

컴퓨터공학 All in One

C/C++ 문법, 자료구조 및 심화 프로젝트 (나동빈)
제 8강 - 배열

학습 목표

배열

- 1) 배열을 사용해야 하는 이유에 대해서 바르게 이해합니다.
- 2) 배열의 선언 및 접근 방법에 대해서 이해합니다.

배열

배열을 사용하는 이유

- 1) 3개의 숫자가 있을 때 가장 큰 숫자를 구하는 프로그램을 작성한다고 생각합시다.
- 2) 기본적으로 변수의 이름을 a, b, c로 설정하면 된다는 것을 예상할 수 있습니다.

배열

배열을 사용하는 이유

- 1) 그렇다면 숫자가 100개라면 어떻게 처리할 수 있을까요?
- 2) 비효율적이긴 하지만 변수 100개를 만들어서 선언할 수는 있습니다.

배열

배열을 사용하는 이유

- 1) 하지만 숫자가 1,000,000개라면? 이 때는 일일이 변수를 만들어서 처리할 수 없을 겁니다.
- 2) 따라서 동일한 자료형을 여러 개 담는 기술이 필요합니다. 바로 배열입니다.

배열

배열을 사용하는 이유

- 1) 배열을 사용하면 동일한 성격의 데이터를 다수 표현할 수 있습니다.

인덱스	0	1	2	3	4	5	6	7	8	9
데이터	6	5	4	3	9	8	0	1	2	7

배열

배열의 선언과 초기화

배열의 선언은 다음과 같이 할 수 있습니다.

자료형 배열명[배열의 크기] = {초기화 값};

초기화 값은 없을 수도 있습니다.

배열

배열의 선언과 초기화

다음과 같은 문법으로 배열을 초기화하고 개별적인 값에 접근할 수 있습니다.

```
#include <stdio.h>

int main(void) {
    int a[10] = { 6, 5, 4, 3, 9, 8, 0, 1, 2, 7 };
    int i;
    for (i = 0; i < 10; i++) {
        printf("%d ", a[i]);
    }
    system("pause");
}
```


배열

배열의 원소 중에서 최대값 찾기

```
#include <stdio.h>
#include <limits.h>

int main(void) {
    int a[10] = { 6, 5, 4, 3, 9, 8, 0, 1, 2, 7 };
    int i, maxValue = INT_MIN;
    for (i = 0; i < 10; i++) {
        if (maxValue < a[i]) maxValue = a[i];
    }
    printf("%d\n", maxValue);
    system("pause");
}
```

배열

INT_MIN

- 1) INT_MIN은 최댓값을 구하기 위해 자주 사용되는 기능입니다.
- 2) <limits.h> 헤더파일에 정의가 되어 있는 것으로 int형 범위의 최솟값을 반환합니다.
- 3) INT_MAX 또한 사용할 수 있습니다.

배열

문자열과 배열

- 1) 원시적인 C언어는 기본적으로 자체적인 문자열 자료형을 제공하지 않습니다.
- 2) 따라서 C언어에서는 문자(Character)를 여러 개 묶어 놓는 형태로 문자열을 표현합니다.
- 3) C++에서는 이러한 불편함을 알고 있기 때문에 자체적으로 string 자료형을 제공합니다.

배열

문자열과 배열

- 1) 일단 C언어에서는 하나의 문자는 1바이트만을 담으므로 다음과 같이 문자의 배열을 사용합니다.
- 2) 다시 말해 문자열은 사실 배열이라고 할 수 있습니다.

```
char a[20] = "HELLO WORLD"
```

배열

문자열과 배열

- 1) 기본적으로 문자열을 선언할 때는 문자열의 크기보다 배열의 크기가 크도록 해야 합니다.
- 2) 문자열을 입력 받을 때 및 출력할 때는 %s라는 형식 지정자를 사용합니다.

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int main(void) {
    char a[20];
    scanf("%s", &a);
    printf("%s\n", a);
    system("pause");
}
```

배열

문자열과 배열

- 1) C언어에서 문자열로 문자의 배열을 쓴다는 점을 처음에 배웠을 때는 귀찮게 생각이 될 수 있습니다.
- 2) 하지만 C언어는 그 대신 특정한 인덱스의 문자에 바로 접근할 수 있다는 장점이 있습니다.

```
#include <stdio.h>

int main(void) {
    char a[20] = "Hello World";
    a[4] = ' ', ' ';
    printf("%s\n", a);
    system("pause");
}
```

배열

문자열에 포함된 'l' 개수 출력하기

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>

int main(void) {
    char a[] = "Hello World";
    int count = 0;
    for (int i = 0; i <= 10; i++) {
        if (a[i] == 'l') count++;
    }
    printf("%d\n", count);
    system("pause");
    return 0;
}
```

배운 내용 정리하기

배열

- 1) 배열을 이용하면 많은 양의 데이터를 쉽게 처리할 수 있습니다.
- 2) C언어에서의 문자열은 내부적으로 문자의 배열입니다.