

Array data structure

Array (Array) is one of the oldest and most important data structures. Arrays can store some fixed elements and these elements have the same type. Most data structures use arrays to implement algorithms. Here are the important concepts related to Arrays.

Array (Array) is one of the oldest and most important data structures. Arrays can store some fixed elements and these elements have the same type. Most data structures use arrays to implement algorithms. Here are the important concepts related to Arrays.

Element : Each item stored in an array is called an element.

Index : Each position of an element in an array has a numerical index used to identify the element.

The array consists of records of the same type, fixed size, each element defined by the index

Arrays are basic data structures that are constantly allocated

Advantages of array:

1. Access elements with time constant $O(1)$
2. Use memory effectively
3. Local memory calculation

Defect

Cannot change the size of the array when the program is executing

Dynamic

Dynamic array: allocating memory for arrays dynamically during the process of running the program in C as malloc and calloc, in C++ is new

Using dynamic arrays we start with an array of 1 element, when the number of elements exceeds the ability, we double the size of the array and copy the old array element into the first half of the new array.

Advantages : avoid wasting memory when declaring a large size array from the beginning

Disadvantages :

1. You must perform additional copy operation every time you change the size.
2. Some operation time is no longer constant

Representation of array data structures

Arrays can be declared in a variety of ways in programming languages. To illustrate, we use array declaration in C language:

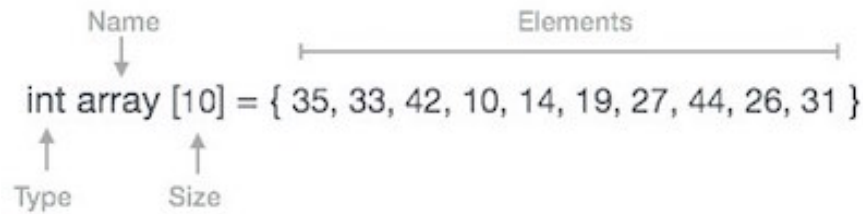
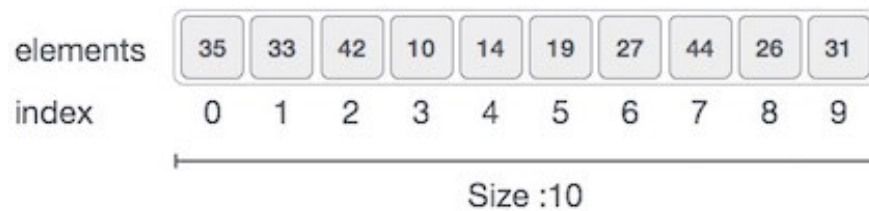


Illustration of element and index:



Here are some points to remember about array data structures:

Index begins with 0.

Array length is 10, meaning the array can store 10 elements.

Each element can be accessed through that element's index. For example, we can get the element's value at index 6 to 27.

Basic operations are supported by arrays

Here are the basic operations supported by an array:

Browse : Print all array elements according to the way each element is printed.

Insert : Add an element to the array at the given index.

Delete : Delete an element from the array at the given index.

Search : Search for an element using good index by value.

Update : Update the value of an element at some index.

In the C language, when an array is initialized to its original size, it assigns default values ??to the elements of the array in the following order:

Data type Default value
bool false
char 0
int 0
float 0.0
double 0.0f
void
wchar_t 0

Operation inserts elements into arrays

Insert operation is to insert one or more data elements into an array. Depending on the requirement, the new element can be inserted into the first, last, or any given index position of the array.

In the next section we will deploy the insert operation in a real example. In this example, we will insert data at the end of the array.

For example

Suppose LA is an unordered linear array with N elements and K being a positive integer satisfying $K \leq N$. Below is an algorithm to insert element A into position K of the LA array.

Algorithms

1. Start
2. Assign $J = N$
3. Assign $N = N + 1$
4. Repeat steps 5 and 6 when $J > K$
5. Assign $LA [J + 1] = LA [J]$
6. Assign $J = J - 1$
7. Assign $LA [K] = ITEM$
8. Finish

The following is the full code of the above algorithm in C language:

```
#include <stdio.h>
int main () { int LA [] = { 1 , 3 , 5 , 7 , 8 }; int item = 10 , k = 3;
for (int i = 0; i < 5; i++) printf ("%d\n", LA[i]);
for (int i = 4; i >= k; i--) LA[i+1] = LA[i];
LA[k] = item;
for (int i = 0; i < 6; i++) printf ("%d\n", LA[i]);
return 0; }
```

Compiling and running the above C program will result:

```
Danh sach phan tu trong mang ban dau:
LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 7
LA[4] = 8
Danh sach phan tu cua mang sau hoat dong chen:
LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 10
LA[4] = 7
LA[5] = 8
```

Operation deletes the element from the array

The delete operation is to delete an existing element from an array and reorganize the remaining elements in that array.

For example

Suppose LA is a linear array with N elements and K is a positive integer that satisfies $K \leq N$. Here is the algorithm to delete an element in the LA array at position K.

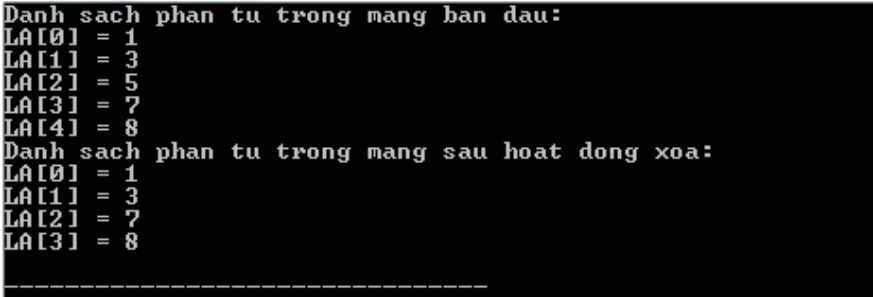
Algorithms

1. Start
2. Assign $J = K$
3. Repeat steps 4 and 5 while $J < N$
4. Assign $LA [J-1] = LA [J]$
5. Assign $J = J + 1$
6. Assign $N = N-1$
7. Finish

The following is the full code of the above algorithm in C language:

```
#include <stdio.h>
main () { int LA [] = { 1 , 3 , 5 , 7 , 8 }; int k = 3 , n = 5 ;
```

Compiling and running the above C program will result:



```
Danh sach phan tu trong mang ban dau:
LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 7
LA[4] = 8
Danh sach phan tu trong mang sau hoat dong xoa:
LA[0] = 1
LA[1] = 3
LA[2] = 7
LA[3] = 8
```

Search activity

You can perform the search for the element in the array based on the element's value or index.

For example

Suppose LA is a linear array with N elements and K is a positive integer that satisfies $K \leq N$. Here is the algorithm to find an ITEM element using sequential search method (or search for route). calculation).

Algorithms

1. Start
2. Assign $J = 0$
3. Repeat steps 4 and 5 when $J < N$
4. If $LA [J]$ is equal to ITEM TO STEP 6
5. Assign $J = J + 1$
6. In value J, ITEM
7. Finish

The following is the full code of the above algorithm in C language:

```
#include <stdio.h>
main () { int LA [] = { 1 , 3 , 5 , 7 , 8 }; int item = 5 , n = 5 ;
```

Compiling and running the above C program will result:

```
Danh sach phan tu trong mang ban dau:
LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 7
LA[4] = 8
Tim thay phan tu 5 tai vi tri 3
-----
```

Update operation (Update operation)

The update operation is to update the value of an existing element in the array at the given index.

Algorithms

Suppose LA is a linear array with N elements and K is a positive integer that satisfies $K \leq N$. Here is an algorithm to update the element value at position K of the LA array.

1. Start
2. Set up $LA [K-1] = ITEM$
3. Finish

The following is the full code of the above algorithm in C language:

```
#include <stdio.h>
int main () { int LA [] = { 1 , 3 , 5 , 7 , 8 }; int k = 3 , n = 5 ,
```

Compiling and running the above C program will result:

```
Danh sach phan tu trong mang ban dau:
LA[0] = 1
LA[1] = 3
LA[2] = 5
LA[3] = 7
LA[4] = 8
Danh sach phan tu trong mang sau hoat dong update:
LA[0] = 1
LA[1] = 3
LA[2] = 10
LA[3] = 7
LA[4] = 8
-----
```

According to Tutorialspoint

Previous article: Setting environment in Data structure

Next lesson: What is algorithm?

You finished reading the article "**Array data structure**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.