

# «ԾՐԱԳՐԱՎՈՐՄԱՆ ՀԻՄՈՒՆՔՆԵՐ» դասընթաց

այլ ոլորտներից դեպի տեխնոլոգիական ոլորտ  
սկսնակների համար



edu2020.am

## ԴԱՍ #7



ՀԱՅ-ՌՈՒՄԱԿԱՆ  
ՀԱՄԱԼՍԱՐԱՆ



ՀԱՅԱՍՏԱՆԻ ՀԱՆՐԱՊԵՏՈՒԹՅԱՆ  
ԲԱՐՁՐ ՏԵԽՆՈԼՈԳԻԱԿԱՆ  
ԱՐԴՅՈՒՆԱԲԵՐՈՒԹՅԱՆ ՆԱԽԱՐԱՐՈՒԹՅՈՒՆ

# Մտատիկ զանգվածներ

1. Զանգվածը (array) նույն տիպի տվյալների հաջորդականություն է
2. Տվյալները տեղադրվում են հիշողության մեջ հաջորդաբար
3. Հնարավոր է դիմել հաջորդականության ցանկացած էլեմենտին

Զանգվածի  
տարրեր (int)  
Ինդեքս

|    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|
| 20 | 24 | 26 | 31 | 45 | 65 | 34 |
| 0  | 1  | 2  | 3  | 4  | 5  | 6  |



# Մտատիկ զանգվածներ

```
int main() {
```

```
    int a[10];
```

```
    short s[5];
```

```
    long long_array[51];
```

```
    float float_array[6];
```

```
    double double_array[9];
```

```
}
```

10 չափանի **int** տիպի զանգված  
“a” – ն փոփոխականի անունն է:

9 չափանի **double** տիպի  
զանգված





# Մտասիկ զանգվածներ

```
#include <iostream>
int main() {

    int a[10];
    short s[10];
    long long_array[10];
    float float_array[10];
    double double_array[10];
    std::cout << "size of a " << sizeof(a) << std::endl;
    std::cout << "size of s " << sizeof(s) << std::endl;
    std::cout << "size of double_array " << sizeof(double_array) << std::endl;
}
```

<https://repl.it/@VahagVardanyan/ArraySize>



# Զանգվածների ներկայացումը հիշողության մեջ

```
#include <iostream>
```

```
int main() {  
    int a[5];  
    a[0] = 5;  
    a[1] = 7;  
    a[2] = 89;  
    a[3] = 8;  
    a[4] = 21;  
}
```

Զանգվածի առաջին էլեմենտը  
տեղակայված է 0 ինդեքսով:  
a[0] – առաջին էլեմենտ  
a[1] – երկրորդ էլեմենտ  
a[4] – վերջին՝ հինգերորդ էլեմենտ

|        | a[0] | a[1] | a[2] | a[3] | a[4] |
|--------|------|------|------|------|------|
| a →    | 5    | 7    | 89   | 8    | 21   |
| Ինդեքս | 0    | 1    | 2    | 3    | 4    |



# Զանգվածի սկզբնաբեքավորում

```
#include <iostream>
```

```
int main() {  
    int a[10];  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl; // Bad Code  
    }  
  
    for (int i = 0; i < 10; i++) {  
        a[i] = 5;  
    }  
    std::cout << std::endl;  
    std::cout << "After init" << std::endl;  
    std::cout << std::endl;  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
}
```

<https://repl.it/@VahagVardanyan/arrayinit>



# Զանգվածի սկզբնաբաժանում

```
#include <iostream>
```

```
int main() {  
    int a[10];  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
  
    for (int i = 0; i < 10; i++) {  
        a[i] = 5;  
    }  
    std::cout << std::endl;  
    std::cout << "After init" << std::endl;  
    std::cout << std::endl;  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
}
```

Զանգվածի առաջին էլեմենտը  
տեղակայված է 0 ինդեքսով:  
a[0] – առաջին էլեմենտ  
a[1] – երկրորդ էլեմենտ  
a[9] – վերջին էլեմենտ

<https://repl.it/@VahagVardanyan/arrayinit>



# Չանգվածի սկզբնաբեքավորում

```
#include <iostream>
```

```
int main() {  
    int a[10];  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
  
    for (int i = 0; i < 10; i++) {  
        a[i] = 5;  
    }  
    std::cout << std::endl;  
    std::cout << "After init" << std::endl;  
    std::cout << std::endl;  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
}
```

i փոփոխականը ստանում է արժեքներ 0 – ից 9  
a[i] – զանգվածի հերթական էլեմենտ  
Տպել զանգվածի բոլոր էլեմենտները

<https://repl.it/@VahagVardanyan/arrayinit>





# Զանգվածի սկզբնաթեքավորում

```
#include <iostream>
```

```
int main() {  
    int a[10];  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
  
    for (int i = 0; i < 10; i++) {  
        a[i] = 5;  
    }  
    std::cout << std::endl;  
    std::cout << "After init" << std::endl;  
    std::cout << std::endl;  
    for (int i = 0; i < 10; i++) {  
        std::cout << a[i] << std::endl;  
    }  
}
```

Զանգվածի բոլոր էլեմենտներին  
վերագրել 5

<https://repl.it/@VahagVardanyan/arrayinit>



# Զանգվածի հայտարարման եղանակներ

1. `int arr[50];`
2. `const int n = 50;`  
`int arr[n];`
3. `int arr[] = { 10, 20, 30, 40 };`
4. `int arr[6] = { 10, 20, 30, 40, 50, 60 };`
5. `int arr[6] = { 10, 20, 30, 40 };`



# const բանալի բառ

```
#include <iostream>
int main() {
    const int arraySize = 5;
    int a[arraySize];
    for (int i = 0; i < arraySize; i++) {
        a[i] = 5;
    }
}
```

|        | a[0] | a[1] | a[2] | a[3] | a[4] |
|--------|------|------|------|------|------|
| a →    | 5    | 5    | 5    | 5    | 5    |
| հնդէքս | 0    | 1    | 2    | 3    | 4    |



# Տպել զանգվածի առաջին և վերջին էլեմենտները

```
#include <iostream>

int main() {
    const int size = 6;

    int arr[size] = { 10, 20, 30, 40 };

    std::cout << arr[0] << std::endl;

    std::cout << arr[size - 1] << std::endl;
}
```

<https://repl.it/@VahagVardanyan/ArrayFirstLast>



# Զանգվածի էլեմենտների գումար

```
#include <iostream>

int main() {
    const int n = 5;
    int arr[n];
    for (int i = 0; i < n; i++) {
        std::cin >> arr[i];
    }

    int sumOfArray = 0;
    for (int i = 0; i < n; i++) {
        sumOfArray += arr[i];
    }
    std::cout<<sumOfArray<<std::endl;
}
```

<https://repl.it/@VahagVardanyan/sumOfArray>





# Զանգվածի էլեմենտների գումար

```
#include <iostream>
```

```
int main() {  
    const int n = 5;  
    int arr[n];  
    for (int i = 0; i < n; i++) {  
        std::cin >> arr[i];  
    }  
  
    int sumOfArray = 0;  
    for (int i = 0; i < n; i++) {  
        sumOfArray += arr[i];  
    }  
    std::cout<<sumOfArray<<std::endl;  
}
```

Մուտքագրել զանգվածի  
էլեմենտները

<https://repl.it/@VahagVardanyan/sumOfArray>



# Զանգվածի էլեմենտների գումար

```
#include <iostream>
```

```
int main() {  
    const int n = 5;  
    int arr[n];  
    for (int i = 0; i < n; i++) {  
        std::cin >> arr[i];  
    }  
  
    int sumOfArray = 0;  
    for (int i = 0; i < n; i++) {  
        sumOfArray += arr[i];  
    }  
    std::cout<<sumOfArray<<std::endl;  
}
```

Ամեն քայլում զանգված  
հերթական էլեմենտը  
գումարվում է *sumOfArray*-ին

<https://repl.it/@VahagVardanyan/sumOfArray>



# Զանգվածի էլեմենտների գումար

```
#include <iostream>
```

```
int main() {  
    const int n = 5;  
    int arr[n];  
    for (int i = 0; i < n; i++) {  
        std::cin >> arr[i];  
    }  
  
    int sumOfArray = 0;  
    for (int i = 0; i < n; i++) {  
        sumOfArray = sumOfArray + arr[i];  
    }  
    std::cout<<sumOfArray<<std::endl;  
}
```

i == 0: sumOfArray = 0 + a[0];  
i == 1: sumOfArray = 0 + a[0] + a[1];  
i == 2: sumOfArray = 0 + a[0] + a[1] + a[2];  
i == 3: sumOfArray = 0 + a[0] + a[1] + a[2] + a[3];  
i == 4: sumOfArray = 0 + a[0] + a[1] + a[2] + a[3] + a[4];

<https://repl.it/@VahagVardanyan/sumOfArray>



# Խնդիրներ

1. Ջանգվածի զույգ էլեմենտների գումար
2. Ջանգվածի այն էլեմենտների քանակը, որոնց արժեքը մեծ է 0-ից
3. Ջանգվածի այն էլեմենտների քանակը, որոնց արժեքը մեծ է զանգվածի վերջին էլեմենտի արժեքից
4. Ջանգվածի բոլոր էլեմենտներին գումարել 10



# Էլեմենտի որոնում զանգվածում

```
#include <iostream>
int main() {
    const int size = 5;
    int arr [size];
    std::cout << "Input Array of " << size << " element" << std::endl;
    for (int i =0; i < size; i++) {
        std::cin >> arr[i];
    }
    int nubmer_to_find;
    std::cout << "Input number to find" << std::endl;
    std::cin >> nubmer_to_find;
    bool found = false;
    for (int i = 0; i < size; i++) {
        if (arr[i] == nubmer_to_find) {
            found = true;
            break;
        }
    }
    if (found) {
        std::cout << "Number is in array" << std::endl;
    } else {
        std::cout << "Number is not in array" << std::endl;
    }
}
```

<https://repl.it/@VahagVardanyan/ArrayFind>





# Զանգվածի սահմաններից դուրս դիմում

```
#include <iostream>

int main() {
    const int n = 10;
    int arr[n];
    for (int i = 0; i < n; i++) {
        arr[i] = 100;
    }

    std::cout << arr[-1] << std::endl;
    std::cout << arr[10000000] << std::endl;
}
https://repl.it/@VahagVardanyan/OutOfBoundArray
```



# Գտնել զանգվածի ամենամեծ էլեմենտը

```
#include <iostream>
```

```
int main() {  
    const int size = 5;  
    int arr[size];  
    for (int i = 0; i < size; i++) {  
        std::cin >> arr[i];  
    }  
  
    int max = arr[0];  
    for (int i = 1; i < size; i++) {  
        if (arr[i] > max) {  
            max = arr[i];  
        }  
    }  
    std::cout << "Max of array " << max << std::endl;  
}
```

<https://repl.it/@VahagVardanyan/ArrayMax>



# Գտնել զանգվածի ամենամեծ էլեմենտը

```
#include <iostream>
int main() {
    const int size = 5;
    int arr[size];
    for (int i = 0; i < size; i++) {
        std::cin >> arr[i];
    }

    int max = arr[0];
    for (int i = 1; i < size; i++) {
        if (arr[i] > max) {
            max = arr[i];
        }
    }
    std::cout << "Max of array " << max <<
    std::endl;
}
```

25, 10, 67, -9, 2

max = 25

i == 1: max = 25  
i == 2: max = 67  
i == 3: max = 67  
i == 4: max = 67

<https://repl.it/@VahagVardanyan/ArrayMax>



# Զանգվածի չափ

```
#include <iostream>

int main() {
    int numbers[] = {8, 25, 36, 44, 52, 60, 75, 89};
    std::cout << sizeof(numbers)/sizeof(int) <<
std::endl;
}
```

<https://repl.it/@VahagVardanyan/ArraySizeComp>



# Խնդիրներ

- Առանձնացնել զանգվածի զույգ և կենտ թվերը 2 տարբեր զանգվածներում: Նոր զանգվածների չափը հավասար է սկզբնական զանգվածի չափին (լրացնել 0-ով):
- Հաշվել, թե տրված թիվը քանի՞ անգամ է հանդիպում տրված զանգվածում:
- Հաշվել զանգվածի բոլոր էլեմենտների քառակուսիների գումարը
- Գրել ծրագիր, որը մուտքում կստանա զանգված և զանգվածի բոլոր զույգ թվերին կգումարի 1 իսկ բոլոր կենտ թվերից կհանի 1
- Գտնել զանգվածի երկրորդ ամենամեծ թիվը





# Տնային աշխատանք

- Խնդիրներ 10 - 18



# Շնորհակալություն. Հարցե՞ր

