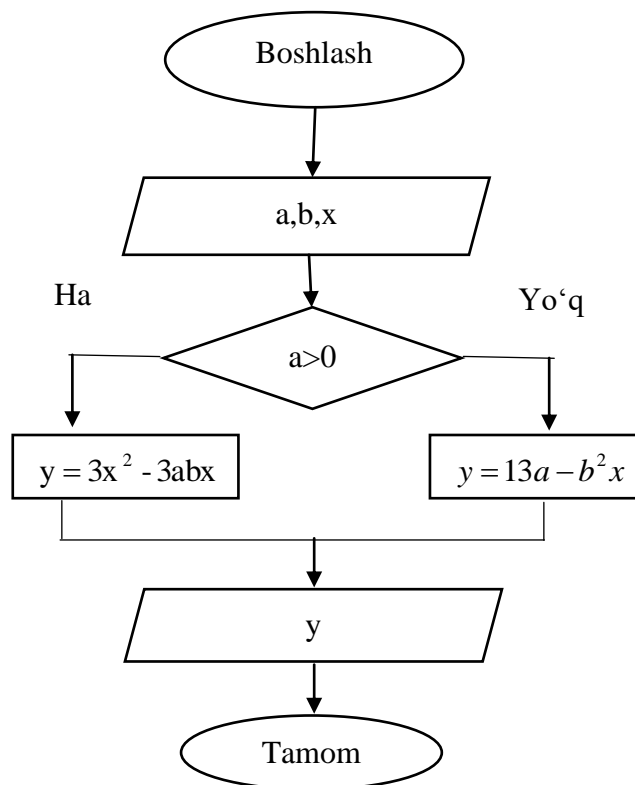
Operatoridagi mantiqiy ifoda boshqaruvni uzatish shartini belgilaydi.

Operatorning ishlash tartibi quyidagicha: agar keltirilgan mantiqiy ifoda true (rost) qiymatni qabul qilsa, ya‘ni qo‘yilgan shart bajarilsa; operator – 1 bajariladi, aks holda else xizmatchi so‘zdan keyingi operator – 2 bajariladi.

Misol. y funksiyaning qiymatini aniqlash dasturi tuzilsin. Vizual muhitida amalga oshirilsin.

$$y = \begin{cases} 3x^2 - 3abx & \text{agar } a > 0 \\ 13a - b^2x & \text{agar } a \leq 0 \end{cases}$$

Bu misolni yechish algoritmi quyidagi ko‘rinishda bo‘ladi:



42-rasm. Dastur blok sxemasi

Misolni vizual muhitda dasturlash uchun 4 ta Label, 2 ta Edit, 2 BitBtn va RadioGroup komponentalari kerak bo‘ladi.

Forma darchasiga o‘rnatilgan komponentalar xususiyatlarini quyidagicha belgilaymiz:

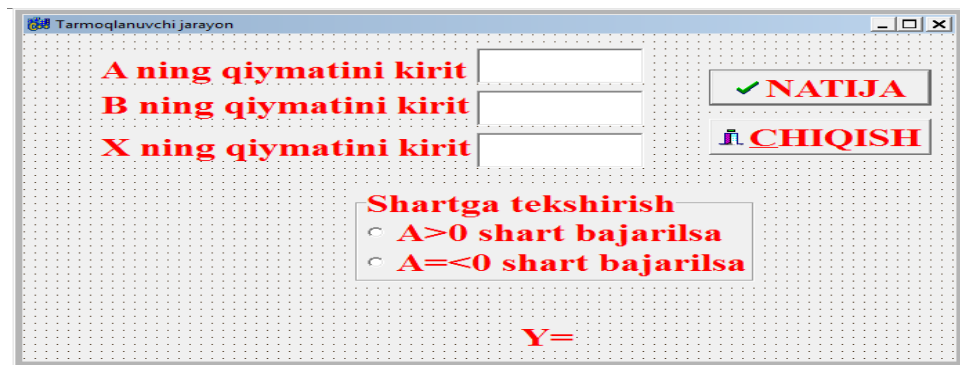
6-jadval.

Komponentalar xususiyatlarini kiritish

Komponenta nomi	Xususiyat nomi (Object Inspector darchasining xolati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	“Tarmoqlanuvchi jarayon” so‘zi kiritiladi.
Label1	Caption (Properties)	“A ning qiymatini kirit” so‘zi kiritiladi.
Label2	Caption (Properties)	“B ning qiymatini kirit” so‘zi kiritiladi.

Label3	Caption (Properties)	“X ning qiymatini kirit” so‘zi kiritiladi.
Label4	Caption (Properties)	“Y=” so‘zi kiritiladi.
Edit1	Text (Properties)	“Edit1” so‘zini o‘chirib tashlang.
Edit2	Text (Properties)	“Edit2” so‘zini o‘chirib tashlang.
BitBtn1	Kind (Properties)	“bkOK” xususiyati tanlanadi.
	Caption (Properties)	“Natija” so‘zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.
BitBtn2	Kind (Properties)	“bkClose” xususiyati tanlanadi.
	Caption (Properties)	“Chiqish” so‘zi kiritiladi.
	OnClick (Events)	Close(); kiritiladi.
RadioGroup1	Caption (Properties)	“Shartga tekshirish” so‘zi kiritiladi.
	Items (Properties)	“A>0 shart bajarilsa va A=<=0 shart bajarilsa” so‘zi kiritiladi

Mavjud komponentalar xususiyatlari kiritilgandan so‘ng dastur dizayni quyidagi ko‘rinishga keladi:



43-rasm. Dastur ko‘rinishi

Dastur dizayni tayyor bo‘laganidan so‘ng quyidagi dastur matni kiritiladi:

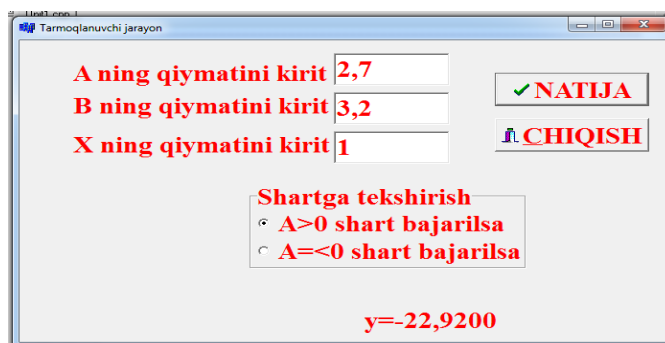
```
//-----
#include <vcl.h>
#include <math.h>
#pragma hdrstop
#include "Unit1.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
```

```

}
//-----
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
{
float y,x,a,b;
a=StrToFloat(Edit1->Text);
b=StrToFloat(Edit2->Text);
x=StrToFloat(Edit3->Text);
if (a>0) {
    y= 3*pow(x,2)-3*a*b*x;
    RadioGroup1->ItemIndex=0;
    Label4->Caption ="y="+FloatToStrF(y,ffFixed,6,4);
    }
    else {
    y=13*a-pow(b,2)*x;
    RadioGroup1->ItemIndex=1;
    Label4->Caption ="y="+FloatToStrF(y,ffFixed,6,4);
    } }
//-----
void __fastcall TForm1::BitBtn2Click(TObject *Sender)
{
Close();
}
//-----

```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:



44-rasm. Natija oynasi

Nazorat savollari:

1. Tarmoqlanuvchi jarayon deb qanday hisoblash jarayonlariga aytiladi?
2. Tarmoqlanuvchi jarayon tarkibi qanday qismlardan iborat?
3. Tarmoqlanuvchi operatorlarini sanab bering.

8 - AMALIY MASHG‘ULOT

DASTURLASH TIZIMLARIDA ILOVALAR YARATISH TEKNOLOGIYASI

Ishdan maqsad: Dasturlash yordamida zamonaviy ilovalar yaratish texnologiyasi o‘rganish.

Reja

1. Nazariy qism bilan tanishib chiqing.
2. Fayllar bilan ishlashni mo‘ljallangan protsedura va funksiyalarni o‘rganib chiqing.
3. Ma‘lumotlarning faylli toifasi yordamida dastur tuzing.

Nazariy qism

C++ ixtiyoriy dasturida o‘zida tur, o‘zgarmas, makros, funksiya va sinflar aniqlangan standart kutubxonaga murojaat mavjud bo‘ladi.

Ularni dasturda qo‘llash uchun *#include* direktivasi yordamida dasturning boshlang‘ich matniga mos e‘lonlardan tashkil topgan sarlavha fayllarni (заголовочный файл) kiritish lozim.

Kutubxona funksiyalarini ularni qo‘llanilishiga qarab quyidagi guruhlariga bo‘lish mumkin: kiritish/chiqarish, qatorlarni qayta ishlash, matematik funksiyalar, dinamik xotira bilan ishlash, qidirish, saralash va boshqalar.

C++da qiritish/chiqarish oqimlar orqali amalga oshiriladi. C++da oqimlar orqali kiritish/chiqarish funksiyalarini qo‘llash uchun dasturda quyidagi sinflarni ishlatish kerak:

- <**istream.h**>- kiritish oqimi;
- <**ostream.h**> - chiqarish oqimi;
- <**iostream.h**> - kiritish/chiqarish.

Kiritish/chiqarishda ma‘lumotlar baytlar oqimi sifatida qaraladi. Fizik jihatdan oqim faylni tashkil qiladi.

Quyidagi faylli oqimlar fayllar bilan ishlash uchun xizmat qiladi:

1) *ofstream name(" path\file_name");* - faylli chiqarish. Oqimni aniq fayl bilan bog‘laydi, faylni yozish uchun ochadi. Bu yerda name - oqim nomi (ixtiyoriy identifikator), faylga ma‘lumotlarni yozish va yoki undan o‘qishda shu nomdan foydalaniladi. path\file_name - fizik faylning yo‘li.

Masalan: *ofstream book("c:\kitob\bob\bo‘lim1.dat");*

Agar faylning yo‘li ko‘rsatilmasa, u holda fayl joriy katalogda yaratiladi:

ofstream books("bo‘lim1.dat");


2) *ifstream name(" path\file_name");* - faylni undan ma‘lumotlar o‘qish uchun ochadi.

Masalan: *ifstream book("c kitob\bob\bo‘lim1.dat");*


ifstream books("\bo‘lim1.dat");

O‘qish va yozish uchun ochilgan fayllarni albatta yopish kerak -
name.close();
Book.close(): books.close();

Fayllar bilan ishlash uchun Forma ilovasini yaratishda turli bo‘limlardan iborat menyular tashkil qilish, fayllarni saqlash, ochish tugmalarini o‘rnatgan holda bu amallarni bajarish imkoniyatlari mavjud:

 **TMainMenu** – bosh menyu yaratish uchun ishlatiladi. Menyuni yaratish uch qadamdan iborat: 1) *TMainMenu* formaga o‘rnatish; 2) obyektlar inspektorining *Items* xususiyati yordamida menyu dizaynerini chaqirish; 3) menyu dizaynerida menyu qismlarini kiritish.

 **OpenDialog** – kerakli faylni ochish uchun ishlatiladigan komponenta.



 **SaveDialogs** – faylni saqlash uchun ishlatiladigan komponenta.

Bu komponentalar Windowsning muloqot oynalari hisoblanadi va dialog komponentalar menyusida joylashgan bo‘ladi.


Quyidagi masalada ularning ba‘zi birlari bilan tanishtirib chiqamiz.

Masala. Fayllar bilan ishlash uchun yangi forma yarating.

Qo‘yilgan masalani yechish uchun quyidagi tartibda amallarni bajaring:

a) yaratilgan formaga Standart bo‘limidan  **MainMenu** komponentasini tanlang va kerakli joyga o‘rnating. *MainMenu* komponentasi formaga yangi menyu yaratish uchun ishlatiladi.  piktogrammasining ustida 2 marta bosing va kerakli menyu nomlarini object inspector bo‘limidagi **caption**da yangi menyu nomini kiring.

b) standart menyu bo‘limidan **Memo** komponentasini o‘rnating. Object inspector bo‘limida **Lines** xossasiga kirib, **Memo1** so‘zini o‘chirib tashlang. Object inspector bo‘limida **align** xossasini **alclientga** almashtiring.

d) **dialogs** menyu bo‘limidan  **OpenDialogs** komponentasini o‘rnating. Object inspector bo‘limida **Filter** xossasiga kirib, filter bo‘limiga ***.txt** so‘zini kiring.

e) **dialogs** menyu bo‘limidan  **SaveDialogs** komponentasini o‘rnating. Object inspector bo‘limida **Filter** xossasiga kirib, filter bo‘limiga ***.txt** so‘zini kiring.

Dastur kodi:

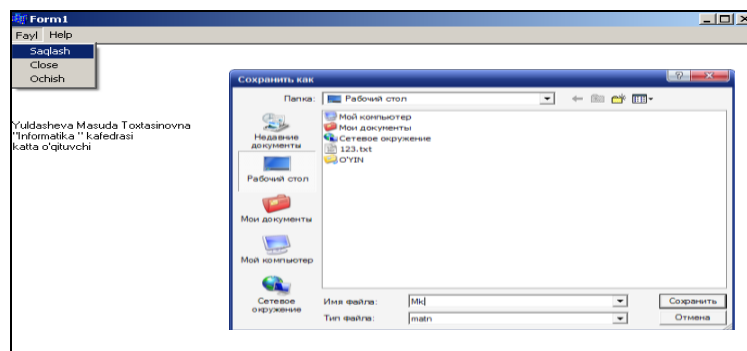
```
//-----  
#pragma package(smart_init)  
#pragma resource "*.dfm"  
TForm1 *Form1;  
//-----  
__fastcall TForm1::TForm1(TComponent* Owner)  
: TForm(Owner)  
{
```

```

}
//-----
void __fastcall TForm1::Saqlash1Click(TObject *Sender)
{
    if(Form1->SaveDialog1->Execute())
    Form1->Memo1->Lines->SaveToFile(Form1->SaveDialog1-
    >FileName+'.txt');
}
//-----
void __fastcall TForm1::Ochish2Click(TObject *Sender)
{
    if(Form1->OpenDialog1->Execute())
    Form1->Memo1->Lines->LoadFromFile(Form1->OpenDialog1-
    >FileName);
}
//-----
void __fastcall TForm1::Help1Click(TObject *Sender)
{
    //showmessage('do you have question');
}
//-----
void __fastcall TForm1::Close1Click(TObject *Sender)
{
    Form1->Close();
}
//-----

```

Natijada quyida ko'rsatilgan oyna chiqadi (10-rasm):



45-rasm. Menyu tashkil qilingan oyna ko'rinishi

Nazorat savollari:

1. Fayl nima?
2. C++ tilida fayllarning necha xil turi mavjud?
3. Fayl bilan ishlash uchun qaysi funktsiya va protseduralar mavjud?

Foydalanilgan adabiyotlar

1. Kadirov M.M. Axborot texnologiyalari. O‘quv qo‘llanma, 1-qism. – T.: Sano-standart, 2018. - 320 b.
2. Kadirov M.M. Texnik tizimlarda axborot texnologiyalari. Darslik, 2-qism. – T.: Fan va texnologiya, 2018. - 288 b.
3. Kunwoo Lee. Principles of CAD/CAM/CAE: The Computer Aided Engineering Design Series. 5st Edition. Addison Wesley Longman, USA, 2015.
4. Alex Allain. Jumping into C++. USA, 2014. p 340.
5. Nazirov Sh.A., Qobulov R.V., Bobojonov M.R., Raxmanov Q.S. C va C++ tili. Darslik. –T.: Voris, 2013. - 488 b.
6. Ganiyev S.K., Karimov M.M., Tashev K.A. Axborot xavfsizligi. Oliy o‘quv yurti talabalari uchun darslik. –T.: Fan va texnologiya, 2017. - 372 b.

Qo‘shimcha adabiyotlar

7. “O‘zbekiston Respublikasini yanada rivojlantirish bo‘yicha Harakatlar strategiyasi to‘g‘risida” O‘zbekiston Respublikasi Prezidentining Farmoni. 2017-yil 7-fevral, PF-4947-son.
8. Kenjabayev A.T., Ikramov A., Allanazarov A,Sh. Axborot-kommunikatsiya texnologiyalari. O‘quv qo‘llanma. – T.: O‘zbekiston faylasuflari milliy jamiyati nashriyoti, 2017. - 408 b.
9. Nazirov Sh.A., Ivanova G.S., Gaynazarov S.M. Dasturlash texnologiyalari. Darslik. – T.: O‘zbekiston faylasuflari milliy jamiyati nashriyoti, 2014. -280 b.

Internet saytlari

10. www.gov.uz – O‘zbekiston Respublikasi hukumat portali.
11. www.lex.uz– O‘zbekiston Respublikasi Qonun hujjatlari ma’lumotlari milliy bazasi.
12. www.ru.wikipedia.org
13. <http://www.cadcamcae.lv/>
14. labcenter-electronics.com
15. <http://ziyonet.uz/uzc>

Mundarija

1-amaliy mashg'ulot	MATHCAD tizimi. tizimning ishlash texnologiyasini o'rganish. MATHCAD integrallashgan sohasida muhandislik masalalarini yechish.	3
2-amaliy mashg'ulot	MATHCAD tizimida funksiya grafiklarini amalga oshirish.	8
3-amaliy mashg'ulot	MATLAB muhiti, uning integrallashgan sohasini o'rganish. Simulink paketi.	13
4-amaliy mashg'ulot	MATLAB vektor, matritsa, massivlar bilan ishlash, matlab tizimida dasturlash. texnik ob'ektlarni immitatsion modelini yaratish.	22
5-amaliy mashg'ulot	Himoyalashning kriptografik usullaridan foydalanish. sezar usuli	27
6-amaliy mashg'ulot	Obyektga yo'naltirilgan dasturlash texnologiyalari	34
7-amaliy mashg'ulot	Mantiqiy dasturlash texnologiyasi	40
8-amaliy mashg'ulot	Dasturlash tizimlarida ilovalar yaratish texnologiyasi	45
Foydalanilgan adabiyotlar.		48

Tuzuvchilar:

Kadirov M.M., To‘laganov Z.Ya., Zokirova F.R.

Texnik tizimlarda axborot texnologiyalari

Amaliy mashg‘ulotlarni bajarish uchun
o‘quv-uslubiy ko‘rsatmalar

Muharrir: Miryusupova Z.M.