**2-topshiriq**

Iteratorlar va ularning qo‘llanilishi. Xotirani taqsimlovchilar va ularga

qo‘yilgan talablar

1-misol:

Standart algoritmlar va iteratorlardan foydalanib,

amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida Standart algoritmlar va iteratorlar

sinflari va uning qayta yuklangan turlaridan foydalanish ko‘nikmalarini egallsh.

Masalaning qo‘yilishi:

Standart algoritmlar va iteratorlar asosida yaratilgan to‘plam elementlarini

copy(from, from\_end, to),

copy\_if(from, from\_end, to, pred),

copy\_n(from, count, to),

copy\_backward(from, from\_end, to\_end),

move(from, from\_end, to),

move\_backward(from, from\_end, to\_end),

swap\_ranges(a, a\_end, b),

fill(to, to\_end, value), fill\_n(to, n, value),

generate(to, to\_end, gen),

generate(v.begin(), v.end(), rand),

generate\_n(to, n, gen), reverse(begin, end),

reverse\_copy(from, from\_end, to),

frnd(begin, end, value),

mismatch(a, a\_end, b, b\_end, eq),

max\_element(begin, end, comp),

min\_element(begin, end, comp),

for\_each(begin, end, fun),

replace(begin, end, old\_val,

new\_val), remove(begin, end, value),

remove\_if(begin, end, pred),

sort(begin, end, comp),

iteratorlar yordamida qayta ishlash.

Masala:

Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtirib yoshi bo‘yicha saralovchi dastru tuzing.

Masalani yechish g‘oyasi:

Talaba haqida har xil turdagi ma’lumotlarni boshqarish uchun “Talaba”

nomli struktura yaratiladi. istream\_iterator iteratoridan foydalanib, matnli fayldan

ma’lumotlar o‘zlashtiriladi va ostream\_iterator iteratoridan foydalanib,

ma’lumotlarni matnli faylga yoki natijalar oynasiga chiqarish mumkin. Yosh

bo‘yicha saralash uchun sort(), ma’lumotlarni vector ga joylashtirish uchun copy(),

ixtiyoriy yoshdan ixtiyoriy yoshgacha ma’lumotlarni chiqarish uchun copy\_if()

algoritmlaridan foydalaniladi. “Talaba” sun‘iy tur bo‘lganligi uchun kiritish va

chiqarish oqimlari(>>, <<) qayta yuklanadi.

Dastur matni:

1 . #include " s td a fx .h "

2 . #include <algorithm>

3. #include <vector>

4 . #include < se t>

5. #include <queue>

6 . # include<iostream>

7 . # include< s tr in g >

8 . # include< s td lib .h >

9 . # include<time.h>

1 0 . # in c lu d e <fstream>

11. using namespace System;

1 2 . using namespace s td ;

13. s t ru c t Talaba{ / / ***TaLaba nomli struktura***

14. s t d : : s t r in g fir s tn am e , secondname;

15. s iz e \_ t age;

16. } ;

17. / / ***"<" amallarni qayta yuklash***

18. bool comparator(const Talaba& p1, const Talaba& p2) {

19. return p1.age < p2.ag e ; }

2 0 . s t d : : ostream& operator << ( s t d : : ostream& out, const

Talaba& p) {

2 1 . out << p.firs tn am e << " " << p.secondname << " " <<

p.ag e ;

2 2 . return ou t;

23. }

24. / / ***">>" amallarni qayta yuklash***

25. s t d : : istream& o p e r a to r > > ( s td ::istream& in , Talaba& p) {

26. in >> p.firs tn am e >> p.secondname >> p.ag e ;

27. return in ; }

28. / / ***standart algoritmlarda parameter uchun qocLLashga***

29. s t ru c t p red ic a te {

30. s iz e \_ t begin, end;

31. p r e d ic a te ( in t p1, in t p2) : b eg in (p1) , end(p2) { }

32. bool operator ( ) ( const Talaba &p) {

33. return ( p.age > begin) && ( p.age < end); }

34. } ;

35. / / ***Matnli faylga bocg clanish***

36. void F ile \_ o u t\_ in ( ) {

37. s t d : : ifs tre am f i n ( " in p u t .t x t " ) ;

38. s t d : : ofstream f o u t ( "o u tp u t .tx t " ) ;

39. i f ( f in = =NULL){c o u t< < " fa y l not found"; re tu rn ; }

4 0 . s t d : :v e c to r<Talaba> v; / / ***Talaba tipida v konteyner***

4 1 . / / ***copy aLgoritmidan foydalanib, fayLdan ma'Lumot***

***o cziashtirish***

4 2 . s td ::c o p y ( s td ::is t r e am \_ ite r a to r <Talaba> ( f in ) ,

s t d : : is t r e am \_ ite r a to r <Talaba> ( ) , s td : : in s e r t e r ( v ,

v .e n d ( ) ) ) ;

4 3 . / / ***sart yoqdamida saralash***

4 4 . s td ::s o r t ( v .b e g in ( ) , v .e n d () , comparator);

4 5 . / / ***copy algoritmidan foydalanib, faylga ma'Lumotlarni***

***yozish***

4 6 . s td ::c o p y \_ i f (v .b e g in ( ) , v .e n d () ,

s t d : : o s tr e am \_ ite ra to r<Talaba> ( s td ::c o u t , "\n") ,

p r ed ic a te (2 0 , 2 5 ) ) ;

4 7 . s td ::c o p y (v .b e g in ( ) , v .e n d () ,

s t d : : o s tr e am \_ ite ra to r<Talaba> ( fo u t , "\n") ) ;

48 . / / s td ::c o p y (v .b e g in ( ) , v .e n d ( ) ,

s td ::o s tre am \_ ite ra to r< T a la b a > (co u t, "\ n " ) ) ;

49 . }

50. in t main(){

51. F ile \_ o u t\_ in ( ) ; g e t c h a r ( ) ; }

Dastur natijasi:

Natijalar oynasi

Salimov Nuriddin 21

Sharipov Komil 21

Mallayev Oybek 22

Holiqov Abdulla 23

Abdullayev Jalil 24

Input.txt fayli fayl

Mallayev Oybek 22

Salimov Nuriddin 21

Holiqov Abdulla 23

Abdullayev Jalil 24

Sharipov Komil 21

Yuqorida keltirilgan algoritmalarning baza birlariga misollar keltiramiz:

2.4.2. reverse\_copy algoritmiga misol:

1 . #include " s td a fx .h "

2 . / / reverse\_copy ga m iso lla r

3. #include <iostream> / / s td ::c o u t

4 . #include <algorithm> / / s td ::rev e r s e \_ co p y

5. #include <vector> / / s td ::v e c to r

6 . using namespace s td ;

7 . in t main ( ) {

8 . in t Int\_m a ssiv [] = { 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 } ;

9 . v e c to r< i n t > myvector;

10. m y v e c to r .r e s iz e (9 ); / / jo y a jr a t i s h

11. reverse\_copy (In t\_m a ssiv , Int\_massiv+9,

m y v e c to r.b eg in ());

1 2 . / / ta rk ib n i chop q i l i s h :

13. cout << "myvector t a r k i b i : " ;

14. fo r (v e c to r< i n t > : : i t e r a t o r it=m y v e c to r .b e g in ();

it!=m y v e c to r .e n d (); + + i t )

15. cout << f f << \* i t ;

16. cout << f\nf ;

17. g e t c h a r ( ) ;

18. return 0 ;

19. }

Dastur natijasi: myvector tarkibi: 9 8 7 6 5 4 3 2 1

2.4.3. frnd(begin, end, value) algoritmiga misol:

1 . #include " s td a fx .h "

2 . / / find misol

3. #include <iostream> / / s td ::c o u t

4 . #include <algorithm> / / s td ::f in d

5. #include <vector> / / s td ::v e c to r

6 . using namespace s td ;

7 . in t main ( ) {

8 . in t Int\_m a ssiv [] = { 10, 20, 30, 40 } ;

9 . in t \* p;

10. p = find (In t\_m a ssiv , Int\_massiv+4, 3 0 ) ;

11. i f (p != Int\_massiv+4)

12. cout<<"Int\_massiv da element to p ilg a n : " <<\*p<<f\nf ;

13. e ls e

14. cout<<"Int\_massiv da element topilmadi \n";

15. v e c to r<i n t > myvector (In t\_m a s s iv ,In t\_m a s s iv + 4 );

16. v e c to r<i n t > : : i t e r a t o r i t ;

17. i t = find (m y v e c to r.b eg in (), myvector.end(), 3 0 ) ;

18. i f ( i t != myvector.end())

19. cout<<"myvector da element to p i ld i : " << \* i t < < f\nf ;

2 0 . e ls e

2 1 . cout << "myvector da element topilmadi \n";

2 2 . g e t c h a r ( ) ; return 0 ;

23. }

Dastur natijasi:

Int\_massiv da element topilgan: 30

myvector da element topildi: 30

2.4.4. mismatch(a, a\_end, b, b\_end, eq) algoritmiga misol:

1 . #include " s td a fx .h "

2 . #include <iostream> / / s td ::c o u t

3. #include <algorithm> / / std::mismatch

4 . #include <vector> / / s td ::v e c to r

5. #include < u t i l i ty > / / s td ::p a i r

6 . bool mypredicate ( in t i , in t j ) {

7 . return ( i = = j ) ; }

8 . using namespace s td ;

9 . in t main ( ) {

1 0 . v e c to r< i n t > myvector;

1 1 . fo r ( in t i = 1 ; i < 6 ; i+ + ) myvector.push\_back ( i \* 1 0 ) ;

12. / / myvector: 10 20 30 40 50

13. in t Int\_m a ssiv [] = { 1 0 ,2 0 ,8 0 ,3 2 0 ,1 0 2 4 } ;

14. / / Int\_massiv: 10 20 80 320 1024

15. p a ir< v e c to r< i n t > : : i t e r a t o r , in t \*> mypair;

16. / / stand a rt taqqoslash yordamida:

17. mypair = mismatch (m y v e c to r.b eg in (), myvector.end(),

In t\_m a ss iv );

18. cout << "B ir in c h i mos kelmaydigan e lem en tla r: " <<

\*m y p a ir .f ir s t ;

19. cout << " va " << \*mypair.second << f\nf ;

2 0 . + +m y p a ir .fir s t ; ++mypair.second;

2 1 . / / predikt taqqoslash yordamida:

2 2 . mypair = std::mismatch (m y p a ir .f ir s t , myvector.end(),

mypair.second, mypredicate);

23. cout << "Ik k in ch i mos kelmaydigan e lem en tla r: " <<

\*m y p a ir .f ir s t ;

24. cout << " va " << \*mypair.second << f\nf ;

25. g e t c h a r ( ) ; return 0 ;

26.}

Dastur natijasi:

Birinchi mos kelmaydigan elementlar: 30 va 80

Ikkinchi mos kelmaydigan elementlar: 40 va 320

1 . #include " s td a fx .h "

2 . #include <iostream> / / s td ::c o u t

3. #include <algorithm> / / std::min\_element,

std::max\_element

4 . bool myfn(in t i , in t j ) { return i <j ; }

5. s t ru c t myclass {

6 . bool o p e ra to r () ( in t i , in t j ) { return i <j ; }

7 . } myobj;

8 . using namespace s td ;

9 . in t main ( ) {

10. in t Int\_m a ssiv [] = { 3 , 7 , 2 , 5 , 6 , 4 , 9 } ;

1 1 . / / stand a rt taqqoslash yordamida:

12. cout << "Eng k ich ik element - " <<

\*min\_element(Int\_massiv,Int\_massiv+7) << f\nf ;

13. cout << "Eng k a tta element - " <<

\*max\_element(Int\_massiv,Int\_massiv+7) << f\nf ;

14. / / myfn funktsiy asidan komp s i f a t id a fo yd a lanish:

15. cout << "Eng k ich ik element " <<

\*min\_element(Int\_massiv,Int\_massiv+7,myfn) << f\nf ;

16. cout << "Eng k a tta element " <<

\*max\_element(Int\_massiv,Int\_massiv+7,myfn) << f\nf ;

17. / / obyekt s i f a t id a myobj-dan komp s i f a t id a

fo yd a lanish:

18. cout << "Eng k ich ik element " <<

\*min\_element(Int\_massiv,Int\_massiv+7,myobj) << f\nf ;

19. cout << "Eng k a tta element " <<

\*max\_element(Int\_massiv,Int\_massiv+7,myobj) << f\nf ;

2 0 . g e t c h a r ( ) ; return 0 ;

2 1 . }

Dastur natijasi:

Eng kichik element - 2

Eng katta element - 9

Eng kichik element 2

Eng katta element 9

Eng kichik element 2

Eng katta element 9

1 . #include " s td a fx .h "

2 . #include <iostream>

3. #include <algorithm>

4 . #include <vector>

5. void myfunction ( in t i )

6 . s td ::c o u t << f f << i ;

7 . }

8 . s t ru c t myclass {

/ / s td ::c o u t

/ / s td ::fo r \_ e a c h

/ / s td ::v e c to r

{ / / fu n c tio n :

/ / fun ction o b je c t type:

9 . void o p e ra to r () ( in t i ) { s td ::c o u t << f f << i ; }

1 0 . } myobject;

1 1 . using namespace s td ;

1 2 . in t main ( ) {

13. v e c to r<i n t > myvector;

14. myvector.push\_back(10);

15. myvector.push\_back(20);

16. myvector.push\_back(30);

17. cout << "myvector t a r k i b i : " ;

18. for\_each (m y v e c to r.b eg in (), myvector.end(),

myfunction);

19. cout << f\nf ;

2 0 . cout << "myvector c o n ta in s :" ;

2 1 . for\_each (m y v e c to r.b eg in (), myvector.end(), myobject);

2 2 . cout << f\nf ;

23. g e t c h a r ( ) ; return 0 ;

Dastur natijasi:

myvector tarkibi: 10 20 30

myvector contains: 10 20 30

2.4.7. replace(begin, end, old\_val, new\_val) algoritmiga misol:

1 . #include " s td a fx .h "

2 . #include <iostream> / / s td ::c o u t

3. #include <algorithm> / / s td ::r e p la c e

4 . #include <vector> / / s td ::v e c to r

5. using namespace s td ;

6 . in t main ( ) {

7 . in t Int\_m a ssiv [] = { 10, 20, 30, 30, 20, 10, 10, 20 } ;

8 . v e c to r<i n t > myvector (In t\_m a ssiv , Int\_massiv+8 ) ;

24.}

9 . / / 10 20 30 30 20 10 10 20

10. rep la ce (m y v e c to r.b eg in (), myvector.end(), 20, 9 9 ) ;

11. / / 10 99 30 30 99 10 10 99

1 2 . cout << "myvector t a r k i b i : " ;

13. fo r (v e c to r<i n t > : : i t e r a t o r it=m y v e c to r .b e g in ();

it!=m y v e c to r .e n d (); + + i t )

14. cout << f f << \* i t ;

15. cout << f\nf ;

16. g e t c h a r ( ) ;

17. return 0 ;

18.}

Dastur natijasi: myvector tarkibi: 10 99 30 30 99 10 10 99

2.4.8. remove va remove\_if(begin, end, pred) algorimiga misol:

1 . #include " s td a fx .h "

2 . #include <iostream> / / s td ::c o u t

3. #include <algorithm> / / std::remove va remove\_if

4 . bool IsOdd ( in t i ) { return ( ( i %2)==1); }

5. using namespace s td ;

6 . in t main ( ) {

7 . in t I n t \_ m a s s iv [ ] = { 1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 } ;

8 . in t \* pbegin = Int\_massiv; / / A

9 . in t \* pend = Int\_massiv+s iz e o f (In t\_m a s s iv )/s iz e o f ( i n t ) ;

10. pend = remove\_if (pbegin, pend, IsOdd);

11. cout << " remove\_if dan keyin Range t r a k i b i : " ;

1 2 . fo r ( in t \* p=pbegin; p!=pend; ++p)

13. cout << f f << \*p; cout << f\nf ;

14. in t Int\_massiv\_2[] = { 3 , 2 , 3 , 4 , 3 , 6 , 7 , 3 , 9 } ;

15. pbegin = Int\_massiv\_2;

16. in t \* pend\_2 =

Int\_massiv\_2+s iz e o f (In t\_m a s s iv \_ 2 )/s iz e o f ( i n t ) ;

17. pend\_2 = remove (pbegin, pend\_2, 3 ) ;

18. cout << " remove dan keyin Range t r a k i b i : " ;

19. fo r ( in t \* p=pbegin; p!=pend\_2; ++p)

2 0 . cout << f f << \*p; cout << f\nf ; }

Dastur natijasi:

remove\_if dan keyin Range trakibi: 2 4 6 8

remove dan keyin Range trakibi: 2 4 6 7 9

1 . #include <iostream> / / s td ::c o u t

2 . #include <algorithm> / / s td : :s o r t

3. #include <vector> / / s td ::v e c to r

4 . bool myfunction ( in t i , in t j ) { return ( i <j ) ; }

5. s t ru c t myclass {

6 . bool o p e ra to r () ( in t i , in t j ) { return ( i <j ) ; } }

7 . myobject;

8 . using namespace s td ;

9 . in t main ( ) {

10. in t Int\_m a ssiv [] = { 3 2 ,7 1 ,1 2 ,4 5 ,2 6 ,8 0 ,5 3 ,3 3 } ;

11. v e c to r<i n t > myvector (In t\_m a ssiv , Int\_massiv+8 ) ;

12. / / 32 71 12 45 26 80 53 33

/ / stand art taqqoslash (operator <) yordamida:

13. so r t (m y v e c to r.b eg in (), m y v e c to r.b eg in ()+ 4 );

14. / / ( 1 2 32 45 71)26 80 53 33

15. / / funksiyadan foydalanib

16. so r t (m y v e c to r.b eg in ()+ 4 , myvector.end(), myfunction);

17. / / 12 32 45 71(26 33 53 80)

18. / / objectdan fo yd a lib

19. so r t (m y v e c to r.b eg in (), myvector.end(), myobject);

20. / / ( 1 2 26 32 33 45 53 71 80)

2 1 . / / e lem en tla rn i chop q i l i s h :

2 2 . cout << "myvector t a r k i t i b i : " ;

23. fo r (v e c to r<i n t > : : i t e r a t o r it=m y v e c to r .b e g in ();

it!=m y v e c to r .e n d (); + + i t )

24. cout << f f << \* i t ; cout << f\nf ; g e t c h a r ( ) ;}

Dastur natijasi:

myvector tarkitibi: 12 26 32 33 45 53 71 80

Laboratoriya mashg‘ulotini bajarish bo‘yicha variantlar.

Ushbu jadvaldagi vazifalar standart algoritmlardan foydalanilgan holda bajariladi.

**LABORATORIYA ISHI**

№ Masalani berilishi

1. Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtirib, ismi bo‘yicha saralovchi dastur

tuzing

2. Matnli faylda N ta stolning nomi, narxi va hajmi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtirib, narxi bo‘yicha saralovchi dastur

tuzing.

3. Matnli faylda N ta talabaning familiyasi, ismi va bahosi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtirib, 5, 4 va 3 bahoga o‘qiydigan talabalar

soni aniqlaydigan dastur tuzing.

4. Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtmb, yoshi eng katta va eng kichkina

talabani aniqlaydigan dastur tuzing.

5. Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtmb, Familiyasi A dan boshlanadigan

talabalarni aniqlaydigan dastur tuzing.

6. Matnli faylda N ta talabaning familiyasi, ismi va yoshi yozilgan. Ushbu

ma’lumotlarni to‘plamga o‘zlashtirib, Ismi B dan boshlanadigan talabalarni

va Familiyasi C dan boshlanadiganlarni aniqlaydigan dastur tuzing.

7. Matnli faylda N ta talabaning familiyasi, ismi va kelgan viloyati yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, talabalari eng kam kelgan

viloyatni aniqlaydigan dastur tuzing.

8. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, ota onasi yo‘qlarini

aniqlaydigan dastur tuzing.

9. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, oilada yo‘lg‘iz o‘g‘il

bo‘lganlarini aniqlaydigan dastur tuzing.

10. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, turmushga chiqqanlarni

aniqlaydigan dastur tuzing.

11. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, oilalali-Uylanganlarini

aniqlaydigan dastur tuzing.

12. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, TTJ dan yashaydiganlarini

aniqlaydigan dastur tuzing.

13. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, kvartirada yashaydiganlarini

aniqlaydigan dastur tuzing.

14. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, Dasturlash fanidan Olinpiada

chempini bo‘lganlarini aniqlaydigan dastur tuzing.

15. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, Stependiant talabalarni

aniqlaydigan dastur tuzing.

16. Matnli faylda N ta talabaning familiyasi, ismi va oilaviy ahvoli yozilgan.

Ushbu ma’lumotlarni to‘plamga o‘zlashtirib, imkoniyati cheklanganlarni

aniqlaydigan dastur tuzing.

2 misol:

**LABORATORIYA ISHI**

**2. STL kutubxonalari. Konteyner sinflar**

**2.1. Chiziqli konteynerlar (array, vector, deque, list, forward\_list)**

**2.1.1. array sinfidan foydalanib, amaliy dasturlar yaratish**

Ishning maqsadi: C++ dasturlash tilida array sinflaridan foydalanib,

konteynernerlar dan foydalanish ko‘nikmalarini shakllantirish.

Masalaning qo‘yilishi:

array asosida yaratilgan to‘plamni kamayish va ko‘payish tartibida saralash.

Masala: ***Talaba*** nomli class yaratish. Konteynerlarning array sinfi yordamida

to‘plamlar yartish. Array ning maxsus metodidan foydalanib, talabalar to‘plamini

talabalarning yoshlari bo‘yicha saralovchi dastur tuzish.

Masalani yechish g‘oyasi: talaba nomli yangi nolar fazosi yartiladi. Uning

tarkibiga Talaba classi va Ism, Fam, Yosh, Step nomli to‘plamlar joylashtiriladi.

Talaba classining ruxsat berilmagan maydonlaridan foydalanish uchun friend

funksiya yaratiladi.

Dastur matni:

1 . #include " s td a fx .h "

2 . #include < s tr in g >

3. #include < ite r a to r >

4 . #include <iostream>

5. #include <algorithm>

6 . #include <array>

7 . using namespace s td ;

8 . const in t n=3;

9 . array< s t r in g , n> Ism;

10. a rray < s t r in g , n> Fam;

11. a rray <i n t , n> Yosh;

12. a rray <f l o a t , n> Step;

13. namespace ta la b a \_ 2 {

14. c la s s Talaba{

15. in t yosh;

16. s t d : : s t r in g ism;

17. s t d : : s t r in g fam;

18. f lo a t s tep ;

19. p u b lic : ***//SaraLash uchun docst funksiyadan foydaLanish***

20. frien d void Saralash\_show(a rray <Talaba, n> &T);

**21. void set(int Y , std::string I, std::string F, float S ){**

2 2 . s t a t i c in t i = 0 ;

23. Yosh[i] = Y; Ism [i] = I ; Fam[i] = F; S t e p [ i] = S;

24. yosh = Y;

25. ism = I ;

26. fam = F;

27. step = S; i+ + ;

28. }

29. void g e t ( ) {

30. cout<<"Ism i: " <<ism<<endl;

31. cout<<"F am iliy a s i: " <<fam<<endl;

32. cout<<"Yoshi: " <<yosh<<endl;

33. cout<<"S tep en d iy a s i: " < < s tep < < en d l;}

34. } ; / /

35. void show(a rray <Talaba, n> &T) {

36. fo r ( in t i = 0 ; i < n; i+ + )

37. { cou t< < i+ 1 < < " - ta la b a haqida haqida"<<endl;

38. T[ i ] . g e t ( ) ; }

39. } / / ***Konteynerga qiymat o czLashtirish***

4 0 . void SetArray(a rray <Talaba, n> &T) {

4 1 . s tr in g I ,F ; in t Y; f lo a t S;

4 2 . fo r ( in t i = 0 ; i < n; i+ + )

4 3 . {

4 4 . cou t< < i+ 1 < < " - ta la b a haqida k i r i t in g " <<endl;

4 5 . cout<<"Ism: " ; c in > > I ; c in .ig n o r e ( ) ;

4 6 . cout<<"Fam: " ; cin>>F;

4 7 . cout<<"Yosh: " ; cin>>Y;

4 8 . cout<<"S tep : " ; c in> >S ;

4 9 . T[ i ] . s e t ( Y , I , F , S ) ; }

50. } ***//saralb chiqarish uchun funksiya***

51. void Saralash\_show(a rray <Talaba, n> & ) {

52. fo r ( in t j = 0 ; j < n; j+ + ) {

53. fo r ( in t i = 0 ; i < n; i+ + ) {

54. i f (Y o sh [j]= =T[ i ] .y o s h ) {

55. cou t< < j+ 1 < < " - ta la b a haqida haqida"<<endl;

56. T[ i ] . g e t ( ) ; } } } }

57. }

58. in t main(){

59. / / ***n ta eLementdan iborat T nomLi konteyner***

60. a rray < t a la b a \_ 2 ::Talaba, n> T;

61. T [ 0 ] . s e t ( 2 2 , "Islom 'V 'S a lim o v ", 4 0 0 0 0 0 );

62. T [ 1 ] . s e t ( l 9 , "O q il" j MQodirovMj 600 0 0 0 );

63. T [ 2 ] . s e t ( 2 0 , "KomirV'Tashev", 1200000);

64. ta la b a \_ 2 ::sh ow (T );

65. cout<<endl<<"Y o sh la ri bofyicha saralanga ta la b a la r

r o fy x a t i " <<endl;

6 6 . s td ::s o r t (Y o s h .b e g in ( ) , Y o sh .en d ());

67. ta la b a \_ 2 ::S a ra la sh \_ sh ow (T );

6 8 . g e t c h a r ( ) ;

69. }

Dastur natijasi:

1 - talaba haqida haqida

Ismi: Islom

Familiyasi: Salimov

Yoshi: 22

Stependiyasi: 400000

2 - talaba haqida haqida

Ismi: Oqil

Familiyasi: Qodirov

Yoshi: 19

Stependiyasi: 600000

3 - talaba haqida haqida

Ismi: Komil

Familiyasi: Tashev

Yoshi: 20

Stependiyasi: 120000

Yoshlari bo‘yicha saralanga talabalar

ro‘yxati

1 - talaba haqida haqida

Ismi: Oqil

Familiyasi: Qodirov

Yoshi: 19

Stependiyasi: 600000

2 - talaba haqida haqida

Ismi: Komil

Familiyasi: Tashev

Yoshi: 20

Stependiyasi: 120000

3 - talaba haqida haqida

Ismi: Islom

Familiyasi: Salimov

Yoshi: 22

Stependiyasi: 400000

Quyida keltirilgan 3- jadvaldagi vazifalarda kiruvchi ma’lumotlar matinli

fayldan o‘zlashtrilishi kerak. Natijalar ham matinli faylga saqlanishi kerak.

Laboratoriya mashg‘ulotini bajarish bo‘yicha variantlar

**2-misol**

1 Talabalar haqida ma’lumotlar berilgan (class Talaba ). Ixtiyoriy talabani

(Fam, Iismi, Sharfi) bo‘yicha qidirib topuvchi dastur tuzing.

2 Kitoblar haqida ma’lumotlar berilgan (class Kitob). Ixtiyoriy kitobni nomi,

avtori va yili bo‘yicha qidirib topuvchi dastur tuzing.

3 Stol va stullar haqida ma’lumotlar berilgan (class Sto\_stul). Ixtiyoriy stol va

stulni 6 ta parametri bo‘yicha qidirib topuvchi dastur tuzing.

4

Sinf xonalari haqida ma’lumotlar berilgan (class Classroom). Ixtiyoriy sinfxonani

no meri va band yoki band emasligi bo‘yicha qidirib topuvchi dastur

tuzing.

5 Avtomobillar haqida ma’lumotlar berilgan (class Avto). Ixtiyoriy

avtomobil ni 6 ta parametri bo‘yicha saralovchi dastur tuzing.

6 Muzeylar haqida ma’lumotlar berilgan (class Museum). Eng ko‘p

eksponantlarga ega bo‘lgan muzeyni qidirib topuvchi dastur tuzing.

7 Sport komplekslari haqida ma’lumotlar berilgan (class Sprt\_compleks).

Tovarlari arzon sport kompleksini qidirib topuvchi dastur tuzing.

9

Mashxur insonlar haqida ma’lumotlar berilgan (class Famous\_person).

Yaratilgan kashfiyotlarning nomi bo‘yicha Mashxur insonlarni qidirib

topuvchi dastur tuzing.

9 Universitetlar haqida ma’lumotlar berilgan (class University). Ilmiy

salohiyati eng yuqori 3 ta universitetni qidirib topuvchi dastur tuzing.

10 Litseylar haqida ma’lumotlar berilgan (class Lyceums). Olinpiadada 3 yil

ichida g‘o lib bo‘lgan Litseyni qidirib topuvchi dastur tuzing.

11

Mashxur futbolistlar haqida ma’lumotlar berilgan (class

Famous\_futballer). Eng ko‘p urilgan gollar, sariq va qizil kartochkalar,

chiroyli va muhim gollari bo‘yicha mashxur futbaklchini qidirib topuvchi

dastur tuzing.

12

Kompyuterlar haqida ma’lumotlar berilgan (class Computers). Ushbu

komyuterlarning qaiysi birlari kompyuter o‘yinlarini qo‘llab quvvatlashishini

va ularning sonini aniqlovchi dastur tuzing.

13 Telefonlar haqida ma’lumotlar berilgan (class Telephone). Yadrolari va

operativ xotirasi kuchli telefonni qidirib topuvchi dastur tuzing.

14 Restoranlar haqida ma’lumotlar berilgan (class Restaurants). Milliy

taomlari ko‘p restoranni qidirib topuvchi dastur tuzing.

15 Taomlar haqida ma’lumotlar berilgan (class Meal). Suyuq va qo‘yiq

taomlarni nomalri asosida qidirib topuvchi dastur tuzing.

16 Hayvonlar haqida ma’lumotlar berilgan (class Hayvon). Yozziyi

hayvonlarni turi bo‘yicha topuvchi dastur tuzing.

17 Parrandalar haqida ma’lumotlar berilgan (class Parranda). Eng ko‘p

tuxum qo‘yadigan parannalarni turi qidirib topuvchi dastur tuzing.

19 A’lochi talabalar haqida ma’lumotlar berilgan (class Alochilar). Fanni va

undan olgan bahosi bo‘yicha a‘lochi talabani qidirib topuvchi dastur tuzing.

19 Uy hayvonlari haqida ma’lumotlar berilgan (class Uy\_hayvon). Eng foydali

uy hayvonini foydali jihatlari bo‘yicha qidirib topuvchi dastur tuzing.

20

Kompyuter o‘yin dasturlari haqida ma’lumotlar berilgan (class Games).

O‘ngni rivojlantiruvchi kompyuter o‘yinini har xil parametrlar asosisda

qidirib topuvchi dastur tuzing.

21

Past o‘zlashtiruvchi talabalar haqida ma’lumotlar berilgan (class Passiv).

Eng past o‘zlashtiruvchi talabani har xil parametlarri bo‘yicha qidirib

topuvchi dastur tuzing.

22 Viloyatlar haqida ma’lumotlar berilgan (class Viloyatlar). Ixtiyoriy

viloyatni har xil parametrlar bo‘yicha qidirib topuvchi dastur tuzing.

2.1.2. vector sinfidan foydalanib, amaliy dasturlar yaratish

Ishning maqsadi: C++ dasturlash tilida vecot sinflaridan foydalanib,

konteynernerlardan foydalanish ko‘nikmalarini shakllantirish.

Masalaning qo‘yilishi:

vector asosida yaratilgan to‘plam elementlarini vectorning maxsus

funksiyalari yordamida qayta ishlash.

Masala: Berilgan int turidagi to‘plamda - ishorali elementlarning eng

kattasini o‘chiruvchi dastur tuzing.

Masalani yechish g‘oyasi: talaba nomli yangi nomlar fazosi yartiladi. Uning

tarkibiga Talaba classi va Ism, Fam, Yosh, Step nomli to‘plamlar joylashtiriladi.

Talaba classining ruxsat berilmagan maydonlaridan foydalanish uchun friend

funksiya yaratiladi.

Dastur matni:

1 . #include " s td a fx .h "

2 . #include <iostream>

3. #include <vector>

4 . using namespace System;

5. using namespace s td ;

6 . namespace V e c to r la r { / / ***nomLar sohasi yaratiLdi***

7 . in t max,index;

8 . ***//vektorga qiymat o czLashtirish***

9 . void s e t\_ V e c to r ( s td ::v e c to r<i n t > & , in t n) {

1 0 . fo r ( in t i = 0 ; i < n; i+ + )

1 1 . { in t m; cou t< < i+ 1<<" son: " ; cin>>m;

12. A.push\_back(m);

13. i f (A[ i ] < 0 ) { max = A[ i ] ; } }

14. } / / ***Vektor qiyatlarni chiqarish***

15. void g e t\_V e c to r ( s td ::v e c to r<i n t > & , in t r ) {

16. cout<<"Tof plam e lem e n tla r i"<<endl;

17. fo r ( in t k : A ) { s td ::c o u t << k << " " ;

18. } cout<<endl;

19. } / / ***vaektorning max elementini toppish funksiyasi***

20. void Max \_V ec to r(std ::v e c to r<i n t > & , in t г ) {

2 1 . fo r ( in t i = 0 ; i < n; i+ + ) {

22. i f (A[ i ] < 0 && max<A[i]){m a x = A[ i ] ; in d e x = i;} }

23. cout<<"Max= " <<max<<endl;

24. } / / ***vektorning max elementini o cchirish***

25. void De le te\_Max\_Ve ctor(std::v e c to r<i n t > & ) {

26. A.e r a s e (A.b e g in ( )+ in d e x ) ;} }

27. in t main() {

28. cout<<"B e rilg an in t tu r id a g i t o f plamda - i s h o r a l i

elementlarning maksimalini o f chiru vchi d a s tu r "<<endl;

29. in t n; cin>>n;

30. s t d : :v e c to r<i n t > A;

31. V e c to r la r ::s e t\_V e c to r (A ,n ) ;

32. V e c to r la r ::g e t\_V e c to r (A ,n );

33. V e c to rla r::M a x \_V e c to r(A ,n );

34. V ec torlar::Delete\_Max\_V ec tor(A);

35. V e c to r la r ::g e t\_V e c to r (A ,n );

36. g e t c h a r ( ) ;

37. }

Dastur natijasi:

Berilgan int turidagi to‘plamdan ishorali elementlarning maksimalini o‘chiruvchi

dastur

4

1 son: 2

2 son: -4

3 son: -55

4 son: 3

To‘plam elementlari

2 -4 -55 3

Max= -4

To‘plam elementlari

2 -55 3

Laboratoriya mashg‘ulotini bajarish bo‘yicha variantlar

Quyidagi masalalardagi funksiyalar vector sinfidan foydalanilgan holda

yechilishi kerak. Har-bir masalada kiritish va chiqarish funksiyalari hamda vector

elementlarni fayldan yuklovchi va faylga saqlovchi funksiyalar bo‘lishi shart.

**3-misol**

1. Vector elementlarini max va min elementlarini o‘chiruvchi funksiyalar

yarating.

2. Berilgan N ta ixtiyoriy sonni max va min lari o‘rtasidagi sonlarini o‘chiruvchi

funksiyalar yarating.

3. Vector ning 1- manfiy elementi bilan ohirgi manfiy elementi o‘rtasidagi

elementlarini chiqaruvchi va o‘chiruvchi funksiyalar yarating.

4. Vector elementlari ichidan ixtiyoriy qidirilga son bor yo‘qligini aniqlash va

ushu sonni o‘chiruvchi funksiyalari yarating.

5. Vectorni ixtiyoriy berilgan index gacha bo‘lgan elementlarini o‘chiruvchi

funksiyalar yarating.

6. Vectorni ixtiyoriy qidirilgan elementigacha bo‘lgan elementlarini o‘chiruvchi

funksiyalar yarating.

7. Vectorni eng katta elementigacha bo‘lgan elementlarini o‘chiruvchi

funksiyalar yarating.

8. Vectorni eng kichik elementigacha bo‘lgan elementlarini o‘chiruvchi

funksiyalar yarating.

9. Masiivni eng kichik elementidan keyingi elementlarini chiqaruvchi va

o‘chiruvchi funksiyalar yarating.

10.Vectorni eng katta elementidan keyingi elementlarini chiqaruvchi va

o‘chiruvchi funksiyalar yarating.

11.Vectorda qidirilgan ixtiyoriy son nechta ekanligini aniqlovchi va ushbu sonni

o‘chiruvchi funksiyalar yarating.

12.Vectorda bir xil elementlarini o‘chiruvchi funksiyalar yarating.

13.Vectorda bir xil elementlar nechta ekanligini aniqlovchi va ushbu

elementlarni o‘chiruvchi funksiyalar yarating.

14.Ikkita ixtiyoriy vectorni bir xil ekanligini aniqlovchi va bir xil bo‘lsa bittasini

o‘chiruvchi funksiyalar yarating.

15.Ikkita ixtiyoriy vectorlarni nechta elementi bir xil emasligini aniqlovchi va bir

bo‘lmaganlarini o‘chiruvchi funksiyalar yarating.

16.A vectordan B vectorni tashkil qilish mumkin yoki mumkin emasligini

aniqlovchi va tashkil qilib bo‘lsa B vectorni o‘chiruvchi funksiyalar yarating.

17.A vectordan B vectorni elementlarini o‘chiruvchi funksiyalar yarating.

19.Vectorda eng ko‘p takrorlangan elementlarini chiqaruvchi va o‘chiruvchi

funksiyalar yarating.

19.Vectorda eng kam takrorlangan elementlarini chiqaruvchi va o‘chiruvchi

funksiyalar yarating.

20.Vectordan ixtiyor index dagi elementini topuv chi va o‘chiruvchi funksiyalar

yarating.