

Arreglos en C++

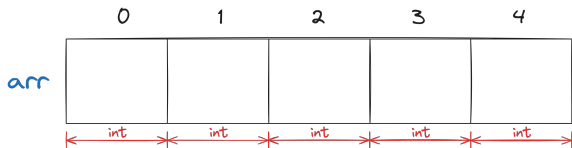
Conceptos básicos

Tomás Peiretti

¿Qué es un arreglo (array)?

Un arreglo es una **serie de elementos de un mismo tipo** almacenados en posiciones contiguas de memoria que pueden ser referenciados individualmente a través de un índice.

Por ejemplo, un arreglo llamado **arr** que contiene 5 valores de tipo **int** puede representarse de la siguiente manera:



¿Qué es un arreglo (array)?

Al igual que las variables, un arreglo debe ser declarado antes de ser utilizado:

tipo nombre[tamaño];

Donde, el número de elementos (tamaño) del arreglo debe ser una expresión constante

- Puede utilizarse una constante literal (10, 200, 1573, etc), ó
- Puede utilizarse una constante global

```
1 #define tam 200
2 int main() {
3     int arr1[10]; // constante literal
4     int arr2[tam]; // constante global
5     return 0;
6 }
```

Acceder a un elemento

Para acceder a un elemento del arreglo se utiliza la sintaxis **nombre[indice]**. Ejemplos:

```
1 int main() {
2     char arreglo[100];
3     // imprimir el elemento que se encuentra en
  el indice 43
4     cout << arreglo[43] << endl;
5     // guardar en el 3ra posicion el caracter X
6     arreglo[2] = 'X';
7     // imprimir los primeros 10 elementos
8     for (int i=0; i<10; i++)
9         cout << arreglo[i] << endl;
10
11     return 0;
12 }
```

Inicializar un arreglo

Por defecto, los arreglos de alcance local se encuentran sin inicializar, por lo que contienen valores "basura" almacenados en las diferentes posiciones.

```
1 int main() {  
2     int arr0[5]; // sin inicializar  
3     int arr1[] = {1, 23, -2, 4, 1};  
4     int arr2[5] = {10, 15, 2};  
5     int arr3[5] = {};  
6     int arr4[] {5, 14, 25};  
7  
8     int arr5[5];  
9     for(int i=0; i<5; i++)  
10         arr5[i]=7;  
11  
12     return 0;  
13 }
```

	arr0[0]	arr0[1]	arr0[2]	arr0[3]	arr0[4]
arr0	19182734	-991822	6901824	16882	73617261

	arr1[0]	arr1[1]	arr1[2]	arr1[3]	arr1[4]
arr1	1	23	-2	4	1

	arr2[0]	arr2[1]	arr2[2]	arr2[3]	arr2[4]
arr2	10	15	2	0	0

	arr3[0]	arr3[1]	arr3[2]	arr3[3]	arr3[4]
arr3	0	0	0	0	0

	arr4[0]	arr4[1]	arr4[2]
arr4	5	14	25

	arr5[0]	arr5[1]	arr5[2]	arr5[3]	arr5[4]
arr5	7	7	7	7	7

Tamaño físico vs tamaño lógico

El **tamaño físico** de un arreglo es el tamaño que se define al momento de declarar un arreglo. Es la cantidad maxima de elementos/posiciones que se pueden utilizar.

En cambio, el **tamaño lógico** refiere a la cantidad de posiciones/elementos que se estan utilizando y siempre **debe ser menor o igual al tamaño físico**.

```
1 // Ejemplo: arreglo con TF=10 y TL=5
2 int main() {
3     int arreglo[10] = {1, 2, 3, 4, 5};
4     for (int i=0; i<5; i++)
5         cout << arreglo[i] << endl;
6     return 0;
7 }
```

0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	19182734	-991822	6901824	16882	73617261

Operaciones con arreglos

¿Qué operaciones debemos saber realizar con los arreglos para aprobar AEDD?

- Recorrer (de izq a derecha, en un rango, de der a izq, etc)
- Buscar un elemento
- Unir dos arreglos
- Invertir un arreglo
- Eliminar/agregar un elemento
- Ordenar (BubbleSort, MergeSort, SelectionSort, InsertionSort)

Ejercicios

- Beecrowd 1173
- Beecrowd 1174
- Beecrowd 1177
- Beecrowd 1178
- Beecrowd 1180
- Codeforces ([GYM] 100 Easy Problems) P25
- Beecrowd 1175
- Beecrowd 1171
- Beecrowd 1410
- Codeforces ([GYM] 100 Easy Problems) P02