**数组作为函数参数**

**一、数组作为函数参数**

1. 数组名作为函数参数
   1. 一一对应，我们所定义的函数形参列表要与传进去的实参一一对应
   2. C语言规定是值传递，即单向传递，只能实参给形参
   3. 数组名传递给函数参数，相当于传递数组的首地址进去（在后面的指针中会讲到）【注意：该方式是地址传递，可以相互改变的，在函数中改变后，该值就会改变了】
   4. 因此我们如果不想被改变，一般会加上const关键字
2. 代码来实验一下
   1. 值传递

int sub1(int);

int main(){

int n = 5;

int n1 = sub1(n);

printf("changed value : %d\n", n1); 4

printf("original val %d\n", n); 5

return 0;

}

// 简单一点实现n-1函数

int sub1(int n){

return n-1;

}

* 1. 地址传递，自行运行一下【注意：可以用断点调试看一下形参合实参的地址】

void sub1(int[], int);

void print(int[], int);

int main(){

int n = 4;

int arr[4] = {1, 2, 3, 4}; // 很奇怪我用 arr[n] = {1, 2, 3, 4} 会出现excess错误

print(arr, n);

sub1(arr, n);

print(arr, n);

}

// 数组中每个元素都-1,我们不需要返回值，我们只需要看会不会变

void sub1(int arr[], int n){

for(int i = 0; i < n; i++){

arr[i] -= 1;

}

}

// 打印函数

void print(int arr[], int n){

for(int i = 0; i < n; i++)

printf("%d ", arr[i]);

printf("\n");

}

* 1. 等我们学到指针就会明白了