

## C++学习笔记：array与vector

array大小是固定的，可以使用vector。

```
#include <iostream>
#include <vector>
using namespace std;

int main() {
    // 创建向量用于存储整型数据
    vector<int> vec;
    int i;

    // 显示 vec 初始大小
    cout << "vector size = " << vec.size() << endl;

    // 向向量 vec 追加 5 个整数值
    for(i = 0; i < 5; i++){
        vec.push_back(i);
    }

    // 显示追加后 vec 的大小
    cout << "extended vector size = " << vec.size() << endl;

    return 0;
}
```

执行以上代码，输出结果：

```
vector size = 0
extended vector size = 5
```

## C++学习笔记：char数组初始化

char数组直接初始化是特殊的，最后一个元素为null。

```
char a1[] = {'C', '+', '+'};           // 初始化，没有 null
char a2[] = {'C', '+', '+', '\0'};     // 初始化，明确有 null
char a3[] = "C++";                     // null 终止符自动添加
const char a4[6] = "abcdef";           // 报错，没有 null 的位置
```

a4是错误的，实际上a4[7]="abcdef"是正确的。a1的size为3，但a3的size为4。

## C++学习笔记：size, sizeof

sizeof是对对象求内存大小，例如int a1[] = { 1,2,3 }，则sizeof(a1) = 12。size指所含元素个数。

对向量可以使用vector.size()求元素个数。

对string型数组，可以使用strlen，strlen从起始位置计数到“\0”结束，所以对字符型数组不同的初始化方式会有不同的strlen值。

```
int main()
{
    char str1[] = "hello";
    char str2[] = {'h','e','l','l','o'};
    cout << "str1:" << strlen(str1) << endl; // 结果是5
    cout << "str2:" << strlen(str2) << endl; // 结果是21
    return 0;
}
```

对其他类型数组，C++中没有求size的函数，一般可以使用宏定义一个求size的函数。

```
#define ARRAY_LEN(array,len){len = (sizeof(array) / sizeof(array[0]));}
```