

MASSIVLAR BILAN ISHLASH

Reja

- Massiv nima?
- C++ dasturlash tilida massivni e'lon qilish.
- C++ dasturlash tilida massiv elementlariga murojaat qilish.
- C++ dasturlash tilida massiv elementlariga qiymat berish.
- Amaliy mashqlar.

Shartli belgilar



Eslab qoling



Bilib oling



Misol uchun



Mumkin emas



Uyga topshiriqlar



Amaliy yordam

Massiv nima?



Eslab qoling

Massiv bu - bir turga mansub bo'lgan yagona nom bilan saqlanuvchi tartiblangan ma'lumotlar majmuasi. Massivlar yagona o'zgaruvchi bilan kompyuter xotirasiga saqlanadi, uning elementlari ma'lum bir indekslar bilan tartiblab joylashtiriladi. Massivlar yagona nom bilan bir nechta qiymatni o'zida mujassamlashtiradi.



Qo'shimcha ma'lumot



Misol uchun

C++ dasturlash tilida massiv quyidagi qismlardan tashkil topadi:

- Massiv nomi.
- Massiv elementlarining tipi.
- Massivning uzunligi.

C++ dasturlash tilida massivni e'lon qilish



Bilib oling

Dastur ishlatilishi davomida massivlar aniq nomga ega bo'lishi va uning elementlari ma'lum bir turda bo'lishi kerak. Bir o'lchovli massivlar kompyuter xotirasiga quyidagi shaklda saqlanadi



Qo'shimcha ma'lumot

Array in C++

array variable

`arr [0];`

index of the element
to be accessed





Bilib oling

C++ dasturlash tilida massiv hosil qilish qoidasi:

```
dataType arrayName[arraySize];
```

- **dataType** - bir guruhga mansub elementlar to'plamining tipi.
- **arrayName** - element to'plamiga murojaat qilish uchun nom.
- **arraySize** - to'plamdagi elementlar soni.



Bilib oling

C++ dasturlash tilida massiv hosil qilish qoidasi:

```
int Array[7];
```

- **int** - bir guruhga mansub elementlar to'plamining tipi.
- **Array** - element to'plamiga murojaat qilish uchun nom.
- **7** - to'plamdagi elementlar soni.



Eslab qoling

C++ dasturlash tilida massivni 4 xil usul orqali ifodalash ya'ni e'lon qilish mumkin.

Har bitta usulning o'ziga yarasha yaxshi va yomon tomonlari mavjud hisoblanadi. Ular bilan quyidagi slaydlarda tanishib chiqamiz.



Misol uchun

C++ dasturlash tilida massiv hosil qilishning 1-usuli.

```
int Array[7];
```



Misol uchun

C++ dasturlash tilida massiv hosil qilishning 2-usuli.

```
int n = 10  
int Array[n];
```



Misol uchun

C++ dasturlash tilida massiv hosil qilishning 3-usuli.

```
int Array[] = {10, 20, 30, 40};
```



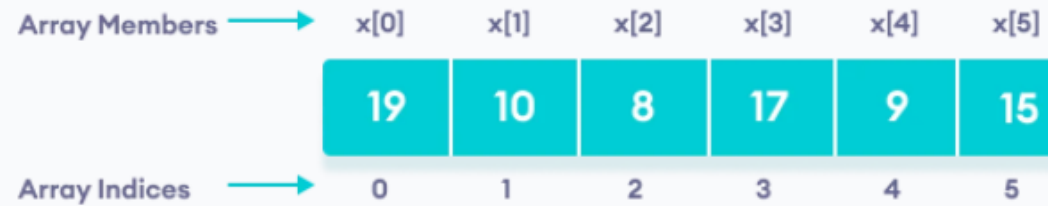
Misol uchun

C++ dasturlash tilida massiv hosil qilishning 4-usuli.

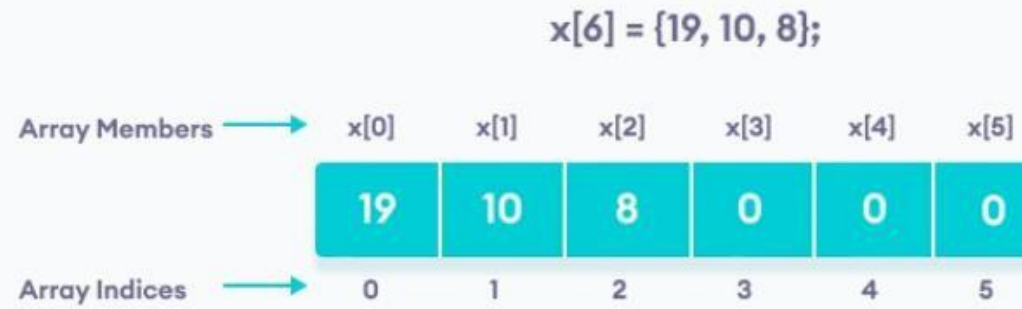
```
int Array[6] = {10, 20, 30, 40};
```



```
// declare and initialize an array  
int x[6] = {19, 10, 8, 17, 9, 15};
```



```
// store only 3 elements in the array  
int x[6] = {19, 10, 8};
```





Misol uchun

```
double balance[5] = {1000.0, 2.0, 3.4, 17.0, 50.0};
```

```
string cars[4] = { "Volvo", "BMW", "Ford", "Mazda" };
```

```
char letters[] = { 'A', 'B', '+', '-', '0', '1' };
```

```

#include<iostream>

using namespace std;

int main(){

    int Raqamlar[] = {15, 18, 25, 22, 14, 18};

    cout << " Massivning xotiradan olgan xajmi: " << sizeof(Raqamlar) << endl;
    cout << " Massivning elementining xotiradan olgan xajmi: " << sizeof(Raqamlar[0]) << endl;

    int n = sizeof(Raqamlar) / sizeof(Raqamlar[0]);

    for(int i :- 0; i < n; i++){
        cout << Raqamlar[i] << " ";
    }

    return 0;
}
```

C++ dasturlash tilida massiv elementlariga murojaat qilish



Eslab qoling

C++ dasturlash tilida massiv elementiga murojaat qilish uchun dastavval C++ dasturlash tilida yaratilgan massivning nomi yoziladi va to'rtburchak qavs ochgan holda qavs ichiga massiv elementi joylashgan indeks yoziladi. Shu orqali C++ dasturlash tilida massiv elementiga murojaat qilish mumkin bo'ladi.

```
ArrayName[index];
```



Eslab qoling

```
int arr[] = { 2, 3, 15, 8, 48, 13 };
```

element	2	3	15	8	48	13
index	0	1	2	3	4	5

```
// arr[0] = 2  
// arr[1] = 3  
// arr[2] = 15  
// arr[3] = 8  
// arr[4] = 48  
// arr[5] = 13
```




Misol uchun

```
int arr[] = { 2, 3, 15, 8, 48, 13 };  
  
for(int i=0; i<6; i++){  
    cout<<arr[i]<<" ";  
}
```

```
2 3 15 8 48 13
```


C++ dasturlash tilida massiv
elementlariga qiymat berish



Eslab qoling

C++ dasturlash tilida massiv elementiga qiymat berish uchun massivning nomi yoziladi va to'rtburchak qavs ochgan holda qavs ichiga massiv elementi joylashgan indeks yoziladi. Undan so'ng esa teng belgisi qo'yilgan holda massiv elementiga qiymat o'zlashtiriladi.

```
ArrayName[index] = value;
```

```
int arr[] = { 2, 3, 15, 8, 48, 13 };
```

```
cout<<"Avvalgi qiymatlar: ";  
for(int i=0; i<6; i++){  
    cout<<arr[i]<<" ";  
}
```

```
arr[1] = 45;
```

```
arr[3] = 77;
```

```
cout<<"\nKeyingi qiymatlar: ";  
for(int i=0; i<6; i++){  
    cout<<arr[i]<<" ";  
}
```

```
Avvalgi qiymatlar: 2 3 15 8 48 13  
Keyingi qiymatlar: 2 45 15 77 48 13
```



Amaliy mashqlar



Uyga topshiriq

1

int turidagi elementlardan iborat massiv berilgan. Massivni birinchi yoki oxirgi elementi 6 bolsa, true qaytaruvchi dastur tuzing.

```
firstLast6([1, 2, 6]) → true
```

```
firstLast6([6, 1, 2, 3]) → true
```

```
firstLast6([13, 6, 1, 2, 3]) → false
```



Amaliy yordam



Uyga topshiriq

2

2 ta int turida a va b massiv berilgan. Agar ularning birinchi elementlari yoki oxirgi elementlari teng bolsa, true qaytaring.

```
commonEnd([1, 2, 3], [7, 3]) → true
```

```
commonEnd([1, 2, 3], [7, 3, 2]) → false
```

```
commonEnd([1, 2, 3], [1, 3]) → true
```




Uyga topshiriq

3

int turida uzunligi 3 ga teng bo'lgan massiv berilgan. Massivni birinchi elementini olib oxiriga qo'ying va natijani qaytaring.

```
rotateLeft3([1, 2, 3]) → [2, 3, 1]  
rotateLeft3([5, 11, 9]) → [11, 9, 5]  
rotateLeft3([7, 0, 0]) → [0, 0, 7]
```



Uyga topshiriq

4

int turida uzunligi 3 ga teng bo'lgan massiv berilgan. Massivni teskari qilib (aylantirib) qaytaring.

`reverse3([1, 2, 3]) → [3, 2, 1]`

`reverse3([5, 11, 9]) → [9, 11, 5]`

`reverse3([7, 0, 0]) → [0, 0, 7]`



Uyga topshiriq

5

int turida uzunligi 3 ga teng bo'lgan massiv berilgan. Massivni birinchi va oxirgi elementini solishtirib, kattasini toping va massivni barcha elementlarini shu katta songa almashtiring. Natijani qaytaring.

```
maxEnd3([1, 2, 3]) → [3, 3, 3]
```

```
maxEnd3([11, 5, 9]) → [11, 11, 11]
```

```
maxEnd3([2, 11, 3]) → [3, 3, 3]
```



Amaliy yordam



Uyga topshiriq

6

int turida uzunligi 3 ga teng bo'lgan 2ta a va b massiv berilgan.
Shu massivlarni o'rtasidagi elementlaridan iborat yangi massivni qaytaring.

```
middleWay([1, 2, 3], [4, 5, 6]) → [2, 5]
```

```
middleWay([7, 7, 7], [3, 8, 0]) → [7, 8]
```

```
middleWay([5, 2, 9], [1, 4, 5]) → [2, 4]
```



Uyga topshiriq

7

`int` turida `a` massiv berilgan. Yangi massiv `a` massivni uzunligidan 2 marta ko'p va yangi massivni oxirgi elementi `a` massivni oxirgi elementiga teng. Yangi massivning qolgan elementlari 0 ga teng. Yangi massivni qaytaring.

`makeLast([4, 5, 6]) → [0, 0, 0, 0, 0, 6]`

`makeLast([1, 2]) → [0, 0, 0, 2]`

`makeLast([3]) → [0, 3]`



Uyga topshiriq

8

int turida uzunligi 3 ga teng bo'lgan massiv berilgan. Agar massivda 2 elementidan keyin 3 elementi kelsa 3 elementini 0 ga almashtiring va qaytaring.

`fix23([1, 2, 3]) → [1, 2, 0]`

`fix23([2, 3, 5]) → [2, 0, 5]`

`fix23([1, 2, 1]) → [1, 2, 1]`



Uyga topshiriq

9

Uzunligi 2 ga teng bo'lgan int turida 2ta a va b massiv berilgan. Qaysi massivni elementlarini yigindisi katta bo'lsa, o'sha massivni qaytaring. Teng bo'lsa, a massivni qaytaring.

```
biggerTwo([1, 2], [3, 4]) → [3, 4]
```

```
biggerTwo([3, 4], [1, 2]) → [3, 4]
```

```
biggerTwo([1, 1], [1, 2]) → [1, 2]
```



Uyga topshiriq

10

Juft uzunlikdagi int turida massiv berilgan. O'rtadagi 2ta elementdan iborat yangi massivni qaytaring.

```
makeMiddle([1, 2, 3, 4]) → [2, 3]
```

```
makeMiddle([7, 1, 2, 3, 4, 9]) → [2, 3]
```

```
makeMiddle([1, 2]) → [1, 2]
```



Amaliy yordam



Uyga topshiriq

11

Uzunligi 2 ga teng bo'lgan 2ta int turida a va b massiv berilgan. Uzunligi 4 ga teng bo'lgan yangi massiv elementlari a va b massiv elementlaridan iborat. Yangi massivni qaytaring.

```
plusTwo([1, 2], [3, 4]) → [1, 2, 3, 4]  
plusTwo([4, 4], [2, 2]) → [4, 4, 2, 2]  
plusTwo([9, 2], [3, 4]) → [9, 2, 3, 4]
```




Uyga topshiriq

12

`int` turida array berilgan. Arrayni birinchi va oxirgi elementlarini almashtiring va qaytaring.

`swapEnds([1, 2, 3, 4]) → [4, 2, 3, 1]`

`swapEnds([1, 2, 3]) → [3, 2, 1]`

`swapEnds([8, 6, 7, 9, 5]) → [5, 6, 7, 9, 8]`



Uyga topshiriq

13

Toq uzunlikdagi array berilgan. O'rtadagi 3ta elementdan iborat yangi massivni qaytaring.

```
midThree([1, 2, 3, 4, 5]) → [2, 3, 4]
```

```
midThree([8, 6, 7, 5, 3, 0, 9]) → [7, 5, 3]
```

```
midThree([1, 2, 3]) → [1, 2, 3]
```



Uyga topshiriq

14

Toq uzunlikdagi massiv berilgan. Massivni birinchi, oxirgi va o'rtasidagi elementlarini solishtiring va eng kattasini qaytaring.

`maxTriple([1, 2, 3]) → 3`

`maxTriple([1, 5, 3]) → 5`

`maxTriple([5, 2, 3]) → 5`



Uyga topshiriq

15

`int` turida massiv berilgan. Agar massivda 1dan keyin 3 kelsa, biz 1 elementini “omadsiz 1” deb ataymiz. Agar massivni birinchi 2ta yoki oxirgi 2ta elementlari birortasi “omadsiz 1” bo‘lsa, `true` qaytaring.

```
unlucky1([1, 3, 4, 5]) → true
```

```
unlucky1([2, 1, 3, 4, 5]) → true
```

```
unlucky1([1, 1, 1]) → false
```



Amaliy yordam

1-holat

1

3

4

5

6

7

8

9

3

2

True

2-holat

7

1

3

5

6

7

8

9

3

2

True

3-holat

7

8

4

5

6

7

8

9

1

3

True

4-holat

5

6

4

5

1

3

8

9

3

2

False



Uyga topshiriq

16

int turida istalgan uzunlikda 2ta a va b massiv berilgan. Shu massivlarni birinchi elementlaridan iborat yangi massivni qaytaring. Agar birorta massiv uzunligi 0 bo'lsa, o'sha massivni hisobga olmang.

```
front11([1, 2, 3], [7, 9, 8]) → [1, 7]
```

```
front11([1], [2]) → [1, 2]
```

```
front11([1, 7], []) → [1]
```



Uyga topshiriq

17

`int` turida massiv berilgan. Massivdagi elementlar yig'indisini toping, lekin massivda 13 soni bo'lsa, 13ni ham va undan 1ta keyingi elementni ham hisobga olmang.

`sum13([1, 2, 2, 1]) → 6`

`sum13([13, 1, 2, 13, 2, 1, 13]) → 3`

`sum13([1, 1]) → 2`

`sum13([1, 2, 2, 1, 13]) → 6`



Uyga topshiriq

18

int turida massiv berilgan. Massivdagi elementlar yig'indisini toping, lekin massivda uchraydigan 6 dan boshlanib, keyingi 7 soni uchraguncha bo'lgan ketma-ketlikdagi elementlarni hisobga olmang.

`sum67([1, 2, 2]) → 5`

`sum67([1, 2, 2, 6, 99, 99, 7]) → 5`

`sum67([1, 1, 6, 7, 2]) → 4`

`sum67([2, 7, 6, 2, 6, 7, 2, 7]) → 18`



Uyga topshiriq

19

`int` turida massiv va `n` soni berilgan. Agar massivdagi barcha qo'shni-qo'shni juft elementlaridan birortasi `n` soniga teng bo'lsa, `true` qaytaring.

```
isEverywhere([1, 2, 1, 3], 1) → true  
isEverywhere([1, 2, 1, 3], 2) → false  
isEverywhere([1, 2, 1, 3, 4], 1) → false
```



Uyga topshiriq

20

int turida massiv berilgan. Agar massivda 2 sonidan 1ta keyin 2 soni kelsa yoki 4 sonidan 1ta keyin 4 kelsa, hamda shu ikki shart bir vaqtda bajarilmasa, true qaytaring.

`either24([1, 2, 2]) → true`

`either24([4, 4, 1]) → true`

`either24([4, 4, 1, 2, 2]) → false`



E'tiboringiz uchun
rahmat