

**O'ZBEKISTON RESPUBLIKASI ALOQA, AXBOROTLASHTIRISH VA
TELEKOMMUNIKATSIYA TEXNOLOGIYALARI DAVLAT QO'MITASI**

**TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
SAMARQAND FILIALI**

**Kompyuter Injiniringi fakulteti
«Axborot Texnologiyalari» kafedrası**

S. A. Nuriyev

**«C++ dasturlash»
fanidan**

laboratoriya mashg'ulotlari uchun uslubiy ko'rsatma

Samarqand – 2014

Mundarija

Laboratoriya ishi №1	4
Mavzu: Ikki oʻlchamli massivlar. Massivlarni qayta ishlash.	4
Maqsad:.....	4
I. Masalaning qoʻyilishi.	4
II. Variantlar	4
III. Hisobot mazmuni	5
IV. Uslubiy koʻrsatmalar	5
Masalaning qoʻyilishi.	5
Ishdan maqsad.	5
Nazariy qism	5
Berilgan masala algoritmi.	6
Asosiy dastur matni.	7
Dastur bajarilish natijalari tahlili.	7
Natijalar listingi.	8
Foydalanilgan adabiyotlar.....	8
Laboratoriya ishi №2.....	9
Mavzu: Funktsiyalar yaratish. Massivlar, funktsiyalar va rekursiv funktsiyalar.	9
Maqsad:.....	9
I. Masalaning qoʻyilishi.	9
II. Variantlar	9
III. Hisobot mazmuni	10
IV. Uslubiy koʻrsatmalar	10
Masalaning qoʻyilishi.	10
Ishdan maqsad.	10
Nazariy qism	10
Berilgan masala algoritmi.	12
Asosiy dastur matni.	12
Dastur bajarilish natijalari tahlili.	14
Natijalar listingi.	15
Foydalanilgan adabiyotlar.....	16
Laboratoriya ishi №3.....	17
Mavzu: Strukturalar va satrlar.....	17
Maqsad.....	17
I. Masalaning qoʻyilishi.	17
II. Variantlar	17
III. Hisobot mazmuni	18
IV. Uslubiy koʻrsatmalar	19
Masalaning qoʻyilishi.	19
Ishdan maqsad.	19

Nazariy qism.....	19
Berilgan masala algoritmi.....	29
Asosiy dastur matni.....	30
Dastur bajarilish natijalari.....	30
Natijalar listingi.....	31
Foydalanilgan adabiyotlar.....	31
Laboratoriya ishi №4.....	32
Mavzu: C++ ning grafik imkoniyatlari. Grafik funksiyalar, Grafik animatsiyalar, rangli formalar hosil qilish.	32
Maqsad.....	32
I. Masalaning qo'yilishi.....	32
II. Variantlar.....	32
III. Hisobot mazmuni.....	32
IV. Uslubiy ko'rsatmalar.....	33
Masalaning qo'yilishi.....	33
Ishdan maqsad.....	33
Nazariy qism.....	33
Berilgan masala algoritmi.....	36
Asosiy dastur matni.....	37
Dastur bajarilish natijalari tahlili.....	37
Natijalar listingi.....	38
Foydalanilgan adabiyotlar.....	38
Laboratoriya ishi №5.....	39
Mavzu: Fayllar bilan ishlash.....	39
Maqsad.....	39
I. Masalaning qo'yilishi.....	39
II. Variantlar.....	39
III. Hisobot mazmuni.....	40
IV. Uslubiy ko'rsatmalar.....	41
Masalaning qo'yilishi.....	41
Ishdan maqsad.....	41
Nazariy qism.....	41
Berilgan masala algoritmi.....	43
Asosiy dastur matni.....	44
Dastur bajarilish natijalari tahlili.....	44
Natijalar listingi.....	45
Foydalanilgan adabiyotlar.....	45

Laboratoriya ishi №1

Mavzu: Ikki o'lchamli massivlar. Massivlarni qayta ishlash.

Maqsad: Massivlar ustida amallar bajarish ko'nikmalarini hosil qilish. Matritsalar ustida bajariladigan amallarni ikki o'lchovli massivlardan foydalanib dasturiy ta'minotini yaratish ko'nikmalarini oshirish.

I. Masalaning qo'yilishi. Ikki o'lchovli massivni e'lon qiling. Variantda ko'rsatilgan matritsalar ustida bajariladigan amallarning dasturini tuzing.

II. Variantlar

№	Variantlar
1	Uchga uch determantni hisoblang
2	A matritsaga B matritsani qo'shing
3	A matritsani transponirlang
4	$N \times N$ birlik matritsa yarating
5	A matritsaning manfiy bo'lmanagan elementlarini chop qiling.
6	A kvadrat matritsaning determantini hisoblovchi dastur tuzing
7	Bir o'lchovli massivni Pufakcha usulida saralang
8	A matritsaning har bir satr elementlarning maximumini chop qiling
9	A matritsaning bosh diagonaldan pastda yotgan elementlarini 0 (nol) bilan almashtiring
10	A kvadrat matritsaga teskari matritsani toping
11	A matritsaning elementlari yig'indisi eng katta bo'lgan satr elementlarini chop qiling
12	Bosh diagonaldan yuqori va past qismi simmetrik bo'gan matritsa hosil qiling
13	Elementlari soni tub bo'lmagan a massiv elementlaridan matritsa hosil qiling
14	O'lchamlari bir xil bo'lgan A matritsa bosh diagonal elementlarini B matritsa bosh diagonal elementlari bilan mos ravishda almashtiring
15	A va B matritsa barcha elementlari yig'indisi katta bo'lgan matritsa elementlarini chop qiling
16	$N \times N$ matritsani ilon izi shaklida chiqaring. Misol: $A[3,3] = \begin{pmatrix} 1 & 2 & 3 \\ 6 & 5 & 4 \\ 7 & 8 & 9 \end{pmatrix}$
17	A matritsaning ikkinchi diagonaldan yuqorida yotgan elementlarining yig'indisini aniqlang
18	Berilgan n sonining kvadratigacha bo'lgan natural sonlarni $N \times N$ kvadrat

	matritsaga oxiridan boshlab ketma –ket joylashtiring.
19	Berilgan A matritsaning eng katta qiymati va eng kichik qiymati o'rinlarini almashtiring (agar ular soni bittadan ko'p bo'lsa birinchi uchragani bilan almashtiring)
20	A matritsaning ikki diagonalidan yuqorida joylashgan elementlari yig'indisini aniqlang

III. Hisobot mazmuni

1. Titul varaqasi: Fan nomi; ish mavzusi; talaba ismi sharifi; bajarish vaqti;
2. Masalaning qo'yilishi;
3. Ishdan maqsad;
4. Ikki o'lchovli massivlarning ta'rifi, qo'llanilishi;
5. Berilgan masala uchun algoritm ta'rifi;
6. Asosiy dastur matni;
7. Dastur bajarilish natijalari tahlili;
8. Natijalar listingi;
9. Foydalanilgan adabiyotlar.

IV. Uslubiy ko'rsatmalar

Masalaning qo'yilishi. A matritsaga B matritsani ko'paytirish.

Ishdan maqsad. Matritsani matritsaga ko'paytirishni ikki o'lchovli massivlardan foydalanib dasturiy ta'minotni tuzish.

Nazariy qism. Ikki o'lchovli massivlarning ta'rifi, e'loni va qo'llanilishi.

Ta'rif: Xotirada ketma – ket joylashgan bir xil turdagi qiymatlarga massiv deyiladi.

Odatda massivlarga zarurat, katta hajmdagi, lekin cheklangan miqdordagi va tartiblangan qiymatlarni qayta ishlash bilan bog'liq masalalarni yechishda yuzaga keladi. Dasturda ishlatiladigan har bir konkret massiv o'zining individual nomiga ega bo'lishi kerak. Bu nom to'liq o'zgaruvchi deyiladi, chunki uning qiymati massivning o'zi bo'ladi. Massivning har bir elementi massiv nomi, hamda kvadrat qavsga olingan va element sektori deb nomlanuvchi indeksni ko'rsatish orqali oshkor ravishda belgilanadi. Murojaat sintaksisi:

`<massiv nomi>[<indeks>]`

Bu ko'rinishga xususiy o'zgaruvchi deyiladi, chunki uning qiymati massivning alohida elementidir. C++ dasturlash tilida index doimo 0 dan boshlanadi va uning eng oxirgi qiymati massiv e'lonidagi uzunlikdan bittaga kam bo'ladi.

Massiv e'loni quyidagicha bo'ladi:

`<tur><nom>[<uzunlik>]={boshlang'ich qiymatlar}.`

Misol. **Int m[5]={1,3,-5,4,0};** bu yerda **m[0]=1; m[4]=0;**

C++ dasturlash tilida massivlar elementining turiga cheklovlar qo'yilmaydi, lekin bu turlar chekli o'lchamdagi ob'ektlarning turi bo'lishi kerak. Chunki kompilyator massivning xotiradan qancha joy egallashini hisoblay olishi kerak. Xususan, massiv komponentasi massiv bo'lishi muvofiq natijada matritsa deb nomlanuvchi ikki o'lchamli massiv hosil bo'ladi. Ikki o'lchovli massivlar quyidagicha e'lon qilinadi:

$\langle \text{tur} \rangle \langle \text{nom} \rangle [\langle \text{uzunlik} \rangle] [\langle \text{uzunlik} \rangle]$

Misol: **int a[3][3]={ 1, 2, 3; 0, -1, 2; -2, 0, 1 }** ko'rinishi:

$$a[i], [j] = \begin{pmatrix} 1 & 2 & 3 \\ 0 & -1 & 2 \\ -2 & 0 & 1 \end{pmatrix}.$$

Berilgan masala algoritmi. A matritsani B matritsaga ko'paytirish algoritmi. Matematikada berilgan A(m,n) matritsani B(n,k) matritsaga ko'paytirish imkoniyati mavjud. Ya'ni ikki matritsani bir biriga ko'paytirish vaqtida birinchi matritsaning ustunlari soni ikkinchi matritsaning satrlari soniga teng bo'lishi kerak, aks xolda ushbu berilgan matritsalarini bir – biriga ko'paytirish mumkin emas.

Misol:

$$A[2,3] = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix}. \quad B[3,3] = \begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}.$$

$C[2,3] = A[2,3] \times B[3,3].$ ya'ni

$$c[1,1] = a_{11} \cdot b_{11} + a_{12} \cdot b_{21} + a_{13} \cdot b_{31};$$

$$c[1,2] = a_{11} \cdot b_{12} + a_{12} \cdot b_{22} + a_{13} \cdot b_{32};$$

$$c[1,3] = a_{11} \cdot b_{13} + a_{12} \cdot b_{23} + a_{13} \cdot b_{33};$$

$$c[2,1] = a_{21} \cdot b_{11} + a_{22} \cdot b_{21} + a_{23} \cdot b_{31};$$

$$c[2,2] = a_{21} \cdot b_{12} + a_{22} \cdot b_{22} + a_{23} \cdot b_{32};$$

$$c[2,3] = a_{21} \cdot b_{13} + a_{22} \cdot b_{23} + a_{23} \cdot b_{33}.$$

Asosiy dastur matni.

```
[*] AxB.cpp
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int A[100][100], B[100][100], C[100][100];
5      int i, j, k, l, n, m;
6      cin >> n >> m >> k;
7      for (i=0; i<n; i++)
8          for (j=0; j<m; j++)
9          cin >> A[i][j];
10
11     for (i=0; i<m; i++)
12         for (j=0; j<k; j++)
13             cin >> B[i][j];
14     cout << endl << "natija:" << endl;
15     for (i=0; i<n; i++)
16         for (j=0; j<k; j++)
17         {
18             C[i][j]=0;
19             for (l=0; l<m; l++)
20                 C[i][j] += A[i][l] * B[l][j];
21         }
22     for (i=0; i<n; i++)
23         for (j=0; j<k; j++)
24             cout << C[i][j] << " ";
25     cout << endl;
26     return 0;
27 }
28
29
```

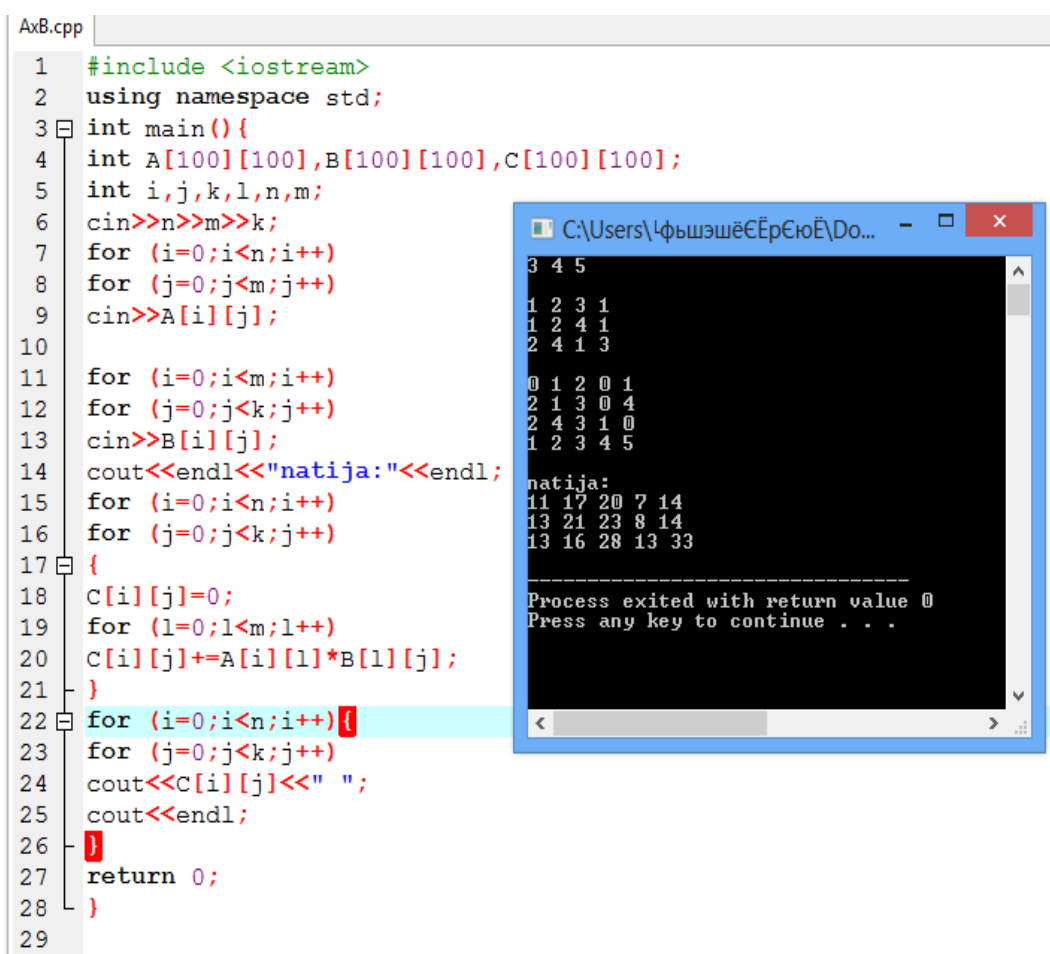
Dastur bajarilish natijalari tahlili.

Dasturning eng asosiy qismi:

```
for (i=0; i<n; i++)
for (j=0; j<k; j++)
{
C[i][j]=0;
for (l=0; l<m; l++)
C[i][j] += A[i][l] * B[l][j];
}
```

Dasturning mana shu qismida berilgan A matritsani B matritsaga ko'paytirishdan hosil bo'lgan qiymatlarni natijaviy C matritsaga ta'minlagan.

Natijalar listingi.



```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int A[100][100], B[100][100], C[100][100];
5     int i, j, k, l, n, m;
6     cin >> n >> m >> k;
7     for (i = 0; i < n; i++)
8         for (j = 0; j < m; j++)
9             cin >> A[i][j];
10
11     for (i = 0; i < m; i++)
12         for (j = 0; j < k; j++)
13             cin >> B[i][j];
14     cout << endl << "natija:" << endl;
15     for (i = 0; i < n; i++)
16         for (j = 0; j < k; j++)
17         {
18             C[i][j] = 0;
19             for (l = 0; l < m; l++)
20                 C[i][j] += A[i][l] * B[l][j];
21         }
22     for (i = 0; i < n; i++) {
23         for (j = 0; j < k; j++)
24             cout << C[i][j] << " ";
25         cout << endl;
26     }
27     return 0;
28 }
29
```

Output:

```
3 4 5
1 2 3 1
1 2 4 1
2 4 1 3
0 1 2 0 1
2 1 3 0 4
2 4 3 1 0
1 2 3 4 5
natija:
11 17 20 7 14
13 21 23 8 14
13 16 28 13 33
-----
Process exited with return value 0
Press any key to continue . . .
```

Foydalanilgan adabiyotlar.

1. C++ tilida programmalash asoslari, Sh. F. Madrahimov, S. M. Gaynazarov, Toshkent – 2009.
2. C/C++ tilida masalalar va topshiriqlar, Yusupov R. A., Mirzanova Sh, Iskandarova S. N. Samarqand – 2013.
3. C/C++ и MS Visual C++ 2010 для начинающих, Пахомов Б. И. Санктпетербург – 2011.

Laboratoriya ishi №2

Mavzu: Funksiyalar yaratish. Massivlar, funksiya va rekursiv funksiya.

Maqsad: Funksiyalar yaratish ko'nikmalarini oshirish. Massivlar ustida bajariladigan vazifalarni funksiya yordamida bajarish. Rekursiv funksiyalardan foydalanib dasturlar yaratish ko'nikmalarini oshirish.

I. Masalaning qo'yilishi. Funksiyalarni e'lon qiling. Variantda ko'rsatilgan misollarni funksiya yordamida dasturini tuzing.

II. Variantlar

№	Variantlar
1	Dastlabki n ta Fibonachi sonlarini aniqlash funksiyasini yarating
2	N! faktorialni hisoblaydigan rekursiv funksiya yarating
3	A matritsani B matritsaga ko'paytirish funksiyasini hosil qiling
4	Ikkita natural sonning eng kichik umumiy karralisini aniqlash funksiyasini yarating
5	Saralashning tanlash usuli funksiyasini yarating
6	Koordinatalar tekisligida $A(x_1, y_1)$ va $B(x_2, y_2)$ nuqtalar orasidagi masofani aniqlash funksiyasini hosil qiling
7	A matritsaning har bir ustun elementlari eng katta qiymatini B massivga ta'minlash funksiyasini yarating
8	Berilgan N sonining raqamlari yig'indisini hisoblovchi funksiya yarating
9	Berilgan N soni tub son ekanligini tekshiruvchi funksiya yarating
10	a va b sonlarining o'rnlarini almashtiruvchi swap(a,b) funksiyasini almashtirish(a,b) funksiyasi orqali yarating
11	Berilgan m va n sonlarining umumiy bo'luvchilari sonini aniqlaydigan funksiya yarating
12	Dastlabki hadi va ayirmasi berilgan arifmetik progressiyaning n – hadini aniqlovchi funksiya yarating
13	Dastlabki hadi va maxraji berilgan geometrik progressiyaning n ta hadi yig'indisini hisoblovchi funksiya yarating
14	a, b, c – koefisientlari berilgan kvadrat tenglamaning $a \cdot x^2 + b \cdot x + c = 0$ ildizlarini aniqlovchi funksiya yarating
15	NxN birlik matritsa hosil qiluvchi funksiya yarating
16	N haqiqiy sonini yahlitlovchi funksiya yarating
17	N soniyani soat, daqiqa va soniyalarga ajratadigan funksiya yarating(misol: 3723 soniya: 1 soat 2 daqiqa 3 soniyaga teng)

18	Berilgan N ta massivning maximuni aniqlovchi funksiya yarating
19	O'nlik sanoq sistemasida berilgan n sonini 2 lik sanoq sistemasiga aylantiradigan funksiya yarating
20	Fibonachi sonlarining n – hadini aniqlovch rekursiv funksiya yarating.

III. Hisobot mazmuni

1. Titul varaqasi: Fan nomi; ish mavzusi; talaba ismi sharifi; bajarish vaqti;
2. Masalaning qo'yilishi;
3. Ishdan maqsad;
4. Funksiyalarning ta'rifi, qo'llanilishi;
5. Berilgan masala uchun algoritm ta'rifi;
6. Asosiy dastur matni;
7. Dastur bajarilish natijalari tahlili;
8. Natijalar listingi;
9. Foydalanilgan adabiyotlar.

IV. Uslubiy ko'rsatmalar

Masalaning qo'yilishi. Berilgan ikki natural sonning eng katta umumiy bo'luvchisini aniqlaydigan oddiy va rekursiv funksiya yarating.

Ishdan maqsad. Oddiy va rekursiv funksiyalar bilan ishlash ko'nikmalarini oshirish. Ikki sonning eng katta umumiy bo'luvchisini aniqlashda Evklid algoritmidan foydalanish. Ikki sonning eng katta umumiy bo'luvchisini rekursiyadan foydalanib aniqlash.

Nazariy qism. Dasturda ishlatiladigan har qanday funksiya e'lon qilinishi shart. Odatda funksiyalar e'loni sarlavha fayllarda e'lon qilinadi va #include direktivasi yordamida dastur matniga qo'shiladi. Funksiya e'lonini funksiya prototipi tavsiflaydi. Funksiya prototipi quyidagi ko'rinishda bo'ladi:

<qaytaruvchi qiymat turi> <funksiya nomi> (<parametrlar ro'yhati>);

Bu yerda <qaytaruvchi qiymat turi> – funksiya ishlashi natijasida u tomonidan qaytaradigan qiymat turi. Agar qaytariladigan qiymat turi ko'rsatilmagan bo'lsa, kelishuv bo'yicha qaytaradigan qiymat turi int deb hisoblanadi. <parametrlar ro'yhati> – vergul bilan ajratilgan funksiya parametrlarining turi va nomlari ro'yhati. Parametr nomini yozmasa ham bo'ladi. Ro'yhat bo'sh bo'lishi ham mumkin. Funksiya prototiplariga misollar:

int almashtirish (int, int);

double min (double x, double y);

void func ();

void chop_etish (void);

Funksiya prototipi tushirib qoldirilishi mumkin, agar dastur matnida funksiya aniqlanishi uni chiqaradigan funksiyalar matnidan oldin yozilgan bo'lsa. Lekin bu holat yaxshi uslub hisoblanmaydi, ayniqsa o'zaro bir – biriga murojaat qiluvchi funksiyalarni e'lon qilishda muammolar yuzaga kelishi mumkin.

Funksiyalar aniqlanishi – Funksiya sarlavhasi va figurali qavsga ('{', '}') olingan qandaydir amaliy mazmunga ega bo'lgan tandan iborat bo'ladi. Agar funksiya qaytaruvchi turi void turidan farqli bo'lsa, uning tanasida albatta mos turdagi parametrga ega return operatori bo'lishi shart. Funksiya tanasida bittadan ortiq return operatori bo'lishi mumkin. Ularning ihtiyotiy birortasini bajarish orqali funksiyadan chiqib ketiladi. Agar funksiyaning qiymati dasturda ishlatilmaydigan bo'lsa, funksiyadan chiqish uchun parametrsiz return operatori ishlatilishi mumkin yoki umuman return ishlatilmaydi. Oxirgi holda funksiyadan chiqish – oxirgi yopiluvchi qavsga yetib kelganda ro'y beradi.

Funksiyani chaqirish uchun uning nomi va undan keyin qavs ichida argumentlar ro'yhati beriladi:

< funksiya nomi >(< argument1 >, < argument2 >, ... , < argumentN >);

Bu yerda har bir < argument > - funksiya tanasiga uzatiladigan va keyinchalik hisoblash jarayonida ishlatiladigan o'zgaruvchi, ifoda yoki o'zgarmasdir. Argumentlar ro'yhati bo'sh bo'lishi mumkin. Funksiyalar ham o'z tanasida boshqa funksiyalarni, o'zini ham chaqirishi mumkin. O'z tanasida o'zini chaqiradigan funksiyalarga *rekursiv funksiyalar* deyiladi. Misol:

```
#include <iostream>
using namespace std;
int max ( int a, int b) {
if (a > b) return a; else
return b; }
int main(){
int x,y;
cin>>x>>y;
cout<<max(x,y);
return 0;
}
```

Ushbu misolda ikkita sonning kattasini aniqlash funksiyasi hosil qilindi va asosiy funksiyada maximumni topadigan funksiyadan foydalangan holda aniqlayapti.

Berilgan masala algoritmi. Matematikadan ma'lum berilgan sonlarning eng katta umumiy bo'luvchisini aniqlash aytilishidan ma'lumki berilgan sonlarning barcha umumiy bo'luvchilari ya'ni ushbu natural sonlarning hammasiga bo'linadigan bo'luvchilari ichidan eng kattasi berilgan sonlarning eng katta umumiy bo'luvchisi hisoblanadi. Eng katta umumiy bo'luvchini aniqlashning algoritmlari mavjud. Misol: EKUB(16,56)=?

16	2	56	2	
8	2	28	2	$16 = 2^4$
4	2	14	2	$56 = 2^3 \cdot 7$
2	2	7	7	EKUB(16,56) = $2^3 = 8$;
1		1		

Bu yerda 16 va 56 ni tub ko'paytuvchilarga ajratiladi va ajratilgan tub ko'paytuvchilar ichidan asosi bir xil bo'lgan tub sonlarning darajasi kichigi 2^3 olinadi. Agar bunday holat bo'lmasa 1 ni qabul qiladi.

Ushbu misolning yana bir usuli Evklid algoritmi hisoblanadi. Bu algoritmda berilgan ikkita sonning kattasidan kichigi ayiriladi va kattasining o'rniga qo'yiladi. Bu holat berilgan ikki son bir – biriga teng bo'lib qolguncha davom etadi. Ular teng bo'lgandagi qiymat berilgan ikki sonning eng katta umumiy bo'luvchisi hisoblanadi. Misol: EKUB(16,56)=?

$a=16$, $b=56$; bu yerda $b>a$, shuning uchun $b=b-a$; $b=56-16=40$; yana $b>a$ va $b=b-a=40-16=24$; yana $b>a$ va $b=b-a=24-16=8$; endi $a>b$ $a=a-b=16-8=8$; va shu joyda ikki son bir – biriga teng bo'lgani uchun takrorlash to'xtatiladi va natija ya'ni ikki sonning eng katta umumiy bo'luvchisi EKUB(a,b)=8.

Asosiy dastur matni.

1. Eng katta umumiy bo'luvchini funksiya yordamida aniqlash dasturi:

```
ekub.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b) {
4      if (a>b) swap(a,b);
5      int i,k=1;
6      for (i=1;i<=a;i++){
7          if (a%i==0 && b%i==0) k=i;
8      }
9      return k;
10 }
11 int main() {
12     int a,b,c;
13     cin>>a>>b;
14     c=ekub(a,b);
15     cout<<c;
16     return 0;
17 }
```

2. Eng katta umumiy bo'luvchini Evklid algoritmini funksiyada qo'llash orqali aniqlash dasturi:

```
[*] evklid.cpp
1  #include <iostream>
2  using namespace std;
3  int evklid(int x,int y)
4  { while (x!=y){
5      if (x>y) x=x-y; else y=y-x;
6  }
7      return x;
8  }
9  int main(){
10     int a,b;
11     cin>>a>>b;
12     cout<<evklid(a,b) ;
13     return 0;
14 }
15
```

3. Eng katta umumiy bo'luvchini rekursiv funksiya yordamida aniqlash dasturi:

```
[*] ekub_rekursiya.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b){
4      if (a==b) return a; else
5      if (a>b) return ekub(a-b,b); else
6      return ekub(a,b-a);
7  }
8  int main(){
9      int x,y;
10     cin>>x>>y;
11     cout<<ekub(x,y) ;
12     return 0;
13 }
```

Dastur bajarilish natijalari tahlili.

1. Eng katta umumiy bo'luvchini funksiya yordamida aniqlash dasturi:

```
ekub.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b) {
4      if (a>b) swap(a,b);
5      int i,k=1;
6      for (i=1;i<=a;i++){
7          if (a%i==0 && b%i==0)    k=i;
8      }
9      return k;
10 }
```

Bu funksiya birlan boshlab ikki sonning kichigigacha bo'lgan natural sonlarni ketma – ket ikkala songa bo'linishini tekshirib chiqiladi. Agar bajarilsa k ga ta'minlab oladi. Oxirgi k ga ta'minlangan son ikki sonning eng katta umumiy bo'luvchisi hisoblanadi.

2. Eng katta umumiy bo'luvchini Evklid algoritmini funksiya qo'llash orqali aniqlash dasturi:

```
[*] evklid.cpp
1  #include <iostream>
2  using namespace std;
3  int evklid(int x,int y)
4  { while (x!=y){
5      if (x>y) x=x-y; else y=y-x;
6  }
7      return x;
8  }
```

Bu funksiya eng katta umumiy bo'luvchini aniqlashda Evklid algoritmidan foydalangan, ya'ni, berilgan ikki son qachonki ular teng bo'lmaguncha kattasidan kichigi ayrilib kattasining joyiga ta'minlanadi. Eng katta umumiy bo'luvchi ikkalasi teng bo'lgandagi holat hisoblanadi.

3. Eng katta umumiy bo'luvchini rekursiv funksiya yordamida aniqlash dasturi:

```

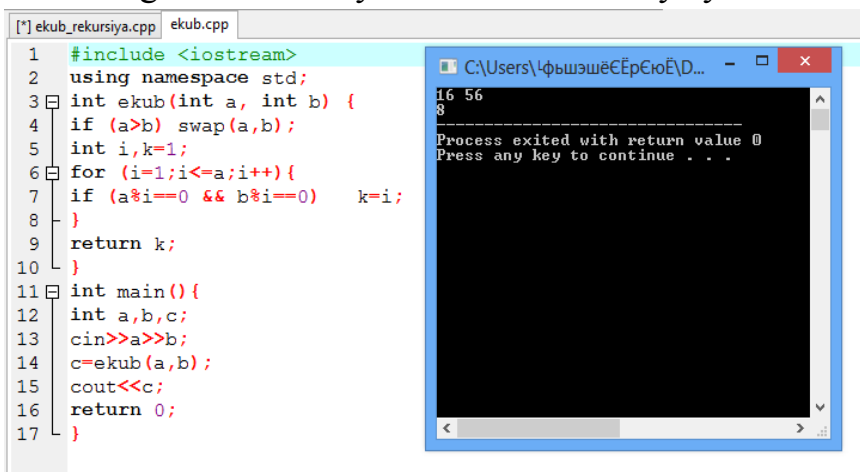
[*] ekub_rekursiya.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b){
4      if (a==b) return a; else
5          if (a>b) return ekub(a-b,b); else
6              return ekub(a,b-a);
7  }

```

Ushbu rekursiv funksiyada ham Evklid algoritmidan foydalangan, ya'ni, ikki sonning qaysi biri katta bo'lsa kattasidan kichigi ayrilib va kichigi ikkalasiga yana shu funksiyaga qayta murojaat qilinadi.

Natijalar listingi.

1. Eng katta umumiy bo'luvchini funksiya yordamida aniqlash dasturi:



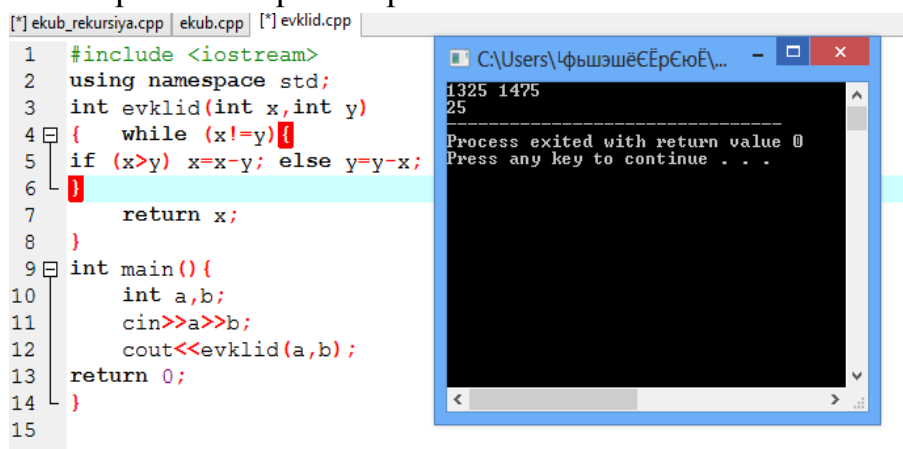
```

[*] ekub_rekursiya.cpp  ekub.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b) {
4      if (a>b) swap(a,b);
5      int i,k=1;
6      for (i=1;i<=a;i++){
7          if (a%i==0 && b%i==0) k=i;
8      }
9      return k;
10 }
11 int main(){
12     int a,b,c;
13     cin>>a>>b;
14     c=ekub(a,b);
15     cout<<c;
16     return 0;
17 }

```

Output window: 16 56
8
Process exited with return value 0
Press any key to continue . . .

2. Eng katta umumiy bo'luvchini Evklid algoritmini funksiyada qo'llash orqali aniqlash dasturi:



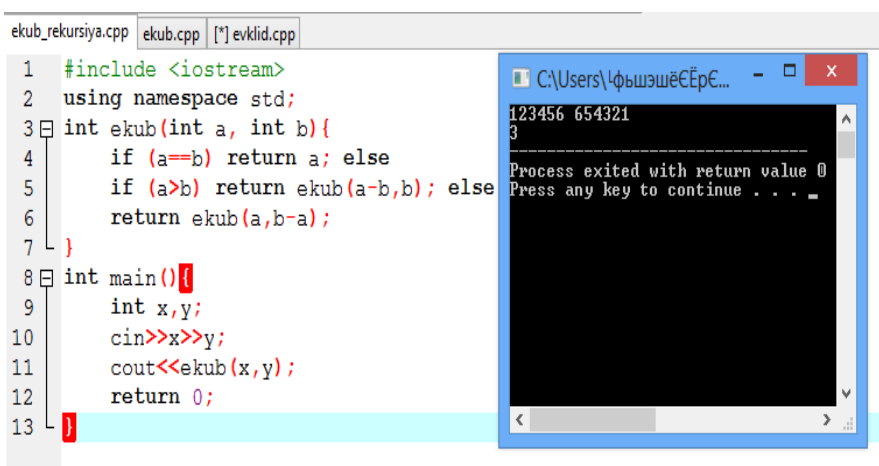
```

[*] ekub_rekursiya.cpp  ekub.cpp  [*] evklid.cpp
1  #include <iostream>
2  using namespace std;
3  int evklid(int x,int y)
4  {   while (x!=y){
5      if (x>y) x=x-y; else y=y-x;
6  }
7      return x;
8  }
9  int main(){
10     int a,b;
11     cin>>a>>b;
12     cout<<evklid(a,b);
13     return 0;
14 }
15

```

Output window: 1325 1475
25
Process exited with return value 0
Press any key to continue . . .

3. Eng katta umumiy bo'luvchini rekursiv funksiya yordamida aniqlash dasturi:



```
ekub_rekursiya.cpp  ekub.cpp  [*] evklid.cpp
1  #include <iostream>
2  using namespace std;
3  int ekub(int a, int b){
4      if (a==b) return a; else
5      if (a>b) return ekub(a-b,b); else
6      return ekub(a,b-a);
7  }
8  int main()
9  {
10     int x,y;
11     cin>>x>>y;
12     cout<<ekub(x,y);
13 }
```

Console output:

```
123456 654321
3
Process exited with return value 0
Press any key to continue . . .
```

Foydalanilgan adabiyotlar.

1. C++ tilida programmalash asoslari, Sh. F. Madrahimov, S. M. Gaynazarov, Toshkent – 2009.
2. C/C++ tilida masalalar va topshiriqlar, Yusupov R. A., Mirzanova Sh, Iskandarova S. N. Samarqand – 2013.
3. C/C++ и MS Visual C++ 2010 для начинающих, Пахомов Б. И. Санктпетербург – 2011.

Laboratoriya ishi №3

Mavzu: Strukturalar va satrlar.

Maqsad: Strukturalar va satrlar bilan ishlash ko'nikmalarini oshirish. Strukturalarni yaratish va qayta ishlash. string.h kutubxonasi standart funksiyalari bilan ishlash ko'nikmalarini oshirish.

I. Masalaning qo'yilishi. Variantda ko'rsatilgan misollarni yechishda zarur bo'ladigan turlarni strukturalar yordamida hosil qilib dasturini tuzing.

II. Variantlar

№	Variantlar
1	Korxona xodimlari familiyasi, ismi, telefon nomerini kiriting. Kiritilgan familiya bo'yicha uning ismi va telefon nomerini aniqlovchi dastur tuzing. Korxona xodimlari soni n.
2	Guruhdagi n ta talabaning ismi – sharifi, tug'ilgan yili, yashash joyi, telefon nomerini kiriting. Talabalarni ismi – sharifi bo'yicha saralab chiqaring.
3	N ta kitob haqidagi ma'lumotlarni kiriting. Kiritiladigan ma'lumotlar kitob nomi, muallifi, varaqlar soni, nashr qilingan yili. Berilgan kitoblardan kiritilgan muallifning belgilangan yilgacha chiqarilgan kitoblari ro'yhatini ekranga chiqaruvchi dastur tuzing.
4	N ta korxona xodimlari ma'lumotlari(familiyasi, ismi, lavozimi, bir kunlik ish xaqi, ishlagan kunlari)ni kiriting. Hodimning oldingi oydagi ishlagan kuniga maoshini hisoblovchi dastur tuzing.(10% miqdorda rag'batlantirish qo'shib berilsin).
5	Guruhdagi n ta talabaning ismi – sharifi, tug'ilgan yili, yashash joyi, kursi, o'tgan semester bo'yicha natijaviy bahosi, telefon nomerini kiriting. Berilgan ma'lumotlar ichidan kiritilgan kurs bo'yicha talabalar ro'yhatini chiqaring.
6	Guruhdagi n ta talabaning ismi – sharifi, tug'ilgan yili, yashash joyi, kursi, o'tgan semester bo'yicha natijaviy bahosi, telefon nomerini kiriting. Berilgan ma'lumotlar ichidan kiritilgan baho bo'yicha talabalar ro'yhatini chiqaring.
7	O'zbekiston oliy liga futbol komandalarining quyidagi ma'lumotlarini

	kiriting: komanda nomi, o'yinlar soni, yutgan o'yinlar soni, duranglar soni, mag'lubiyatlar soni, urgan to'plar soni, o'tkazgan to'plar soni, to'plagan ochkosi. Oliy liga komandalarini futbol parametrlariga asoslangan holda saralangan ro'yhatini chiqaring.
8	Fazoda dekart kordinatalar sistemasida n ta nuqtaning kordinatalari berilgan. Berilgan nuqtalar ichidan biri boshqasidan eng katta masofaga farq qiluvchi nuqtalarning kordinatalarini aniqlovchi dastur tuzing.
9	Probel bilan ajratilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi katta harflar sonini chiqaruvchi dastur tuzing.
10	Satr berilgan. Satrdagi ortiqcha probellarni olib tashlovchi dastur tuzing
11	Satr berilgan. Satrdagi so'zlar sonini hisoblovchi dastur tuzing.
12	Lotin harflaridan iborat satr berilgan. Satrdagi har bir belgini k ta keyinga suradigan dastur tuzing. Lotin alfabiti 26 belgidan iborat va ularni yopiq deb hisoblang.
13	Futbol komandasi o'yinchilari ro'yhati berilgan. O'yinchilar ismi – sharifi, yoshi, raqami, o'ynash pozitsiyasi. Kiritilgan yoshdan kichik o'yinchilar ro'yhatini chiqaruvchi dastur tuzing
14	Ikkita satr berilgan. Birinchi satrda uchramagan ikkinchi satr elementlari sonini aniqlang. Katta kichik harflarni bir biriga teng hisoblang
15	Ikkita satr berilgan. Ikkinchi satrni birinchi satr ichidan qidiring
16	Satrdagi ismlar vergul bilan ajratilgan holda berilgan. Satrda berilgan ismlarni ustun shaklida chiqaring
17	Satr berilgan. Satr davomidan satr teskarisini qo'shing
18	Satr berilgan. Satrdagi butun sonlar miqdorini aniqlang.
19	1 – kurs guruhlaridan birining II – semestr fanlaridan olgan baholari jurnal bo'yicha kiritilgan. Talabalarni olgan natijaviy ballari bo'yicha saralang.
20	O'nlik sanoq sistemasida berilgan n ta sonni k – sanoq sistemasiga o'tkazing

III. Hisobot mazmuni

10. Titul varaqasi: Fan nomi; ish mavzusi; talaba ismi sharifi; bajarish vaqti;
11. Masalaning qo'yilishi;
12. Ishdan maqsad;
13. Struktura va satrlarning ta'rifi, qo'llanilishi;
14. Berilgan masala uchun algoritmi ta'rifi;
15. Asosiy dastur matni;
16. Dastur bajarilish natijalari tahlili;
17. Natijalar listingi;

18.Foydalanilgan adabiyotlar.

IV. Uslubiy ko'rsatmalar

Masalaning qo'yilishi. N ta talaba ismi – sharifi, I – semester fanlaridan olgan natijaviy baholari va telefon nomerlari vergul bilan ajratilgan holda berilgan. Har bir keyingi talaba haqidagi ma'lumot keyingi satrda kiritiladi. Talabalarni natijaviy baholari bo'yicha saralang.

Ishdan maqsad. Satrlar ustida amallar bajarishda funksiyalardan foydalanish. Strukturalar bilan ishlash ko'nikmalarini oshirish. Strukturalar orqali yangi turlar yaratish, satrda berilgan har xil turlarni yangi strukturada yaratilgan turlar bilan ajarish.

Nazariy qism. Bir yoki har xil turdagi berilmalar jamlanmasi struktura deb nomlanadi. Struktura foydalanuvchi tomonidan aniqlangan berilganlarning yangi turi hisoblanadi. Struktura quyidagicha aniqlanadi:

```
Struct <struktura nomi> {  
    < tur1 > < nom1 >  
    < tur2 > < nom2 >  
    ...  
    < tur3 > < nom3 >  
}
```

Bu yerda < struktura nomi > - struktura ko'rinishida yaratilayotgan yangi tur nomi, “< *tur_i* > < *nom_i* >” strukturaning i – maydonining turi va nomi e'loni. Boshqacha aytganda, struktura e'lon qilingan o'zgaruvchilardan tashkil topadi. Unga har xil turdagi berilganlarni o'z ichiga oluvchi qobiq deb qarash mumkin. Talaba haqidagi berilganlarni o'z ichiga oluvchi struktura turining e'lon qilinishini ko'raylik.

```
Struct talaba {  
    char fish[30];  
    unsigned int tug_yil;  
    unsigned int kurs;  
    char yunalish[50];  
    float reyting;  
    unsigned char jinsi[5];  
    char manzili[50];  
    bool status;  
};
```

Unga murojaat quyidagicha:

Talaba a;

a.fish="Botirov Qodir";

a.tug_yil=1994;

a.kurs=1;

a.yunalish="Kompyuter Injiniringi";

a.reyting=87.3;

a.jinsi="erkak";

a.manzili="Samarqand, Shoxruh Mirzo 12/5";

a.status=true;

Standart C++ tili ikki xil turdagi belgilar majmuasini qo'llab quvvatlaydi. Birinchi toifaga "tor" belgilar deb nomlanuvchi 8 – bitli belgilar majmuasi kiradi, ikkinchisiga 16 – bitli "keng" belgilar kiradi. C++ tilida satr uchun maxsus tur aniqlanmagan. Satr char turidagi belgilar massivi sifatida qaraladi va bu belgilar ketma – ketligi satr terminatori deb nomlanuvchi 0 kodli belgi bilan tugaydi '\0'. Satr o'zgaruvchilarni e'lon qilinishida boshlang'ich qiymatlarni qabul qilishi mumkin. Bu holda kompilyator avtomatik ravishda satr uzunligini hisoblaydi va satr oxiriga terminatorni qo'shib qo'yadi:

Char hafta[]="shanba"; char hafta[]={ 's', 'h', 'a', 'n', 'b', 'a', '\0' };;

Satr qiymatini o'qishda oqimli o'qish operatori ">>" o'rniga *getline()* funksiyasini ishlatgan ma'qul hisoblanadi, chunki oqimli o'qishda probel inkor qilinadi. *Getline()* funksiyasi ikkita parametrga ega bo'lib, birinchisi o'qish amalga oshirilayotganda satrga ko'rsatkich, ikkinchi parametrdan esa o'qilishi kerak bo'lgan belgilar soni ko'rsatiladi. Satrni *getline()* funksiyasi orqali o'qishga misol:

#include <iostream>

Using namespace std;

int main() {

char a[6];

cout<<"Satrni kiriting: "<<"\n";

cin.getline(a,6);

cout<<"Siz kiritgan satr: "<<a;

return 0;

}

Dasturda kiritilgan a satrni 5 ta belgini qabul qilishi mumkin, ortiqchalari tashlab yuboriladi.

Satr uzunligini aniqlash funksiyalari.

1. Sintaksisi: **size_t strlen(const char* string)** ushbu funksiya satr uzunligini nol terminatorsiz aniqlaydi.

2. Sintaksisi: **size_t sizeof(const char* string)** ushbu funksiya satr xaqiqiy uzunligini, ya'ni, nol terminator bilan birgalikdagi uzunligini hisoblaydi.

Misol:

```
#include <iostream>
#include <string.h>
Using namespace std;
int main()
{
    char Str[]="1234567890";
    cout <<"strlen(Str)="<<strlen(Str)<<endl;
    cout<<"sizeof(Str)="<<sizeof(Str)<<endl;
    return 0;
}
```

Natija:

strlen(Str)=10

sizeof(Str)=11.

Odatda *sizeof()* funksiyasidan *getline()* funksiyasining ikkinchi argumenti sifatida ishlatiladi. **Cin.getline(satr,sizeof(satr));**

Satrlarni nusxalash.

1. *strcpy()* funksiyasi prototipi:

char* strcpy(char* str1, const char* str2);

misol:

char satr[20]="hello world", str2[20]; strcpy(str2, str1);

bu yerda str1 qiymatlarini mos ravishd str2 ga nusxalaydi. Natijada **str2="hello world"** hosil bo'ladi.

2. *strncpy()* funksiyasi prototipi:

char* strncpy(char*str1, const char*str2,size_t num);

misol:

char Uzun_str[]="01234567890";

char Qisqa_str[]="ABCDEF";

strncpy(Qisqa_str,Uzun_str,3);

Natijada **qisqa_str="012DEF"** hosil bo'ladi.

Satrlarni ulash.

1. *strcat()* funksiyasi sintaksisi quyidagi ko'rinishga ega:

char* strcat(char* str1, const char* str2)

misol:

char satr[80];

strcpy(satr,"Hello world");

strcat(satr,"salom dunyo");

natijada **str="Hello world salom dunyo"** hosil bo'ladi.

2. *strncat()* funksiyasi sintaksisi:

char* strncat(char* str1,const char* str2,size_t num)

misol:

```
char str1[50]="hello world";  
char str2[20]="salom dunyo";  
strncat(str1,str2,5);  
cout<<str1;
```

natija: **helloworldsalom.**

Satrlarni solishtirish.

1. *Strcmp()* funksiya sintaksisi:

int strcmp(const char* str1, const char* str2)

1. $i < 0$ – agar str1 satri str2 satridan kichik bo'lsa;
2. $i = 0$ – agar str1 satri str2 satriga teng bo'lsa;
3. $i > 0$ – agar str1 satri str2 satridan katta bo'lsa;

misol:

```
char satr1[80]="Programmalash tillari:C++,pascal.“;  
char satr2[80]="Programmalash tillari:C++,Pascal.“;  
int i;  
i=strcmp(satr1,satr2);
```

bu yerda str1 ni str2 satri har bir elementi bilan solishtirib chiqadi. Birinchi uchraganiga qarab xulosa qiladi. Misolda natija **i=1** olinadi.

2. *Strncmp()* funksiya sintaksisi:

int strncmp(const char*str1,const char*str2,size_t num);

bu funksiyda str1 va str2 satrlarining dastlabki n ta belisini tekshiradi.

Satrdagi harflar registrini almashtirish.

Berilgan satrdagi kichik harflarni bosh harflarga yoki teskarisiga almashtirishga mos ravishda *_strupr()* va *strlwr()* **char* _strupr(char* str);** funksiyalar nomidagi tagchiziq '_' bo'lmasligi mumkin.

_strlwr() funksiyasi sintaksisi:

char* _strlwr(char* str);

misol:

```
char str[]="10 TA KATTA HARFLAR";  
_strlwr(str);  
cout<<str;
```

natijada ekranda **10 ta kichik harflar** satri chop etiladi. *_strupr()* huddi *_strlwr()* funksiyasidek ishlaydi, lekin satrdagi kichik harflarni katta harflarga almashtiradi.

Programmalash amaliyotida belgilarni qaysidir oraliqqa tegishli ekanligini bilish zarur bo'ladi. Buni "**ctype.h**" sarlavha faylida e'lon qilingan funksiyalar yordamida aniqlash mumkin. Quyida ularning bir qismi tavsifi keltirilgan:

isalpha() - belgini harf (true) yoki yo'qligini (false) aniqlaydi;

isalnum() – belgi raqam yoki harf (true) yoki yo'qligini (false) aniqlaydi;

isdigit() - belgini raqamlar diapazoniga tegishli (true) yoki yo'qligini (false) aniqlaydi;

isascii() - belgini kodi 0...127 oralig'ida (true) yoki yo'qligini aniqlaydi;

Misol:

```
#include <iostream>
```

```
#include <ctype.h>
```

```
#include <string.h>
```

```
using namespace std;
```

```
int main(){
```

```
    string a;
```

```
    cin>>a;
```

```
    for (int i=0;i<a.length();i++)
```

```
    { if (isalpha(a[i])) cout<<a[i]<<" - isalpha - harf"<<endl;
```

```
      if ( isalnum(a[i])) cout<<a[i]<<" - isalnum - harf, raqam"<<endl;
```

```
      if (isdigit(a[i])) cout<<a[i]<<" - isdigit - raqamlar"<<endl;
```

```
      if (isascii(a[i])) cout<<a[i]<<" - isascii - 0 va 127 oraliqda"<<endl;
```

```
      if (iscntrl(a[i])) cout<<a[i]<<" - iscntrl - boshqaruv belgisi"<<endl;
```

```
      if (ispunct(a[i])) cout<<a[i]<<" - ispunct - punktuatsiya belgisi"<<endl; }
```

```
    return 0; }
```

Natija:

```
C:\Users\фьшэшëĖĖрĖюĖ\Documents\C++\belgiacsii.exe
s!#"0?φ12
s - isalpha - harf
s - isalnum - harf, raqam
s - isascii - 0 va 127 oraliqda
! - isascii - 0 va 127 oraliqda
! - ispunct - punktuatsiya belgisi
# - isascii - 0 va 127 oraliqda
# - ispunct - punktuatsiya belgisi
" - isascii - 0 va 127 oraliqda
" - ispunct - punktuatsiya belgisi
? - isascii - 0 va 127 oraliqda
? - ispunct - punktuatsiya belgisi
1 - isalnum - harf, raqam
1 - isdigit - raqamlar
1 - isascii - 0 va 127 oraliqda
2 - isalnum - harf, raqam
2 - isdigit - raqamlar
2 - isascii - 0 va 127 oraliqda

-----
Process exited with return value 0
Press any key to continue . . .
```

Satrnı teskari tartiblash.

Strrev() (*char* strrev(char* str)*) - satrni teskari tartiblash;

Misol:

Char str[]="salom"; cout<<strrev(str); natija: **molas.**

Satrdagi belgini izlash funksiyalari.

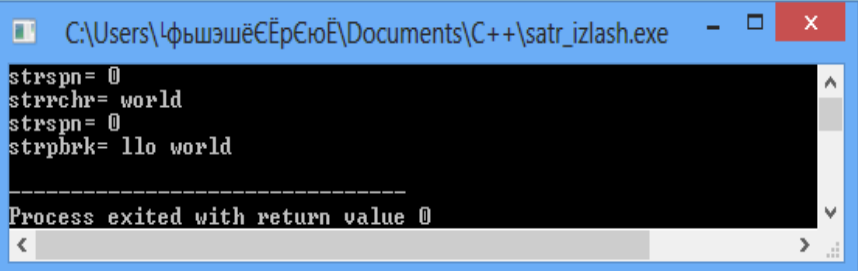
1. *Strchr()* (*char* strchr(const char* string, char c);*) - satrdagi izlagan belgi bor yoki yo'qligini tekshiradi. Belgini satr boshidan izlaydi. Agar belgi uchrasa belgining satrdagi o'rnini chiqaradi, aks holda NULL qiymat qaytaradi. *Strrchr()* funksiyasi huddi shu amalni satr oxiridan teskari tartibda izlaydi.
2. *Strspn()* (*size_t strspn(const char* str1, const char* str2);*) - str1 satridagi belgilarni str2 satrdagi belgilar bilan solishtiradi. Agar str1 satrdagi birorta belgi str2 satrdagi uchramasa o'sha belgi indeksini qaytaradi. Agar hamma belgi uchrasa str1 satr uzunligidan bitta ko'p qiymatni qaytaradi.
3. *Strcspn()* (*size_t strcspn(const char* str1, const char* str2);*) - str1 satrdagi belgilarni str2 satridagi belgilar ichidan izlaydi. Ushbu funksiya str1 satridagi birinchi str2 satridagi uchragan belgini qaytaradi. Bunday belgi bo'lmasa str1 satr uzunligidan bitta ko'p qiymatni funksiyaga qaytaradi.
4. *Strpbrk()* (*char* strpbrk(const char* str1, const char* str2);*) - satr1 satridagi satr2 satriga kiruvchi birinchi belgini izlaydi. Str1 satridagi birinchi uchragan belgidan davomini funksiyaga qaytaradi.

Misol:

```

sotr_izlash.cpp
1  #include <iostream>
2  #include <string.h>
3  #include <ctype.h>
4  using namespace std;
5  int main() {
6  char sotr1[]="hello world";
7  char sotr2[]="salom dunyo";
8  int index=strspn(sotr1,sotr2); cout<<"strspn= "<<index<<endl;
9  char* psotr; psotr=strrchr(sotr1,'w'); cout<<"strrchr= "<<psotr<<endl;
10 int n=strspn(sotr1,sotr2); cout<<"strspn= "<<n<<endl;
11 char* k; k = strpbrk(sotr1,sotr2); cout<<"strpbrk= "<<k<<"\n";
12 return 0;

```



Satr qismlarini izlash funksiyalari.

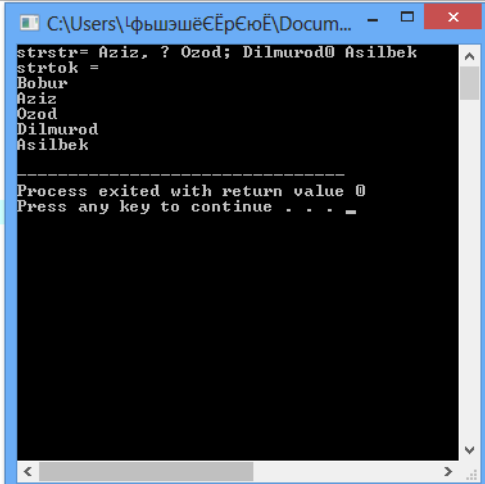
1. *Strstr()* (*char* strstr(const char* str, const char* substr);*) – ushbu funksiya str satrdagi substr qism satrni izlaydi. Agar substr qism satr str satr ichida mavjud bo'lmasa NULL qiymat qaytaradi.
2. *Strtok()* (*char* strtok(char* str, const char* delim);*) - ushbu funksiya str satrida delim ro'yhatida berilgan ajratuvchilar oralig'iga olingan satr qismlarini ajratib olish imkonini beradi.

Misol:

```

strtokk.cpp
1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main()
5  {
6  char Ism[]="Bobur, Aziz, ? Ozod; Dilmurod0 Asilbek";
7  char substr[]="Aziz";
8  cout<<"strstr= "<<strstr(Ism,substr)<<endl;
9  cout<<"strtok = "<<endl;
10 char Ajratuvchi[]=" ;,!?.0123456789";
11 char* Satrdagi_ism;
12 Satrdagi_ism=strtok(Ism,Ajratuvchi);
13 if(Satrdagi_ism) cout<<Satrdagi_ism<<"\n";
14 while(Satrdagi_ism)
15 {
16 Satrdagi_ism=strtok(NULL,Ajratuvchi);
17 if(Satrdagi_ism) cout<<Satrdagi_ism<<"\n";
18 }
19 return 0;
20 }

```



Turlarni o'zgartirish funksiyalari.

1. *Atoi()* (*int atoi(const char* ptr);*) - ushbu funksiya ptr satrini boshidan birinchi raqamdan tashqari belgi uchraguncha songa aylantiradi. Hammasi son bo'lsa satr oxirigacha songa aylantiradi, aks holda birinchi belgi raqam

bo'lmasa 0 qiymat qaytaradi. *atol()* funksiyasi huddi *atoi()* funksiyasidek ishlaydi, faqat *long* turiga aylantiradi.

2. *Atof()* (***double atof (const char* ptr);***) – ushbu funksiya yuqoridagi fuksiyaga o'xshash faqat butun turga emas, balki xaqiqiy turga aylantiradi.

The screenshot shows a C++ program that uses `atoi` and `atof` to convert a string to integer and double respectively. The code is as follows:

```

1 #include <stdlib.h>
2 #include <iostream>
3 using namespace std;
4 int main()
5 {
6     char str[]="15.1325salom1325";
7     int i=atoi(str);
8     double j=atof(str);
9     cout<<"atoi="<<i<<endl<<"atof="<<j<<endl;
10    return 0;
11 }

```

The output window shows the following results:

```

atoi=15
atof=15.1325

-----
Process exited with return value 0
Press any key to continue . . .

```

3. *Itoa()* (***char* itoa(int num, char *str, int radix);***) – Ushbu funksiya **num** sonini **radix** sanoq sistemasiga o'tkazib **str** satriga satr ko'rinishida ta'minlaydi. *Ltoa()* funksiyasi ham *Itoa()* funksiyaidek ishlaydi.

The screenshot shows a C++ program that uses `itoa` to convert an integer to a string in different bases. The code is as follows:

```

1 #include <iostream>
2 #include <stdlib.h>
3 using namespace std;
4 int main()
5 {
6     char satr2[20],satr8[15],satr10[10],satr16[5];
7     int son=15;
8     ltoa(son,satr2,2);
9     itoa(son,satr8,8);
10    itoa(son,satr10,10);
11    itoa(son,satr16,16);
12    cout<<"Son kurinishlari"<<endl;
13    cout<<"2 sanoq sistemasida : "<<satr2<<endl;
14    cout<<"8 sanoq sistemasida : "<<satr8<<endl;
15    cout<<"10 sanoq sistemasida: "<<satr10<<endl;
16    cout<<"16 sanoq sistemasida: "<<satr16<<endl;
17    return 0;
18 }

```

The output window shows the following results:

```

Son kurinishlari
2 sanoq sistemasida : 1111
8 sanoq sistemasida : 17
10 sanoq sistemasida: 15
16 sanoq sistemasida: f

-----
Process exited with return value 0
Press any key to continue . . .

```

4. *Gcvt()* (***char* gcvt(double val, int ndec, char *buf);***) - ushbu funksiya **val** xaqiqiy sonini **ndec** raqamlar sonigacha **buf** satriga o'tkazadi.

The screenshot shows a C++ program that uses `gcvt` to convert a double to a string with a specified number of decimal places. The code is as follows:

```

1 #include <iostream>
2 #include <stdlib.h>
3 using namespace std;
4 int main()
5 {
6     char satr[10]; double son; int raqamlar_soni=4;
7     cout<<"Son ko'rinishidagi raqamlat son: ";
8     cout<<raqamlar_soni<<endl;
9     son=3.154;
10    gcvt(son,raqamlar_soni,satr);
11    cout<<"3.154 sonining satr ko'rinishi: "<<satr<<endl;
12    son=-312.456;
13    gcvt(son,raqamlar_soni,satr);
14    cout<<"-312.456 sonining satr ko'rinishi: "<<satr<<endl;
15    son=0.123E+4;
16    gcvt(son,raqamlar_soni,satr);
17    cout<<"0.123E+4 sonining satr ko'rinishi: ";
18    cout<<satr<<endl;
19    son=12345.456;
20    gcvt(son,raqamlar_soni,satr);
21    cout<<"12345.456 sonining satr ko'rinishi: "<<satr<<endl;
22    return 0;
23 }

```

The output window shows the following results:

```

Son ko'rinishidagi raqamlat son: 4
3.154 sonining satr ko'rinishi: 3.154
-312.456 sonining satr ko'rinishi: -312.5
0.123E+4 sonining satr ko'rinishi: 1230
12345.456 sonining satr ko'rinishi: 1.235e+004

-----
Process exited with return value 0
Press any key to continue . . .

```

String turidagi satrlar.

String turini ishlatishda `<string.h>` kutubxonasidan foydalaniladi.

1. String turini e'lon qilish:

The screenshot shows a C++ IDE with two tabs: 'stringe'lon.cpp' and 'string.cpp'. The code in 'string.cpp' is as follows:

```

1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main() {
5      string s1,s2,s3,s4;
6      char* s="hello world";
7      s1="salom dunyo";
8      s2=s;
9      s3=s1+s2+"!!!";
10     cout<<"s1="<<s1<<endl<<"s2="<<s2<<endl;
11     cout<<"s3="<<s3<<endl<<s1.at(0)<<endl;
12     return 0;
13 }

```

The output window on the right shows the following text:

```

s1=salom dunyo
s2=hello world
s3=salom dunyohello world!!!
s
salom dunyo
-----
Process exited with return value 0
Press any key to continue . . .

```

Ushbu tur bilan ishlaydigan funksiyalar mavjud va ulardan o'zgaruvchi nomi, nuqta va keyin funktsiya nomini yozish orqali foydalaniladi.

2. Satr qismini boshqa satrga nusxalash funktsiyasi:

The screenshot shows a C++ IDE with two tabs: 'stringe'lon.cpp' and 'string.cpp'. The code in 'string.cpp' is as follows:

```

1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main() {
5      string s1,s2,s3;
6      s1="salom dunyo";
7      s2.assign(s1);
8      s3.assign(s1,6,6);
9      cout<<"s2="<<s2<<endl;
10     cout<<"s3="<<s3<<endl;
11     return 0;
12 }

```

The output window on the right shows the following text:

```

s2=salom dunyo
s3=dunyo
-----
Process exited with return value 0
Press any key to continue . . .

```

3. Satr qismini boshqa satrga qo'shish funktsiyasi:

The screenshot shows a C++ IDE with two tabs: 'stringe'lon.cpp' and 'string.cpp'. The code in 'string.cpp' is as follows:

```

1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main() {
5      string s,s1,s2;
6      cin>>s;
7      s1=s;
8      s1.append("salom dunyo",6,5);
9      cout<<s1;
10     return 0;
11 }

```

The output window on the right shows the following text:

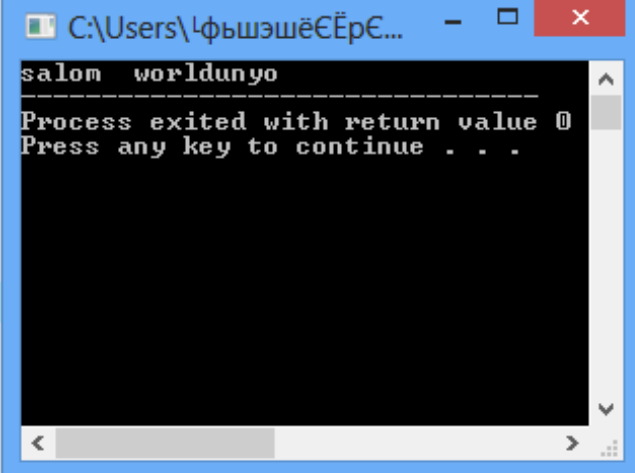
```

hello
hellodunyo
-----
Process exited with return value 0
Press any key to continue . . .

```

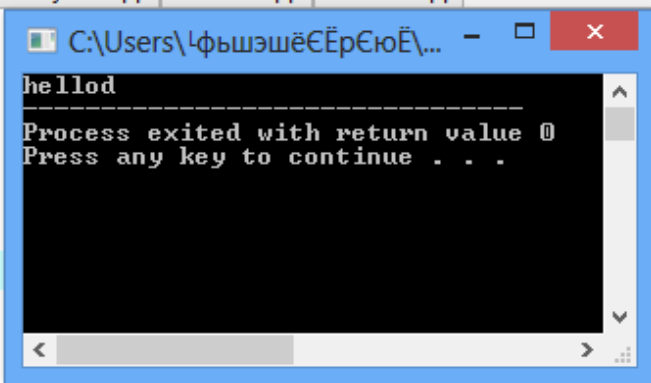
4. Satr qismini boshqa satr ichiga joylashtirish:

```
string'lon.cpp  string.cpp
1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main(){
5      string s,s1,s2;
6      s1="hello world ";
7      s2="salom dunyo";
8      s2.insert(6,s1,5,5);
9      cout<<s2;
10     return 0;
11 }
```



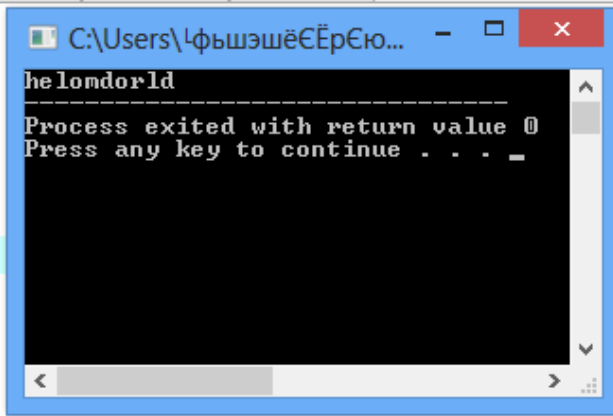
5. Satr qismini o'chirish:

```
string'lon.cpp  string.cpp  new1B.cpp  BloknotRyuko.cpp  Bozorov.cpp  sasasasa.cpp
1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main(){
5      string s1;
6      s1="hello world ";
7      s1.erase(5,5);
8      cout<<s1;
9      return 0;
10 }
```



6. Satr qismini almashtirish funksiyasi:

```
string'lon.cpp  string.cpp  new1B.cpp  BloknotRyuko.cpp  Bozorov.cpp  sasasasa.cpp
1  #include <iostream>
2  #include <string.h>
3  using namespace std;
4  int main(){
5      string s1,s2;
6      s1="helloworld";
7      s2="salomdunyo";
8      s1.replace(3,3,s2,3,3);
9      cout<<s1;
10     return 0;
11 }
```

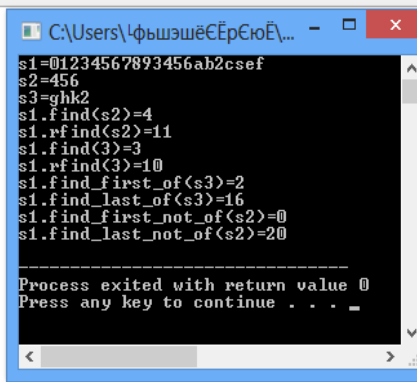


7. Satr qismini izlash:

```

1 #include <iostream>
2 using namespace std;
3 int main() {
4     string s1="01234567893456ab2csef", s2="456", s3="ghk2";
5     cout<<"s1="<<s1<<endl;
6     cout<<"s2="<<s2<<endl;
7     cout<<"s3="<<s3<<endl;
8
9     int i,j;
10    i=s1.find(s2);
11    j=s1.rfind(s2);
12    cout<<"s1.find(s2)="<<i<<endl;
13    cout<<"s1.rfind(s2)="<<j<<endl;
14    cout<<"s1.find(3)="<<s1.find('3') <<endl;
15    cout<<"s1.rfind(3)="<<s1.rfind('3') <<endl;
16    cout<<"s1.find_first_of(s3)="<<s1.find_first_of(s3)<<endl;
17    cout<<"s1.find_last_of(s3)="<<s1.find_last_of(s3)<<endl;
18    cout<<"s1.find_first_not_of(s2)="<<s1.find_first_not_of(s2)<<endl;
19    cout<<"s1.find_last_not_of(s2)="<<s1.find_last_not_of(s2)<<endl;
20    return 0;
21 }

```

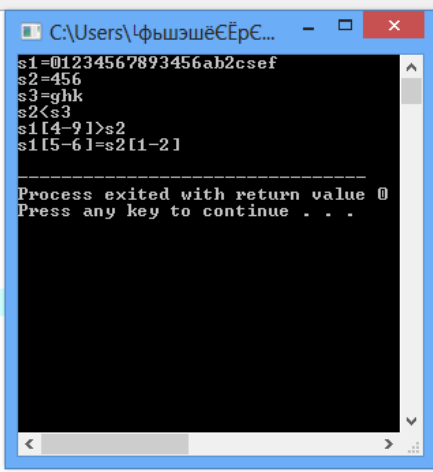


8. Satrlarni solishtirish:

```

1 #include <iostream>
2 #include <string.h>
3 using namespace std;
4 int main()
5 {
6     string s1="01234567893456ab2csef", s2="456",
7         s3="ghk";
8     cout<<"s1="<<s1<<endl;
9     cout<<"s2="<<s2<<endl;
10    cout<<"s3="<<s3<<endl;
11    if(s2.compare(s3)>0) cout<<"s2>s3"<<endl;
12    if(s2.compare(s3)==0) cout<<"s2=s3"<<endl;
13    if(s2.compare(s3)<0) cout<<"s2<s3"<<endl;
14    if(s1.compare(4,6,s2)>0) cout<<"s1[4-9]>s2"<<endl;
15    if(s1.compare(5,2,s2,1,2)==0)
16        cout<<"s1[5-6]=s2[1-2]"<<endl;
17    return 0;
18 }

```



Berilgan masala algoritmi. Ushbu masalada n ta satrda n ta talabani nomi – sharihi, bahosi, telefon nomeri bitta satrda berilgan ma’lumotlari kiritilgan. Masala yechimini asosiy qismi, har bitta talabani nomi – sharihi, bahosi, telefon nomerlarini struktura orqali yangi tur yaratib, unga joylashtirish. Undan keyin talabalarni baholari bo’yicha saralash lozim. Demak, yangi turni struktura yordamida yaratamiz. Keyin satrdagi ma’lumotlarni alohida turlarga *strtok()* funksiyasi yordamida ajratamiz va saralash algoritmilari orqali baho turi bo’yicha saralab chiqamiz. Natijani ekranga uzatamiz.

Asosiy dastur matni.

```
1  #include <iostream>
2  #include <conio.h>
3  #include <string.h>
4  using namespace std;
5  struct talaba{
6      string fish;
7      string baho;
8      string nomer;
9  };
10 int main()
11 {
12     char Ismlar[100]; char* Satrdagi_ism;
13     char Ajratuvchi[]=","; talaba a[100];
14     int n,i,j;
15     cin>>n;
16     cin.getline(Ismlar,sizeof(Ismlar));
17     for (i=1;i<=n;i++)
18     {
19         cin.getline(Ismlar,sizeof(Ismlar));
20         a[i].fish=strtok(Ismlar,Ajratuvchi);
21         a[i].baho=strtok(NULL,Ajratuvchi);
22         a[i].nomer=strtok(NULL,Ajratuvchi);
23     }
24     for (i=1;i<n;i++)
25         for (j=i+1;j<=n;j++)
26         if (a[i].baho<a[j].baho) {
27             swap(a[i],a[j]);
28         }
29     for (i=1;i<=n;i++)
30         cout<<i<<". "<<a[i].fish<<" "<<a[i].baho<<" "<<a[i].nomer<<endl;
31     return 0;
32 }
```

Dastur bajarilish natijalari.

Dastur bajarilishida birinchi navbatda n ta talabani nomi – sharifi, fanlardan olgan natijaviy baholari va telefon nomerlari bir – biridan vergul bilan ajratilgan holda bitta satrda berilgan. Dasturda bajariladigan birinchi ish ushbu talabalarning uchta parametrlari bo'yicha berilgan ma'lumotlarini alohida – alohida ajratib saqlash uchun strukturadan foydalanib talaba strukturasi yaratib olamiz. Ushbu strukturada 3 ta parameter mavjud ya'ni string turida fish, baho va nomer. Har bir satrda berilgan ma'lumotlarni *strtok()* funksiyasi yordamida alohida ajratib talaba massiviga joylashtirib chiqamiz. Keyin talaba massivining baho turi bo'yicha saralashning tanlash usulidan foydalanib tartiblab chiqamiz. Natijada talabalarni baholari bo'yicha saralangan hosil bo'ladi.

Demak, birinchi quyidagicha talaba strukturasi yaratib olamiz:

```
5 struct talaba{
6     string fish;
7     string baho;
8     string nomer;
9 }
```

Keyin quyidagicha kiritilayotgan ma'lumotlarni ajratib talaba massiviga joylashtirib chiqamiz:

```

char Ismlar[100]; char* Satrdagi_ism;
char Ajratuvchi[]=","; talaba a[100];
int n,i,j;
cin>>n;
cin.getline(Ismlar,sizeof(Ismlar));
for (i=1;i<=n;i++)
{
cin.getline(Ismlar,sizeof(Ismlar));
a[i].fish=strtok(Ismlar,Ajratuvchi);
a[i].baho=strtok(NULL,Ajratuvchi);
a[i].nomer=strtok(NULL,Ajratuvchi);
}

```

Va oxirida talaba massivini qo'yilgan baxolari bo'yicha quyidagicha saralab chiqamiz:

```

24 for (i=1;i<=n;i++)
25 for (j=i+1;j<=n;j++)
26 if (a[i].baho<a[j].baho) {
27 swap(a[i],a[j]);
28 }

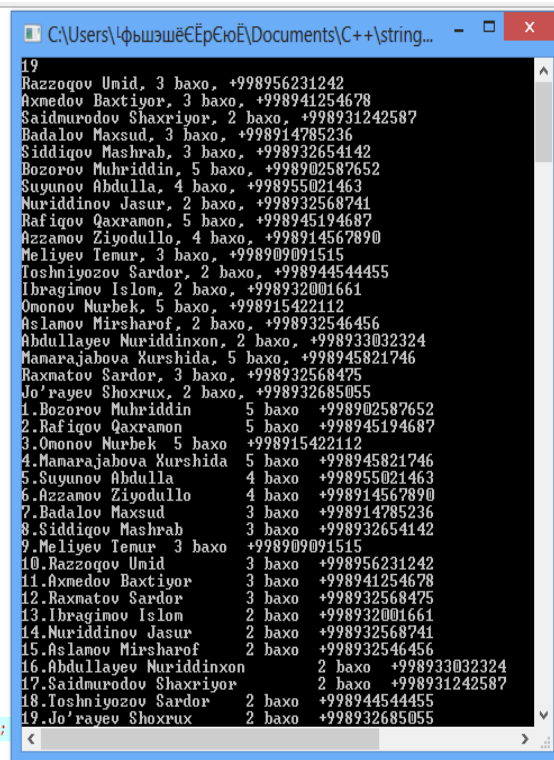
```

Natijalar listingi.

```

1 #include <iostream>
2 #include <conio.h>
3 #include <string.h>
4 using namespace std;
5 struct talaba{
6 string fish;
7 string baho;
8 string nomer;
9 };
10 int main()
11 {
12 char Ismlar[100]; char* Satrdagi_ism;
13 char Ajratuvchi[]=","; talaba a[100];
14 int n,i,j;
15 cin>>n;
16 cin.getline(Ismlar,sizeof(Ismlar));
17 for (i=1;i<=n;i++)
18 {
19 cin.getline(Ismlar,sizeof(Ismlar));
20 a[i].fish=strtok(Ismlar,Ajratuvchi);
21 a[i].baho=strtok(NULL,Ajratuvchi);
22 a[i].nomer=strtok(NULL,Ajratuvchi);
23 }
24 for (i=1;i<=n;i++)
25 for (j=i+1;j<=n;j++)
26 if (a[i].baho<a[j].baho) {
27 swap(a[i],a[j]);
28 }
29 for (i=1;i<=n;i++)
30 cout<<i<<" "<<a[i].fish<<"\t"<<a[i].baho<<"\t"<<a[i].nomer<<endl;
31 return 0;
32 }

```



```

C:\Users\fofawawEepEyoE\Documents\C++\string...
19
Razzoqov Umid, 3 baxo, +998956231242
Axmedov Baxtiyor, 3 baxo, +998941254678
Saidmurodov Shaxriyor, 2 baxo, +998931242587
Badalov Maxsud, 3 baxo, +998914785236
Siddiqov Mashrab, 3 baxo, +998932654142
Bozorov Muhriddin, 5 baxo, +998902587652
Suyunov Abdulla, 4 baxo, +998955021463
Muriddinov Jasur, 2 baxo, +998932568741
Rafiqov Qaxramon, 5 baxo, +998945194687
Azzamov Ziyodullo, 4 baxo, +998914567890
Meliyev Temur, 3 baxo, +998909091515
Toshniyozov Sardor, 2 baxo, +998944544455
Ibragimov Islom, 2 baxo, +998932001661
Ononov Nurbek, 5 baxo, +998915422112
Aslanov Mirsharof, 2 baxo, +998932546456
Abdullayev Nuriddinxon, 2 baxo, +998933032324
Manarajabova Xurshida, 5 baxo, +998945821746
Raxmatov Sardor, 3 baxo, +998932568475
Jo'rayev Shoxrux, 2 baxo, +998932685055
1.Bozorov Muhriddin 5 baxo +998902587652
2.Rafiqov Qaxramon 5 baxo +998945194687
3.Ononov Nurbek 5 baxo +998915422112
4.Manarajabova Xurshida 5 baxo +998945821746
5.Suyunov Abdulla 4 baxo +998955021463
6.Azzamov Ziyodullo 4 baxo +998914567890
7.Badalov Maxsud 3 baxo +998914785236
8.Siddiqov Mashrab 3 baxo +998932654142
9.Meliyev Temur 3 baxo +998909091515
10.Razzoqov Umid 3 baxo +998956231242
11.Axmedov Baxtiyor 3 baxo +998941254678
12.Raxmatov Sardor 3 baxo +998932568475
13.Ibragimov Islom 2 baxo +998932001661
14.Muriddinov Jasur 2 baxo +998932568741
15.Aslanov Mirsharof 2 baxo +998932546456
16.Abdullayev Nuriddinxon 2 baxo +998933032324
17.Saidmurodov Shaxriyor 2 baxo +998931242587
18.Toshniyozov Sardor 2 baxo +998944544455
19.Jo'rayev Shoxrux 2 baxo +998932685055

```

Foydalanilgan adabiyotlar.

1. C++ tilida programmalash asoslari, Sh. F. Madrahimov, S. M. Gaynazarov, Toshkent – 2009.
2. C/C++ tilida masalalar va topshiriqlar, Yusupov R. A., Mirzanova Sh, Iskandarova S. N. Samarqand – 2013.
3. C/C++ и MS Visual C++ 2010 для начинающих, Пахомов Б. И. Санктпетербург – 2011.

Laboratoriya ishi №4

Mavzu: C++ ning grafik imkoniyatlari. Grafik funksiyalar, Grafik animatsiyalar, rangli formalar hosil qilish.

Maqsad: Grafiklar chizish ko'nikmalarini hosil qilish. Grafik funksiyalar bilan ishlash. Animatsiyalar yaratish. Formalar hosil qilish.

I. Masalaning qo'yilishi. Grafik chizuvchi funksiyalarni e'lon qiling. Variantda ko'rsatilgan grafiklar ustida bajariladigan amallarning dasturini tuzing.

II. Variantlar

№	Variantlar
1	Tomoni klaviataturadan kiritiladigan kubning rasmini chizing
2	Tomonlari klaviataturadan kiritiladigan parallelopipedning rasmini chizing
3	Radiusi klaviataturadan kiritiladigan sharning rasmini chizing
4	Tomonlari klaviataturadan kiritiladigan uchburchakka ichki aylana chizing
5	Tomonlari klaviataturadan kiritiladigan uchburchakka tashqi aylana chizing
6	Tomoni klaviataturadan kiritiladigan kubga ichki shar rasmini chizing
7	Tomoni klaviataturadan kiritiladigan kubga tashqi shar rasmini chizing
8	2D formatda avtomobilning rasmini chizing
9	$y = \sin x$ funksiyaning grafigini chizing
10	$y = \cos x$ funksiyaning grafigini chizing
11	$y = a \cdot x^2 + b \cdot x + c$ funksiyaning grafigini chizing
12	Yarim oy rasmini chizing
13	Besh yulduz rasmini chizing
14	Ellipsga ichki kvadrat chizing
15	Ismi sharifningizni tog'ri to'rtburchak ichiga rangli chizing
16	Soat rasmini chizing
17	Monitor rasmini chizing
18	Notebook rasmini chizing
19	2D formatda Uy rasmini chizing
20	2D formatda telefon rasmini chizing

III. Hisobot mazmuni

1. Titul varaqasi: Fan nomi; ish mavzusi; talaba ismi sharifi; bajarish vaqti;
2. Masalaning qo'yilishi;
3. Ishdan maqsad;
4. Grafik funksiyalarning ta'rifi, qo'llanilishi;

5. Berilgan masala uchun algoritm ta'rifi;
6. Asosiy dastur matni;
7. Dastur bajarilish natijalari tahlili;
8. Natijalar listingi;
9. Foydalanilgan adabiyotlar.

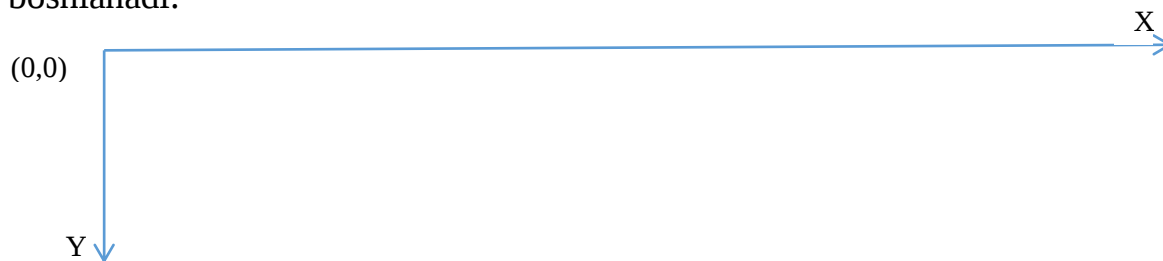
IV. Uslubiy ko'rsatmalar

Masalaning qo'yilishi. $Y = \sin x$ funksiyaning grafigini chizuvchi dastur tuzing.

Ishdan maqsad. $Y = \sin x$ funksiyaning grafigini grafik funksiyalardan foydalanib dasturiy ta'minotni tuzish.

Nazariy qism. Grafik funksiyalarning ta'rifi, e'loni va qo'llanilishi.

Grafik rejimda koordinatalar sistemasi (0,0) nuqtadan quyidagi ko'rinishda boshlanadi:



Graflar bilan ishlovchi ba'zi bir funksiylar.

`void initwindow(int width, int height);` - grafik chizish uchun eni width va balandligi height bo'lgan oyna ochish. Agar ushbu funksiya dasturga kiritilmagan bo'lsa, hatolik sodir bo'ladi.

`void closegraph();` - grafik rejimni yopish.

`void line(int x1, int y1, int x2, int y2);` - $(x1, y1)$ va $(x2, y2)$ nuqtalarni tutashtiruvchi kesma chizadi.

`void rectangle(int x1, int y1, int x2, int y2);` - funksiyasi joriy rang bilan yuqori chap burchagi $(x1, y1)$ va quyi o'ng burchagi $(x2, y2)$ nuqtalarda bo'lgan to'g'ri to'rburchak chizadi.

`void circle(int x, int y, int r);` - funksiyasi markazi (x, y) nuqtada radiusi r ga teng bo'lgan aylana chizadi.

`void arc(int x, int y, int stangle, int endtangle, int radius);` - aylana yoyini chizadi. x, y - aylana markazi koordinatalari, **radius** - aylana radiusi;

stangle, endtangle - yoyning boshlang'ich va oxirgi burchaklari;

`void ellipse(int x, int y, int stangle, int endtangle, int xradius, int yradius);` - ellipse yoyini chizish;

x, y - aylana markazi koordinatalari, **xradius, yradius** - ellipse balandligi va eni; **stangle, endtangle** - yoyning boshlang'ich va oxirgi burchaklari;

void **putpixel**(int x,int y, int color); - funskiyasi (x,y) koordnatasiga color rangli nuqta qo'yadi;

int **getpixel** (int x,int y) – funksiyasi (x,y) koordinatali nuqta rangini aniqlaydi;

int **getcolor**(); - joriy rangni aniqlaydi;

void **setbkcolor**(int color); - fon rangini o'rnatadi;

int **getbkcolor**(); - fon rangini aniqlaydi;

void **cleardevice**(); - grafik eranni tozalab, kursorni (0,0) nuqtaga o'rnatadi;

int **getmaxx**(); - X koordianatasining eng kata qiymatini aniqlaydi;

int **getmaxy**(); - Y koordinatasining eng kata qiymatini aniqlaydi;

void **setcolor**(int color); - chiziq rangini o'rnatadi. Color sifatida rang nomi yoki kodini keltirish mumkin. Quyidagi jadvalda keltirilgan ranglardan foydalanish mumkin.

0	BLACK	Qora
1	BLUE	Ko'k
2	GREEN	Yashil
3	CYAN	Havorang
4	RED	Qizil
5	MAGENTA	Malinarang
6	BROWN	Jigarrang
7	LIGHTGRAY	Och kulrang
8	DARKGRAY	To'q kulrang
9	LIGHTBLUE	Och havorang
10	LIGHTGREEN	Och yashil
11	LIGHTCYAN	Feruzarang
12	LIGHTRED	Och qizil
13	LIGHTMAGENTA	Och malinarang
14	YELLOW	Sariq
15	WHITE	Oq

Chiziq joriy rangi deb, biror shakl chizig'i chizish uchun ishlatiladigan rangga aytiladi. Chiziqning joriy rangi **setcolor** funksiyasi orqali o'rnatiladi. Chiziqning joriy rangi sifatida odatda oq rang o'rnatilgan bo'ladi.

Fonning joriy rangi sifatida odatda qora rang o'rnatilgan bo'ladi. Fon rangini o'zgartirish uchun **setbkcolor** funksiyasidan foydalaniladi.

Joriy holat – bu koordinatalar ko'rsatkichi (x,y) o'rnatilgan nuqtadir. Qoidaga binoan joriy holat biror shaklni chizish bilan o'zgaradi. Ba'zi funksiyalar bajarilgandan keyin joriy holat o'zgarishi mumkin.

Koordinata ko'rsatkichini o'rnatish.

void **moveto**(int x, int y); - koordinata ko'rsatkichini koordinatalar sistemasi boshiga nisbatan (x,y) nuqtaga o'rnatadi. Bu funksiya bajarilganda nuqta ekranni yoritmaydi.

void **moverel**(int dx,int dy) - koordinata ko'rsatkichini joriy holatga nisbatan (dx,dy) ga siljitadi. Bu funksiya bajarilganda nuqta ekranni yoritmaydi.

void **lineto**(int x, int y); - joriy holatdan boshlab (x,y) nuqtaga kesma chizadi. Bu funksiya bajarilgandan keyin joriy holat (x,y) nuqtaga o'zgaradi.

void **linerel**(int dx, int dy); - joriy holatga nisbatan (dx, dy) nuqtagacha kesma chizadi. Funksiya bajarilgandan so'ng joriy holat oldingi holatiga nisbatan (dx, dy) nuqtaga o'zgaradi.

int **getx**(); - joriy holatning X koordinatasini aniqlaydi.

int **gety**(); - joriy holatning Y koordinatasini aniqlaydi.

Bo'yalgan shakllarni chizish.

void **setlinestyle**(int linestyle, unsigned upattern, int thickness);

enum line_styles

{

SOLID_LINE,

DOTTED_LINE,

CENTER_LINE,

DASHED_LINE,

USERBIT_LINE

};

void **setfillstyle**(int pattern, int color); - funksiyasi **color** rangi bilan **style** uslubida shaklni to'ldiradi. **Style** quyidagilarni qabul qilishi mumkin.

0=EMPTY_FILL	Soha fon rangini bo'yash
1=SOLID_FILL	Sohani belgilangan rangda to'ldirish
2=LINE_FILL	Sohani qalin gorizontal chiziq bilan to'ldirish
3=LTSLASH_FILL	Sohani ingichka"/"/" chiziqlar bilan to'ldirish
4=SLASH_FILL	Sohani qalin"/" chiziqlar bilan to'ldirish
5=BKSLASH_FILL	Sohani qalin"//" chiziqlar bilan to'ldirish
6=LTBKSLASH_FILL	Sohani ingichka"/" chiziqlar bilan to'ldirish
7=HATCH_FILL	Sohani to'r bilan to'ldirish
8=XHATCH_FILL	Sohani egri to'r bilan to'ldirish
9=INTERLEAVE_FILL	Sohani zich shtrix bilan to'ldirish
10=WIDE_DOT_FILL	Sohani kam sondagi nuqtalar bilan to'ldirish
11=CLOSE_DOT_FILL	Sohani ko'p sondagi nuqtalar bilan to'ldirish
12=USER_FILL	Sohani dasturchi belgilagan uslubda to'ldirish

void **floodfill**(int x, int y, int border); - chegarasi **border** bo'lgan sohani joriy rangga bo'yaydi. (x,y) bo'yalishi lozim bo'lgan sohadagi ixtiyoriy nuqta.

void **bar** (*int x1, int y1, int x2, int y2*); - o'rnatilgan rang yuqori chap burchagi (x1, y1) va quyi o'ng burchagi (x2,y2) nuqtalar bo'lgan bo'yalgan tog'ri to'rtburchak chizadi.

void **fillellipse**(*int x, int y, int xradius, int yradius*); - bo'yalgan ellipse chizadi.

void **pieslice** (*int x, int y, int stangle, int endangle, int radius*); - aylana sektorini chizadi.

void **sector**(*int x, int y, int stangle, int endangle, int xradius,int yradius*); - ellipse sektorini chizadi.

Grafik rejimdagi matnlar.

void **settextstyle**(*int font, int direction, int charsize*); - kerakli shrifni o'rnatadi.

Font – shriftni aniqlovchi o'zgaruvchi. (0 dan 11 gacha oraliqda qiymat qabul qilishi mumkin). **Direction** – matnni chop qilish yo'nalishi (0 – chapdan o'ngga, 1 – pastdan yuqoriga). **charsize** – matn kattaligini aniqlaydi.

Enum font_names

```
{  
    DEFAULT_FONT,  
    TRIPLEX_FONT,  
    SMALL_FONT,  
    SANS_SERIF_FONT,  
    GOTHIC_FONT,  
    SCRIPT_FONT,  
    SIMPLEX_FONT,  
    TRIPLEX_SCR_FONT,  
    COMPLEX_FONT,  
    EUROPEAN_FONT,  
    BOLD_FONT  
};
```

void **outtextxy**(*int x, int y, char *string*); - x, y nuqtaga matnni chiqarish.

void **outtext**(*char *string*); - joriy nuqtaga matnni chiqarish.

Berilgan masala algoritmi.

$Y = \sin x$ funksiyaning grafigini chizish uchun ikki o'lchovli dekart koordinatalar tekisligi kerak bo'ladi. Dekart koordinatalar tekisligida Absissalar – OX o'qi, ordinatalar – OY o'qi va koordinatalar boshi – O. funksiyaning grafigini chizishda erkli o'zgaruvchi x ga qiymat berib, erksiz o'zgaruvchi Y ning qiymatlarini aniqlab koordinatalar tekisligidagi o'sha (x,y) nuqtalarni tutashritib chiqish natijasida $Y = \sin x$ funksiyaning grafigi hosil qilinadi.

Asosiy dastur matni.

```
sin.cpp |
#include <graphics.h>
#include <conio.h>
#include <math.h>
int main()
{
    initwindow(600, 600);

    line(0, 300, getmaxx(), 300);
    line(400, 0, 400, getmaxy());

    int x, y;
    float pi = 3.1415;

    for (int a = -360; a <= 360; a++)
    {
        x = 400 + a;
        y = 300 - sin(a * pi / 180) * 100;
        putpixel(x, y, WHITE);
        y = 300 - cos(a * pi / 180) * 100;
        putpixel(x, y, YELLOW);

        delay(10);
    }
    getch();
    closegraph();

    return 0;
}
```

Dastur bajarilish natijalari tahlili.

```
line(0, 300, getmaxx(), 300);
line(400, 0, 400, getmaxy());
```

Dasturning ushbu qismi yordamida ikki o'lchivli dekart koordinatalar teksiligi chiziladi.

```

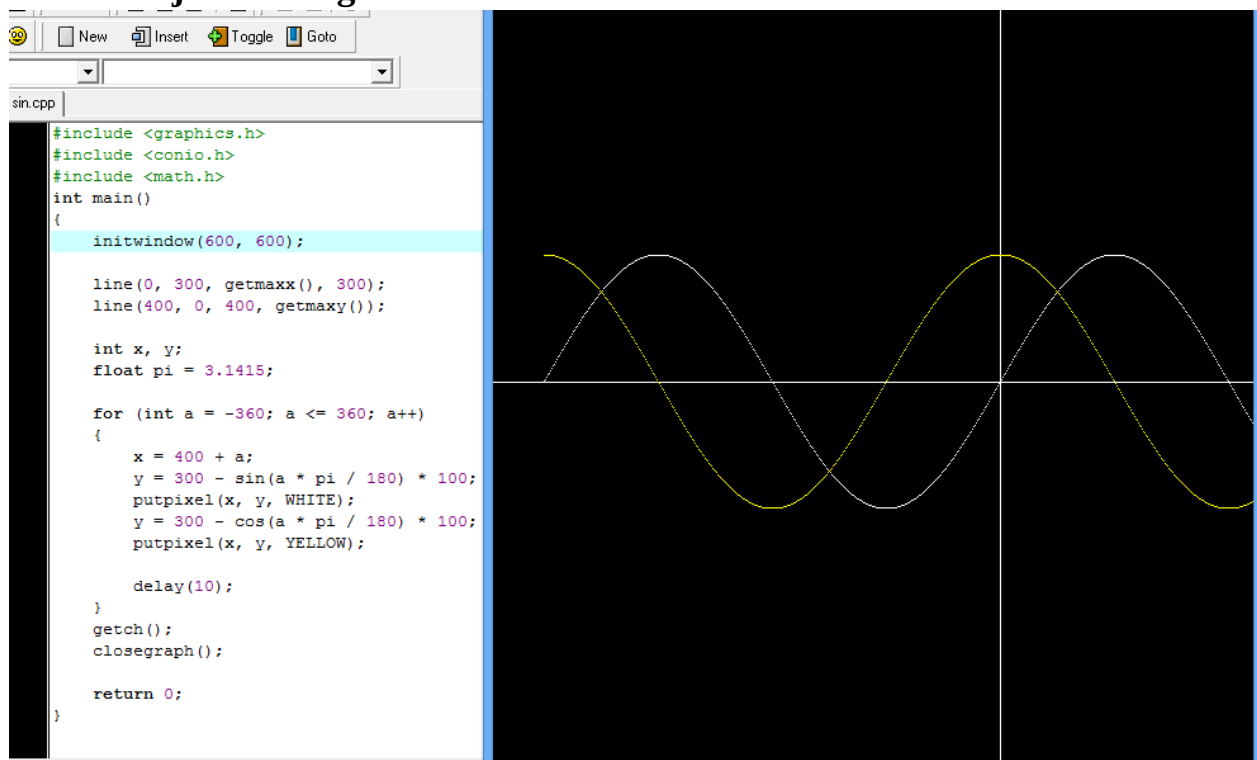
for (int a = -360; a <= 360; a++)
{
    x = 400 + a;
    y = 300 - sin(a * pi / 180) * 100;
    putpixel(x, y, WHITE);
    y = 300 - cos(a * pi / 180) * 100;
    putpixel(x, y, YELLOW);

    delay(10);
}

```

Dasturning ushbu qismi yordamida $Y=\sin x$ va $Y=\cos x$ funksiyalarning grafiklari chiziladi.

Natijalar listingi.



Foydalanilgan adabiyotlar.

1. C++ tilida programmalash asoslari, Sh. F. Madrahimov, S. M. Gaynazarov, Toshkent – 2009.
2. C/C++ tilida masalalar va topshiriqlar, Yusupov R. A., Mirzanova Sh, Iskandarova S. N. Samarqand – 2013.
3. C/C++ и MS Visual C++ 2010 для начинающих, Пахомов Б. И. Санктпетербург – 2011.

Laboratoriya ishi №5

Mavzu: Fayllar bilan ishlash

Maqsad: Fayllar bilan ishlash ko'nikmalarini hosil qilish. Fayldan ma'lumotlar o'qish, yozish, to'g'rilash. Fayllar bilan ishlovchi kutubxona moduli *fstream* bilan tanishish.

I. Masalaning qo'yilishi. O'qish va yozish uchun ikkita fayl yarating. Variantda ko'rsatilgan misollar ustida bajariladigan amallarning dasturini tuzing.

II. Variantlar

№	Variantlar
1	Input.txt faylida berilgan natural n sonining raqamlari yig'indisini output.txt fayliga chiqaring
2	Input.txt faylida berilgan ikkita a va b natural sonlarining eng katta umumiy bo'luvchisini output.txt fayliga chiqaring
3	Input.txt faylida berilgan natural n sonining faktorialini output.txt fayliga chiqaring
4	Input.txt faylida berilgan n ta massiv elementlarini saralab output.txt fayliga chiqaring
5	Input.txt faylida berilgan n ta talaba ismi – sharifini alifbo bo'yicha saralab output.txt fayliga chiqaring
6	Input.txt faylida berilgan natural a va b sonlarining yig'indisini output.txt fayliga chiqaring. $10^{20} \leq a, b \leq 10^{100}$.
7	Input.txt faylida o'nlik sanoq sistemasida berilgan a sonini ikkilik sanoq sistemasiga o'tkazib output.txt fayliga chiqaring
8	Input.txt faylida berilgan $A_{n,m}$ matritsa elementlarini transponirlab output.txt fayliga chiqaring
9	Input.txt faylida n, m, k matritsa o'lchamlari va $A_{n,m}$, $B_{m,k}$ matritsalar berilgan. Matritsalarini ko'paytirib output.txt fayliga chiqaring
10	Input.txt faylida natural n soni berilgan. Dastlabki n ta fibonachi sonlarini output.txt fayliga chiqaring
11	Input.txt faylida satr berilgan. Berilgan satrdagi so'zlar sonini output.txt fayliga chiqaring
12	Input.txt faylida n massiv o'lchami va n ta massiv berilgan. Output.txt fayliga massiv elementlari ichidan dastlabki ikkita eng kichik elementlari yig'indisini output.txt fayliga chiqaring.
13	Input.txt faylida n va m natural sonlari berilgan. $\sum_{i=1}^n \prod_{j=1}^m \frac{i+j}{i*j}$ formulani

	hisoblab output.txt fayliga chiqaring
14	Input.txt faylida nol va birlardan tashkil topgan satr berilgan. Berilgan satrdagi birlar miqdorini output.txt fayliga chiqaring
15	Input.txt faylida nol va birlardan tashkil topgan satr berilgan. Ketma – ket kelgan eng uzun nollar miqdorini output.txt fayliga chiqaring. Misol: 0000000111111111111000000111100000000000000001010011111. Javob:17
16	Input.txt faylida natural n soni berilgan. Berilgan n sonining barcha natural bo'luvchilarini output.txt fayliga chiqaring
17	Input.txt faylida natural n soni berilgan. Berilgan n sonigacha bo'lgan barcha tub sonlarni output.txt fayliga chiqaring
18	Input.txt faylida natural n soni berilgan. Berilgan n sonining raqamlarini teskari tartibda yozishdan hosil bo'lgan sonni output.txt fayliga chiqaring
19	Input.txt faylida berilgan natural a va b sonlarining eng kichik umumiy karralisini output.txt fayliga chiqaring.
20	Input.txt faylida berilgan

III. Hisobot mazmuni

1. Titul varaqasi: Fan nomi; ish mavzusi; talaba ismi sharifi; bajarish vaqti;
2. Masalaning qo'yilishi;
3. Ishdan maqsad;
4. Fayllar bilan ishlovchi funksiyalar ta'rifi , qo'llanilishi;
5. Berilgan masala uchun algoritm ta'rifi;
6. Asosiy dastur matni;
7. Dastur bajarilish natijalari tahlili;
8. Natijalar listingi;
9. Foydalanilgan adabiyotlar.

IV. Uslubiy ko'rsatmalar

Masalaning qo'yilishi. Input.txt faylida berilgan natural n sonining raqamlari yig'indisi bir xonali son bo'lsa Output.txt fayliga chiqaring, aks holda yana raqamarini qo'shing. ($1 \leq n \leq 10^{100}$).

Ishdan maqsad. Berilgan masalani yechishda Uzun arifmetika algoritmlarini o'rganish, foydalanish, fayldan ma'lumotlar o'qish, yozish, qo'shishko'nikmalarini oshirish va ulardan foydalanib dasturiy ta'minot tuzish.

Nazariy qism.

1. *fopen*

FILE* *fopen*(const char* ism, const char* tartib) – parametr tartibda tartib da berilgan harakat uchun ko'rsatilgan ismdagi faylni ochish.

r – faqat o'qish uchun. Fayl faqat o'qish uchun ochiladi.

w – yozish. Fayl yozish uchun ochiladi. Agar *fopen* funksiyaning birinchi parametri sifatida ko'rsatilgan fayl mavjud bo'lsa, u holda yangi ma'lumotlar eskilarining ustiga yoziladi, ya'ni eski fayl aniq o'chiriladi.

a – Qo'shish. Fayl ma'lumotlarini mavjud fayl oxiridan qo'shish uchun ochadi. Agar *fopen* funksiyaning birinchi parametric sifatida ko'rsatilgan fayl mavjud bo'lmasa, u holda u yaratiladi.

Agar fayl matnli kabi ochilayotgan bo'lsa, u holda faylni ochish tartibini aniqlovchi o'zgarmas belgisidan keyin t belgisini qo'shish kerak. Masalan, *rt* satri matnli faylni o'qish uchun ochilishini beradi. Faylni ochish muvoffaqiyatli amalga oshgan holda *fopen* funksiyasi o'qish mumkin bo'lgan yoki yozish mumkin bo'lgan oqimga ko'rsatkichni qaytaradi. Agar qandaydir sabab bilan faylni ochish amali bajarilmay qolsa, *fopen* **NULL** qaytaradi. Ma'lumotlarni olishda hato bo'lgan holatlarda, *ferror* funksiyasiga murojaat qilish kerak.

Sarlavha fayl: <stdio.h>

2. *fprintf*

int *fprintf(FILE* oqim, format, o'zgaruvchi ro'yhati);* - birinchi parameter sifatida ko'rsatilgan oqim bilan bog'langan, faylga formatlangan chiqarishni bajaradi. Oqim bilan bog'langan fayl, yozishga ruxsat beradigan tartibda matnli kabi ochilishi kerak.

Sarlavha fayl: <stdio.h>

3. *fscanf*

int *scanf(FILE* Oqim, const char* format, addressro'yhati);* - birinchi parameter sifatida ko'rsatilgan oqim bilan bog'langan, fayldan o'zgaruvchilarga qiymatlarni formatlashtirilgan o'qishni bajaradi. Oqim bilan bog'langan fayl, o'qishga ruxsat beradigan tartibda matnli kabi ochiladi.

Sarlavha fayl: <stdio.h>

4. *fgets*

char* *fgets(char satr, int belgisoni, FILE* oqim);* - ko'rsatilgan oqimda belgilarni o'qiydi va ularni faylga murojaatda ko'rsatilgan satrda yozadi. O'qish, agar o'qilayotgan belgi *belgisoni* – 1 nomerda bo'lsa yoki navbatdagi belgi yangi qatorga o'tish belgisi hisoblansa to'xtatiladi. Fayldan o'qilgan yangi satr belgisi nol belgisi bilan almashtiriladi. Oqim bilan bog'langan fayl, o'qishga ruxsat beradigan tartibda matnli kabi ochilishi kerak.

Sarlavha fayl: <stdio.h>

5. *fputs*

char* *fputs(char* satr, FILE* Oqim);* - ko'rsatilgan oqimga belgilar satri yoziladi. Satr oxiri belgisi, nol – belgi oqimga yozilmaydi. Oqim bilan bog'langan fayl, yozishga ruxsat beradigan, tartibda matnli kabi ochilishi kerak.

Sarlavha fayl: <stdio.h>

6. *ferror*

int *ferror(FILE* Potok);* - Agar ko'rsatilgan oqim bilan oxirgi amal xatolik bilan tugasa bo'lmagan qiymatni qaytaradi.

Sarlavha fayl: <stdio.h>

7. *feof*

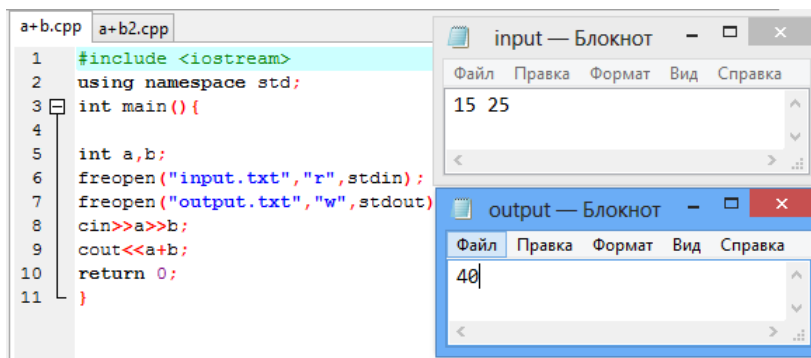
int feof(FILE* Potok); - Agar oqimdan oxirgi o'qish amalini bajarish natijasida fayl oxiriga kelsa, noldan farqli qiymat qaytadi.
Sarlavha fayl: <stdio.h>

8. fclose

int fclose(FILE* Potok) – ko'rsatilgan oqimni yopadi.
Sarlavha fayl: <stdio.h>

Berilgan masala algoritmi.

1 – usul:



```
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     int a,b;
6     freopen("input.txt", "r", stdin);
7     freopen("output.txt", "w", stdout);
8     cin>>a>>b;
9     cout<<a+b;
10    return 0;
11 }
```

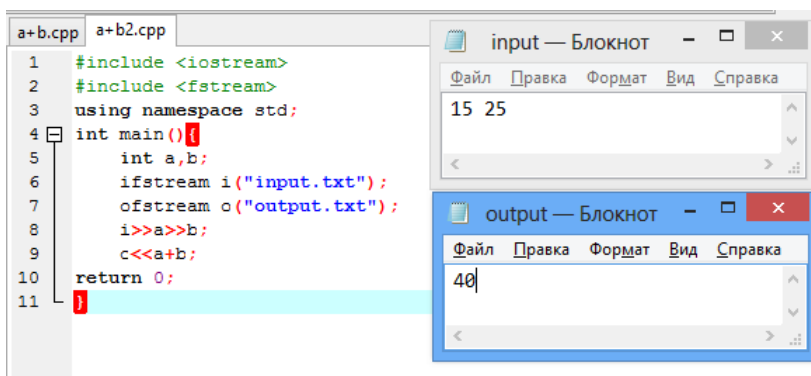
input — Блокнот

15 25

output — Блокнот

40

2 – usul:



```
1 #include <iostream>
2 #include <fstream>
3 using namespace std;
4 int main() {
5     int a,b;
6     ifstream i("input.txt");
7     ofstream o("output.txt");
8     i>>a>>b;
9     o<<a+b;
10    return 0;
11 }
```

input — Блокнот

15 25

output — Блокнот

40

Asosiy dastur matni.

a+b.cpp	a+b2.cpp
1	<code>#include <iostream></code>
2	<code>#include <fstream></code>
3	<code>#include <string.h></code>
4	<code>using namespace std;</code>
5	<code>int main() {</code>
6	<code> string s;</code>
7	
8	<code> ifstream i("input.txt");</code>
9	<code> ofstream o("output.txt");</code>
10	<code> i>>s;</code>
11	<code> int j,k,l,m,n,p=0;</code>
12	<code> n=s.length();</code>
13	<code> for (j=0;j<n;j++)</code>
14	<code> p+=((int)s[j] - 48);</code>
15	<code> while (p>9) {</code>
16	<code> n=p;</code>
17	<code> p=0;</code>
18	<code> while (n>0) {</code>
19	<code> p+=n%10;</code>
20	<code> n/=10;</code>
21	<code> }</code>
22	<code> }</code>
23	<code> c<<p;</code>
24	<code> return 0;</code>
25	<code>}</code>

Dastur bajarilish natijalari tahlili.

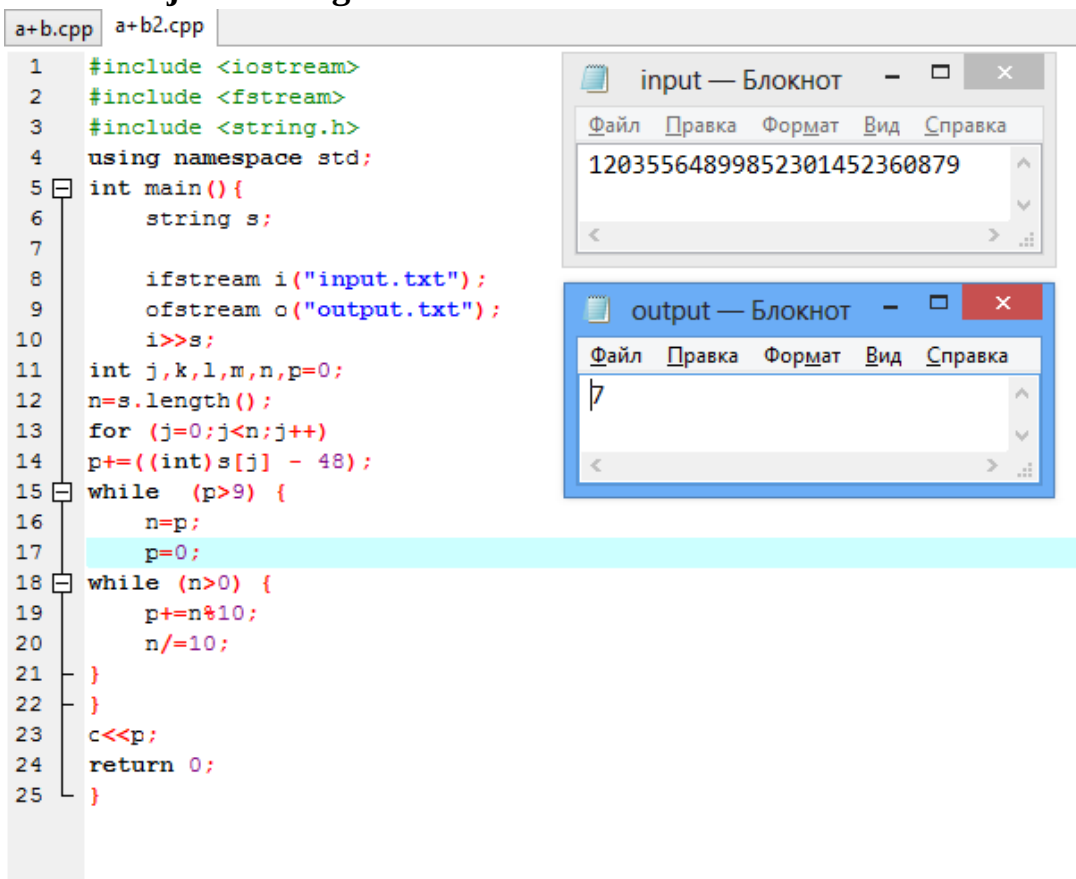
13	<code>for (j=0;j<n;j++)</code>
14	<code>p+=((int)s[j] - 48);</code>

dasturning ushbu qismida input.txt faylida yozilgan n soni butun turga sig'masligi mumkinligini inobatga olib, satrga qabul qilingan sonning raqamlarini qo'shadi.

15	<code>while (p>9) {</code>
16	<code> n=p;</code>
17	<code> p=0;</code>
18	<code>while (n>0) {</code>
19	<code> p+=n%10;</code>
20	<code> n/=10;</code>
21	<code>}</code>
22	<code>}</code>

dasturning ushbu qismida bir marotaba raqamlari yig'ilgan sonni tekshirib ko'radi va u ikki yoki undan ko'p xonali son bo'lsa yana qaytadan raqamlari qo'shiladi.

Natijalar listingi.



The image shows a C++ source code file named `a+b.cpp` and two windows showing the program's execution. The code reads a long number from `input.txt`, processes it by repeatedly dividing by 10, and prints the result to `output.txt`. The `input.txt` window shows the number `12035564899852301452360879`, and the `output.txt` window shows the result `7`.

```
1 #include <iostream>
2 #include <fstream>
3 #include <string.h>
4 using namespace std;
5 int main(){
6     string s;
7
8     ifstream i("input.txt");
9     ofstream o("output.txt");
10    i>>s;
11    int j,k,l,m,n,p=0;
12    n=s.length();
13    for (j=0;j<n;j++)
14        p+=((int)s[j] - 48);
15    while (p>9) {
16        n=p;
17        p=0;
18    }
19    while (n>0) {
20        p+=n%10;
21        n/=10;
22    }
23    c<<p;
24    return 0;
25 }
```

Foydalanilgan adabiyotlar.

1. C++ tilida programmalash asoslari, Sh. F. Madrahimov, S. M. Gaynazarov, Toshkent – 2009.
2. C/C++ tilida masalalar va topshiriqlar, Yusupov R. A., Mirzanova Sh, Iskandarova S. N. Samarqand – 2013.
3. C/C++ и MS Visual C++ 2010 для начинающих, Пахомов Б. И. Санктпетербург – 2011.