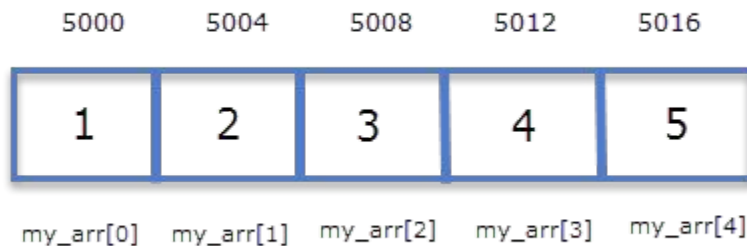


# 1차원 배열



# 복습) 1차원 배열이란?

## ▶ 배열이란?

배열(array)은 같은 타입의 변수들로 이루어진 유한 집합

배열 요소(element): 배열을 구성하는 값

인덱스(index): 배열의 주소

인덱스는 0부터 시작

|        |        |        |        |
|--------|--------|--------|--------|
| 1      | 3      | -9     | 0      |
| arr[0] | arr[1] | arr[2] | arr[3] |

우리 반이 최고다...!

# 코딩 대회



## 대회 안내

- 총 문제는 3개 (난이도 하, 중, 상)
- 문제 별 제한 시간 있음
- 문제 다 풀었을 시 선생님을 불러 답이 맞는지 확인
- 반 별로 문제 당 몇 분 소요되었는지로 평가
- 문제 공개 8분 이후 다른 친구를 도와줄 수 있음. 단, 대신 코딩 해주는 것은 불가능하며 말로만 도움을 줄 수 있음.
- 부정행위(대신 코딩, 인터넷 검색) 적발 시, 10분 시간 추가

## 문제: “덧셈이 필요해” (난이도: 하)

문제 1.

주어진 배열 a와 b에 대해 원소별 덧셈을 수행해 출력하시오.

힌트: 첫번째 원소끼리 더한 후 출력, 두 번째 원소끼리 더한 후 출력, ...

```
int a[] = { 1,3,2,5,0 };
```

```
int b[] = { 8,1,2,3,1 };
```

## 풀이 공개

```
int main() {  
    int a[] = { 1,3,2,5,0 };  
    int b[] = { 8,1,2,3,1 };  
    int result;  
    for (int i = 0; i < sizeof(a) / sizeof(int); i++)    // 배열의 요소 개수만큼 반복  
    {  
        result = a[i] + b[i];  
        printf("%d ", result);    // 배열의 인덱스에 반복문의 변수 i를 지정  
    }  
  
    return 0;  
}
```

출력

9 4 4 8 1

## 문제: “빈도 빈도 bean do” (난이도: 중)

제한시간: 16분

주어진 배열 `numbers`에 대해 원소 별 빈도를 출력하시오. (단, 배열 원소는 항상 0 이상 5 미만이고 배열의 길이는 5이다.)

```
int numbers[] = { 3,3,4,0,1 };
```

힌트: 숫자별 빈도를 담은 배열을 만든다

예시)

```
int numbers[] = { 1,3,1,2,1 };
```

```
숫자: 1, 빈도: 3  
숫자: 2, 빈도: 1  
숫자: 3, 빈도: 1
```

## 풀이 공개

```
int main() {  
  
    int numbers[] = { 3,3,4,0,1 };//숫자  
    int frequency[] = { 0,0,0,0,0 };//빈도수 배열  
    int n = sizeof(numbers) / sizeof(int);//배열 길이  
  
    for (int i = 0; i < n; i++) {  
        int idx = numbers[i];//숫자를 frequency 배열의 인덱스로 저장  
        frequency[idx]++;  
    }  
  
    for (int i = 0; i < n; i++) {  
        if (frequency[i] > 0) {//빈도수가 1 이상인 경우에만  
            printf("숫자: %d, 빈도: %d\n", i, frequency[i]);//출력  
        }  
    }  
    return 0;  
}
```

### 출력

|        |       |
|--------|-------|
| 숫자: 0, | 빈도: 1 |
| 숫자: 1, | 빈도: 1 |
| 숫자: 3, | 빈도: 2 |
| 숫자: 4, | 빈도: 1 |



## 문제: “사용자가 왕이다” (난이도: 상)

**제한시간: 30분**

정수로 이루어진 배열 numbers에 대하여 사용자가 원하는 n번째 순서의 값을 출력하시오. 단, 배열의 길이는 10으로 고정되어 있음. (\*\*정해진 풀이 없음)

```
int numbers[] = { 0,2,8,1,2,7,5,5,3,1 };
```

예시)

```
int numbers[] = { 9,2,3,4,5,6,7,7,9,1 };
```

몇 번째 수를 원하시나요?: 3  
요청하신 3번째 수는 3입니다.

## 문제 풀이

```
int main() {
    int numbers[] = { 0,2,8,1,2,7,5,5,3,1 };
    int n = 10;

    //숫자 정렬
    while (1) {
        int pass = 0;
        for (int i = 0; i < n-1; i++) {
            if (numbers[i] > numbers[i + 1]) {
                int tmp = numbers[i];
                numbers[i] = numbers[i + 1];
                numbers[i + 1] = tmp;
            }
            else {
                pass++;
            }
        }
        if (pass == 9) {
            break;
        }
    }
    printf("\n몇 번째 수를 원하시나요?: ");
    int user;
    scanf("%d", &user); //사용자 입력
    printf("요청하신 %d번째 수는 %d입니다.\n", user, numbers[user-1]);
    return 0;
}
```

### 출력

몇 번째 수를 원하시나요?: 5  
요청하신 5번째 수는 2입니다.

몇 번째 수를 원하시나요?: 3  
요청하신 3번째 수는 1입니다.

몇 번째 수를 원하시나요?: 7  
요청하신 7번째 수는 5입니다.

몇 번째 수를 원하시나요?: 1  
요청하신 1번째 수는 0입니다.

몇 번째 수를 원하시나요?: 10  
요청하신 10번째 수는 8입니다.

# 고생

# 많으셨습니다!

