

---

# C 프로그래밍 및 실습

## 7. 배열

세종대학교

---

# 목차

---

- 1) 배열이란?
- 2) 배열 초기화
- 3) 배열 응용
- 4) 실수 배열과 문자 배열
- 5) 다차원 배열

## 1) 배열이란?

---

- **변수를 여러 개 만들어야 하는 상황을 생각해 보자.**
  - 사용자로부터 5개의 정수를 입력 받아 변수에 저장하고, 이 값을 출력하는 프로그램은 다음과 같이 작성할 수 있다.
  - 하지만, 정수가 100개라면? → 배열을 사용하여 해결

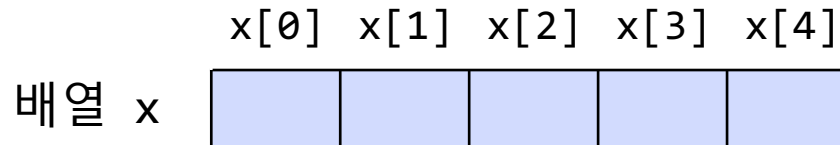
```
int main(void){  
  
    int x0, x1, x2, x3, x4;  
  
    scanf("%d%d%d%d%d", &x0, &x1, &x2, &x3, &x4);  
  
    printf("%d %d %d %d %d\n", x0, x1, x2, x3, x4);  
  
    return 0;  
  
}
```

# 1) 배열이란?

---

## ■ 배열이란?

- 같은 자료형의 변수 여러 개를 하나로 묶은 자료형
- 배열을 이용하여 많은 변수를 한번에 선언하고, 저장된 데이터를 처리할 수 있음
- 앞 예에서 사용한 5개의 변수를 배열 형식으로 표현하면  
✓ x0, x1, x2, x3, x4    →    x[0], x[1], x[2], x[3], x[4]



# 1) 배열이란?

## 배열의 선언

- 변수이름 뒤에 필요한 변수의 개수를 명시
- 배열 선언 구문

**자료형** 변수명[ 배열크기 ];

- 배열 선언 예)

✓ int x[5];

✓ int a, b, x[10];

// 일반 변수 a, b 와 함께 선언도 가능

```
int main(void){  
    int x[5];  
    ...  
    return 0;  
}
```

배열 선언의 예

- 정수형 배열, 배열의 이름 x, 배열의 크기는 5
- 5개의 정수를 저장하는 배열 x

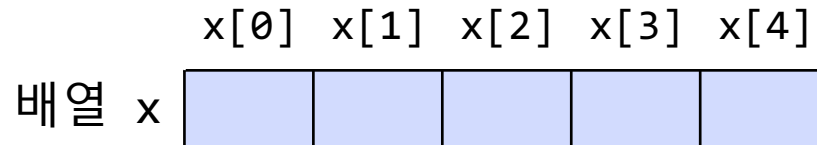
배열 x

x[0]	x[1]	x[2]	x[3]	x[4]

# 1) 배열이란?

## ■ 배열의 사용

- 배열의 **원소** (or 요소): 배열을 구성하는 각 변수를 지칭
- 배열의 원소는 대괄호 [ ] 안에 번호를 넣어서 구분
  - ✓  $x[0]$ ,  $x[1]$ ,  $x[2]$ ,  $x[3]$ ,  $x[4]$
- 배열의 **첨자** or **인덱스**(index)
  - ✓ 대괄호 [ ] 안의 번호로 배열에서 각 원소의 위치를 나타냄
  - ✓ **배열의 인덱스는 항상 0부터 시작함**
  - ✓ 예) 크기가 5인 배열의 인덱스는 0~4까지 임

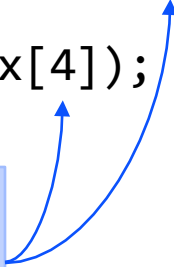


## 1) 배열이란?

- 앞서 본 예제 프로그램을 배열을 이용하여 작성해보자.
  - 사용자로부터 5개의 정수를 입력 받아 변수에 저장하고, 이 값을 출력하는 프로그램

```
int main(void){  
    int x[5];  
  
    scanf("%d%d%d%d%d", &x[0], &x[1], &x[2], &x[3], &x[4]);  
  
    printf("%d %d %d %d %d\n",x[0],x[1],x[2],x[3],x[4]);  
  
    return 0;  
}
```

x[4]를 하나의 변수처럼 사용  
scanf에서 변수 앞에 & 붙여줌



[실행예시]

```
3 5 1 -3 4  
3 5 1 -3 4
```

## 1) 배열이란?

- 앞 프로그램에서 입력되는 정수가 100개라면?
  - 배열을 사용해서 변수 선언은 간단히 해결됨
  - 하지만, 입출력 부분은? → 반복문을 이용하여 해결
- 배열과 반복문과의 만남
  - 배열 원소의 인덱스가 0부터 시작하여 1씩 증가한다는 규칙을 이용하여 배열의 원소에 접근

```
printf("%d ", x[0]);  
printf("%d ", x[1]);  
printf("%d ", x[2]);  
printf("%d ", x[3]);  
printf("%d ", x[4]);
```



```
for( i=0 ; i<5 ; i++ )  
    printf("%d ", x[i]);
```



## 1) 배열이란?

---

- 배열을 이용하여 작성된 최종 프로그램
  - 사용자로부터 5개의 정수를 입력 받아 변수에 저장하고, 이 값을 출력하는 프로그램

```
int main(void){  
    int x[5], i;  
  
    for( i=0 ; i<5 ; i++ )  
        scanf("%d", &x[i]);  
  
    for( i=0 ; i<5 ; i++ )  
        printf("%d ", x[i]);  
    printf("\n");  
  
    return 0;  
}
```

## 1) 배열이란? (실습 1, 실습 2)

---

### [실습1]

- 크기가 7인 배열 x에 아래 점수를 저장하시오. **for문**

80, 71, 91, 95, 77, 79, 88

- for 문을 이용하여 80점 이상의 학생의 인덱스와 점수를 모두 출력하는 프로그램을 작성하시오.

출력 예시

0	80
2	91
3	95
6	88

---

### [실습2]

- 크기가 9인 배열 x를 선언하시오.
- 구구단 3단의 계산 값을 배열에 저장한 후, **for문**
- 배열 내용을 화면에 출력하시오. **for문**

출력 예시

3
6
9
.
.
.
27

## 2) 배열 초기화

- 배열 전체를 초기화하는 방법

- 배열을 선언할 때 만 중괄호 `{ }`를 이용하여 배열 전체를 초기화하는 것이 가능하다.

```
#include <stdio.h>
int main(void){
    int i=0, x[5] = {0, 1, 2, 3, 4} ;

    for(i=0;i<5;i++) {
        printf("%d ", x[i]);
    }
    return 0;
}
```

[실행결과]

0 1 2 3 4

## 2) 배열 초기화

- 초기값의 개수가 배열의 크기보다 작으면, 배열의 뒷부분은 0으로 채워진다.

```
#include <stdio.h>
int main(void){
    int i=0, x[5] = {0, 1, 2} ;

    for(i=0;i<5;i++) {
        printf("%d ", x[i]);
    }
    return 0;
}
```

[실행결과]

0 1 2 0 0

## 2) 배열 초기화

---

- 만약 모두 0으로 초기화 하고 싶다면

```
#include <stdio.h>
int main(void){
    int i=0, x[5] = {0} ;

    for(i=0;i<5;i++) {
        printf("%d ", x[i]);
    }
    return 0;
}
```

[실행결과]

0 0 0 0 0

## 2) 배열 초기화

- 배열의 크기를 지정하지 않으면? → 초기화에 사용된 원소 수에 의해 배열 크기가 결정된다
  - ✓ `sizeof( )` 연산자는 괄호 안의 변수 혹은 데이터 타입의 바이트 수를 알려준다

```
#include <stdio.h>
int main(void){
    int i=0, x[ ] = {0, 1, 2, 3, 4} ;

    for(i=0;i<5;i++) {
        printf("%d ", x[i]);
    }
    printf("\n배열의 크기 = %d \n", sizeof(x)/sizeof(int));
    return 0;
}
```

[실행 결과]

0 1 2 3 4  
배열의 크기 = 5

## 2) 배열 초기화

---

- 다음의 출력 결과는 무엇인가요?

```
#include <stdio.h>
int main(void){
    int x[]={10, 5, 4, 3, 20};
    printf("%d %d\n", x[2], x[4]);
    return 0;
}
```

### 3) 배열 응용 (예제)

- 크기가 5인 정수형 배열을 선언하고
- 배열의 원소를 순서대로 10, 20, 30, 40, 50으로 초기화 하시오.
- 원소를 순서대로 출력하시오.

#### 방법1: 쉬운 방법

```
#include <stdio.h>
int main(void){
    int x[]={10, 20, 30, 40, 50};
    printf("%d %d %d %d %d \n", x[0], x[1], x[2], x[3], x[4]);
    return 0;
}
```

#### 방법2: 반복문 이용

```
#include <stdio.h>
int main(void){
    int x[]={10, 20, 30, 40, 50};
    for(i=0; i<5; i++) printf("%d ", x[i]);
    return 0;
}
```



### 3) 배열 응용 (예제: 배열 원소의 합)

- 크기가 5인 정수형 배열을 선언하고
- 배열의 원소를 순서대로 3, 4, 5, 1, 3으로 초기화 하시오.

```
int x[5]={3, 4, 5, 1, 3};
```

- 배열의 원소의 합을 구하시오.

#### 방법1: 쉬운 방법

```
int x[5]={3, 4, 5, 1, 3};  
int sum=0;  
sum=x[0]+x[1]+x[2]+x[3]+x[4];
```

#### 방법2: 반복문 이용

```
int x[5]={3, 4, 5, 1, 3};  
int i=0, sum=0;  
for(i=0; i<5; i++) sum = sum + x[i];
```

### 3) 배열 응용 (예제: 배열 원소의 평균)

---

- 크기가 5인 배열을 선언하고 0으로 초기화 하시오.
- 사용자로부터 정수 5개를 입력 받으시오.

#### 방법1: 쉬운 방법

```
int x[5]={0};  
scanf("%d%d%d%d%d", &x[0], &x[1], &x[2], &x[3], &x[4]);
```

#### 방법2: 반복문 이용

```
int i=0, x[5]={0};  
for(i=0; i<5; i++) scanf("%d", &x[i]);
```

- ✓ 반복문을 이용하면 사용자로부터 요구사항이 변경될 경우에 쉽게 대응할 수 있다.

### 3) 배열 응용 (예제: 배열 원소의 평균)

- 사용자로부터 정수 5개를 입력 받아 합과 평균 구하기 전체 코드

```
#include <stdio.h>

int main(void){
    int i=0, sum=0, x[5]={0};
    printf("정수 5개를 입력하세요. ");
    for(i=0;i<5;i++) scanf("%d", &x[i]);
    for(i=0;i<5;i++) sum+=x[i];
    printf("%d %f \n", sum, sum/5.0);
    return 0;
}
```

[실행결과]

정수 5개를 입력하세요. 3 4 5 1 3

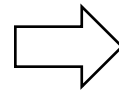
16 3.200000

### 3) 배열 응용 (실습 3)

- 크기가 10인 배열을 선언하고 0으로 초기화 하시오.
- 사용자로부터 정수 10개를 입력 받으시오.
- 10개의 정수 중에서 짝수는 짝수끼리 더하고 홀수는 홀수끼리 더한 후에 각각 합을 출력하시오.

입력 예시1

1 2 3 4 5 6 7 8 9 10

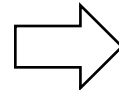


출력 예시1

30  
25

입력 예시2

2 4 6 8 10 12 14 16 18 20



출력 예시2

110  
0

## 4) 실수 배열과 문자 배열

- 실수 변수를 여러 개 만들어야 할 때 배열을 사용함
- 배열은 여러 개의 변수가 동일한 이름을 가지며, 구분은 인덱스로 함
- 실수라 해서 특별히 달라지는 점은 없음

- 실수 배열 선언

```
#include <stdio.h>
int main(void){
    double x[5];

    return 0;
}
```

크기가 5인 실수 배열 선언

## 4) 실수 배열과 문자 배열

- 초기화(1)

```
#include <stdio.h>
int main(void){
    double x[5]={1.0, 2.0, 3.0, 4.0, 5.0};
    int i=0;
    for(i=0;i<5;i++) printf("%.1f ", x[i]);
    printf("\n");
    return 0;
}
```

[실행결과]

1.0 2.0 3.0 4.0 5.0

## 4) 실수 배열과 문자 배열

---

- 초기화(2)

```
#include <stdio.h>
int main(void){
    double x[]={1.0, 2.0, 3.0, 4.0, 5.0};
    int i=0;
    for(i=0;i<5;i++) printf("%.1f ", x[i]);
    printf("\n");
    return 0;
}
```

## 4) 실수 배열과 문자 배열

---

- 초기화(3)

```
#include <stdio.h>
int main(void){
    double x[5]={1.0, 2.0, 3.0};
    int i=0;
    for(i=0;i<5;i++) printf("%.1f ", x[i]);
    printf("\n");
    return 0;
}
```

[실행결과]

1.0 2.0 3.0 0.0 0.0



## 4) 실수 배열과 문자 배열

---

- 초기화(4)

```
#include <stdio.h>
int main(void){
    double x[5]={0.0};
    int i=0;
    for(i=0;i<5;i++) printf("%.1f ", x[i]);
    printf("\n");
    return 0;
}
```

[실행결과]

0.0 0.0 0.0 0.0 0.0

## 4) 실수 배열과 문자 배열

- 실수 5개를 사용자에게서 입력 받고 합과 평균을 출력하시오.

```
#include <stdio.h>
int main(void){
    double x[5]={0.0}, sum=0.0;
    int i=0;
    printf("실수 5개를 입력하세요. ");
    for(i=0;i<5;i++) scanf("%lf", &x[i]);
    for(i=0;i<5;i++) sum=sum+x[i];
    printf("합 = %f, 평균 = %f \n", sum, sum/5);
    return 0;
}
```

[실행결과]

실수 5개를 입력하세요. 1.0 2.0 3.0 4.0 5.0

합 = 15.000000, 평균 = 3.000000

주의: scanf() 함수에서 double형 실수 서식 문자는 %lf

## 4) 실수 배열과 문자 배열

- 다음 코드의 결과는 무엇인가요?

```
#include <stdio.h>
int main(void){
    double x[5]={0.0}, max=0.0;
    int i=0;
    printf("실수 5개를 입력하세요. ");
    for(i=0;i<5;i++) scanf("%lf", &x[i]);
    max=x[0];
    for(i=1;i<5;i++) if(max<x[i]) max=x[i];
    printf("max = %f \n", max);
    return 0;
}
```

[실행결과]

실수 5개를 입력하세요. 7.0 8.0 1.0 9.0 2.0

max = 9.000000

## 4) 실수 배열과 문자 배열

---

- 최대값 구하기 작동 방식(1)



```
max = x[0];  
if(max < x[1]) max = x[1];  
if(max < x[2]) max = x[2];  
if(max < x[3]) max = x[3];  
if(max < x[4]) max = x[4];
```

## 4) 실수 배열과 문자 배열

- 최대값 구하기 작동 방식(2)

7.0	8.0	1.0	9.0	2.0
x[0]	x[1]	x[2]	x[3]	x[4]

```
max = 7.0;  
if(max < x[1]) max = 8.0;  
if(max < x[2]) max = 8.0;  
if(max < x[3]) max = 9.0;  
if(max < x[4]) max = 9.0;
```

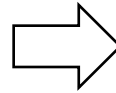
#### 4) 실수 배열과 문자 배열 (실습 4)

---

- 실수 5개를 입력 받은 후 입력 받은 순서와 반대로 출력하시오.

입력 예시

1.1 2.2 3.0 4.0 5.0



출력 예시

5.000000  
4.000000  
3.000000  
2.200000  
1.100000

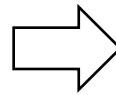
#### 4) 실수 배열과 문자 배열 (실습 5)

---

- 실수 5개를 입력 받은 후, 최솟값을 출력하는 코드를 작성하시오.

입력 예시

1.1 2.2 3.0 4.0 5.0



출력 예시

1.100000

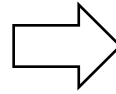
#### 4) 실수 배열과 문자 배열 (실습 6)

- (실습 5)에서 실수 5개를 입력 받은 후, 최솟값을 출력하는 코드를 작성하였다. (실습 5) 프로그램을 이용하시오.
- 최솟값이 저장되어 있는 배열에서 인덱스도 함께 출력하시오.

입력 예시

3.0 2.2 1.1 4.0 5.0

x[0] x[1] x[2] x[3] x[4]



출력 예시

1.100000

2



## 4) 실수 배열과 문자 배열

- 문자 배열 초기화
- 문자 배열 각 원소를 대문자로 변경하시오.

```
#include <stdio.h>
int main(void){
    int i;
    char ar[4]={'d','u','c','k'};
    ar[0]='D', ar[1]='U', ar[2]='C', ar[3]='K';

    for(i=0;i<4;i++) printf("%c ",ar[i]);
    printf("\n");
    return 0;
}
```

D U C K

## 4) 실수 배열과 문자 배열 (예제)

- 문자 배열을 출력하시오.
- 문자 배열 안에 있는 각 단어를 출력하시오.

```
#include <stdio.h>
int main(void){
    int i;
    char ar[9]={'d','u','c','k',' ','p','o','n','d'};

    for(i=0;i<9;i++) printf("%c ",ar[i]);
    printf("\n");
    for(i=0;i<4;i++) printf("%c ",ar[i]);
    printf("\n");
    for(i=5;i<9;i++) printf("%c ",ar[i]);
    printf("\n");
}
```

```
d u c k   p o n d
d u c k
p o n d
```

- 다른 자료형도 실수 배열과 문자 배열과 같은 형식으로 쓸 수 있다.

예) `unsigned int arr[3];`   `short score[3];`

## 5) 다차원 배열

---

- **배열**: 변수를 여러 개 만들기 위한 효과적인 방법
- **일차원 배열**: 여러 개의 변수를 구분할 때 인덱스 한 개만을 이용
- **이차원 배열**: 변수를 구분하기 위해 두 개의 인덱스를 이용

```
int a[100]; // 일차원 배열의 선언  
int b[10][10]; // 동일하게 100개의 변수가 생기지만 인덱스가 두 개임
```

## 5) 다차원 배열

- 왜 인덱스가 두 개 필요할까?

학생들의 성적을 저장하는 배열을 선언한다고 할 경우  
5명의 국어 성적만을 고려한다면

```
int score[5]={78, 93, 20, 44, 88};
```

만약 영어성적까지 고려해야 한다면

10개의 데이터를 저장해야 한다

```
int score[10]={78, 89, 93, 100, 20, 30, 44, 55, 88, 12};
```

첫 번째 학생의 성적

만약 국어 성적의 합을 계산하려고 한다면 두 가지  
성적이 섞여있어서 계산이 쉽지 않을 것이다

## 5) 다차원 배열

- 해결책은 무엇일까?

```
int score[5][2]={{{78, 89}, {93, 100}, {20, 30}, {44, 55}, {88, 12}}};
```

score[0][0]    첫 번째 학생의 국어 성적  
score[0][1]    첫 번째 학생의 영어 성적

- 첫 번째 인덱스 : 학생의 번호를 의미
- 두 번째 인덱스 : 과목의 번호를 의미(국어 0, 영어 1)
- 이러한 방식으로 구분해 주면
  - 학생을 기준으로 계산하는 것이 가능하고
  - 과목을 기준으로 계산하는 것도 쉬워진다

## 5) 다차원 배열

---

### ▪ 배열 선언

```
#include <stdio.h>
int main(void){
    int a[5][2];
    int b[10][10];
    return 0;
}
```

→ 총 10개의 변수가 생긴다 (2개씩 5개)

→ 총 100개의 변수가 생긴다

## 5) 다차원 배열

### ▪ 학생 별 국어, 영어, 수학 성적 출력하기

```
#include <stdio.h>
int main(void){
    int a[2][3]={ {10,20,50}, {20,30,40}}; // {국어,영어,수학}
    int i=0, j=0;
    for(i=0;i<2;i++) {
        for(j=0;j<3;j++) {
            printf("a[%d][%d] = %d \n", i, j, a[i][j]);
        }
    }
    return 0;
}
```

#### [실행결과]

a[0][0] = 10

a[0][1] = 20

a[0][2] = 50

a[1][0] = 20

a[1][1] = 30

a[1][2] = 40

## 5) 다차원 배열

- 학생 별 국어, 영어, 수학 성적 입력한 후 출력하기

```
#include <stdio.h>
int main(void){
    int a[2][3]={ 0 }; // {국어,영어,수학}
    int i=0, j=0;
    for (i = 0; i<2; i++) {
        for (j = 0; j<3; j++) {
            scanf("%d", &a[i][j]);
        }
    }
    for(i=0;i<2;i++) {
        for(j=0;j<3;j++) {
            printf("a[%d][%d] = %d \n", i, j, a[i][j]);
        }
    }
    return 0;
}
```

[출력예시]

```
a[0][0] = 10
a[0][1] = 20
a[0][2] = 50
a[1][0] = 20
a[1][1] = 30
a[1][2] = 40
```

[입력예시]

```
10 20 50 20 30 40
```



## 5) 다차원 배열 (예제: 평균)

---

- 학생 3명의 국어, 영어 성적이 있다.
- 이 자료를 저장하기 위한 2차원 배열을 선언하고 초기화 하시오.
- 국어와 영어과목의 평균을 각각 출력하시오.

	국어	영어
학생 A	20	100
학생 B	70	36
학생 C	30	50

## 5) 다차원 배열 (예제: 평균)

---

```
#include <stdio.h>
int main(void) {
    int i=0;
    int score[3][2]= { {20, 100}, {70, 36}, {30, 50} };
    int kor_sum=0, eng_sum=0;
    for(i=0; i<3; i++) {
        kor_sum = kor_sum + score[i][0];
        eng_sum = eng_sum + score[i][1];
    }
    printf("국어 = %f, 영어 = %f\n", kor_sum/3.0, eng_sum/3.0);
    return 0;
}
```

[실행결과]

국어 = 40.000000, 영어 = 62.000000

## 5) 다차원 배열

### ▪ 초기화 예제(1)

```
#include <stdio.h>
int main(void){
    int a[5][2]={ {78, 89}, {93, 100}, {20, 30}, {44, 55}, {88, 12}};
    return 0;
}
```

학생 0번	a[0][0]=78	a[0][1]=89
학생 1번	a[1][0]=93	a[1][1]=78
학생 2번	a[2][0]=20	a[2][1]=78
학생 3번	a[3][0]=44	a[3][1]=78
학생 4번	a[4][0]=88	a[4][1]=78

## 5) 다차원 배열

### ▪ 초기화 예제(2)

```
#include <stdio.h>
int main(void){
    int a[2][3]={ {10,20,50},{20,30,40}};
    return 0;
}
```

a[0][0]=10	a[0][1]=20	a[0][2]=50
a[1][0]=20	a[1][1]=30	a[1][2]=40

### ▪ 초기화 예제(3)    0으로 초기화

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

## 5) 다차원 배열

- 초기화 할 때 배열의 크기를 비워둘 수 있을까?
  - 컴파일러 입장에서 크기가 모호하므로 허용하지 않음

```
#include <stdio.h>
int main(void){
    int score[][]={{20,100},{70,35},{30,70},{80,80},{90,25}};
    return 0;
}
```

[컴파일 오류 발생]

error C2087: 'score': 첨자가 없습니다.

error C2078: 이니셜라이저가 너무 많습니다.

## 5) 다차원 배열

---

- 맨 처음 인덱스만 비울 수 있다

```
#include <stdio.h>
int main(void){
    int score[ ][2]={{20,100},{70,35},{30,70},{80,80},{90,25}};
    return 0;
}
```

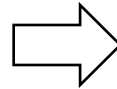
## 5) 다차원 배열 (실습7)

---

- 사용자에게서 총 9개의 숫자를 입력 받아, 3 X 3 배열을 초기화 하시오.
- 3 X 3 표 형태로 출력하시오.

입력 예시1

9 8 7 6 5 4 3 2 1



출력 예시1

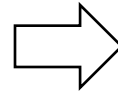
9	8	7
6	5	4
3	2	1

## 5) 다차원 배열 (실습8)

- 1) 3 X 3 배열을 선언하고 모두 0으로 초기화 하시오
- 2) 배열의 값을 3 X 3 표 형태로 출력하시오.
- 3) 사용자로부터 값을 바꾸고 싶은 **행**과 **열** 그리고 새로운 **값**을 받는다.
- 4) 사용자가 입력한 정보를 토대로 배열의 값을 바꾸시오.
- 5) 배열의 값을 3 X 3 표 형태로 한번 더 출력하시오.

입력 예시1

1 2 7



출력 예시1

0	0	0
0	0	0
0	0	0
0	7	0
0	0	0
0	0	0



## 5) 다차원 배열

---

- 반별 학생 3명의 국어, 영어 성적이 있다.

1반		국어	영어
	학생 A	20	100
	학생 B	70	36
	학생 C	30	50

2반		국어	영어
	학생 A	30	100
	학생 B	80	40
	학생 C	40	60

## 5) 다차원 배열

### ▪ 두 반의 국어 성적만 출력하기

```
#include <stdio.h>
int main(void){
    int a[2][3][2]={{{20,100}, {70,36},{30,50}},
                    {{30,100}, {80,40},{40,60}}};
    int i=0, j=0, k=0;
    for(i=0;i<2;i++){
        printf("%d반 국어-----\n", i+1);
        for(j=0;j<3;j++) {
            printf("a[%d][%d][%d] = %d \n",
                i, j, 0, a[i][j][0]);
        }
    }
    return 0;
}
```

1반 국어-----

a[0][0][0] = 20

a[0][1][0] = 70

a[0][2][0] = 30

2반 국어-----

a[1][0][0] = 30

a[1][1][0] = 80

a[1][2][0] = 40