

# 프로그래밍을 통한 논리적 사유 연습





# 10장. 배열

(정의, 1차원)

# 배열의 정의

- 배열 : 동일한 종류, 동일한 자료형의 데이터를 메모리에 일련의 구조화된 집단으로 구성한 것

- 정수형 변수 선언 예

배열을 안 쓴다면...

- 수학, 국어, 영어, 과학, 사회(5과목) 성적 입력

```
int jumsu1, jumsu2, jumsu3, jumsu4, jumsu5;
```

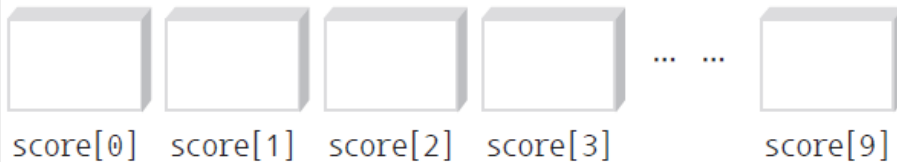
- 직원수가 10,000명인 회사에서 평가시험 점수

```
int score1, score2, score3, score4, score5;  
int score6, score7, score8, score9, score10;  
.....  
int score9996, score9997, score9998, score9999, score10000;
```

# 배열의 정의

- 배열 선언

```
자료형 배열명[배열 크기];  
int score[10];
```



배열 선언

```
score 배열의 첫 번째 원소 : score[0]  
score 배열의 두 번째 원소 : score[1]  
score 배열의 세 번째 원소 : score[2]  
.....  
score 배열의 열 번째 원소 : score[9]
```

- int형 변수 1개의 메모리 크기가 4바이트 -> score[10]의 메모리 크기는 40바이트.
- 배열의 각 요소는 배열 명에 원소 번호를 붙여서 명명.
- 배열의 첫 번째 원소는 0번부터 시작. ex) score[0]

# 배열의 초기화

- 배열을 먼저 선언하고 배열의 각 원소 값을 할당하는 방법

```
int a[3];    // 배열의 선언
a[0] = 10;   // 배열의 초기화
a[1] = 15;
a[2] = 20;
```

- 배열을 선언하면서 초기화하는 방법.

```
int a[3] = {10, 15, 20};    // 배열을 선언함과 동시에 초기화
```

```
int a[3] = {10, 15};        // 배열을 선언함과 동시에 초기화
```

a[0]=10  
a[1]=15  
a[2]= 0

- 배열을 선언함과 동시에 초기화할 때 배열의 원소보다 초기값을 적게 할당하면, 초기값이 할당되지 않는 배열의 원소에는 0이 저장됨.

# 배열의 초기화

- 배열의 원소보다 초기값을 많이 할당한 예 : 오류 발생

```
int a[3] = {10, 15, 20, 30}; // 틀린 예(오류 발생)
```

- 원소의 개수를 표시하지 않았지만, 그 개수만큼 배열의 원소가 자동 생성되는 예 - 배열의 원소를 표시하지 않고 초기값을 할당하는 것이 오류 발생을 줄이는 방법이기도 함.

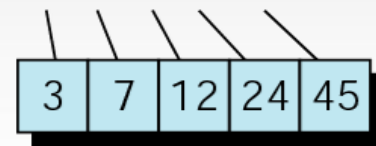
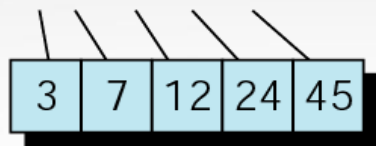
```
int a[ ] = {5, 25, 18, 6, 8};
```

- 배열 선언과 초기값 할당을 따로 하는 예 : 오류 발생
  - 원소의 개수를 반드시 기록해야 함.

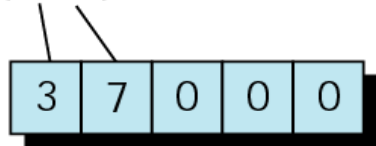
```
int a[ ];      // 오류 발생  
a[0] = 5;  
a[1] = 25;  
a[2] = 18;
```

# 배열의 초기화

`int numbers [5] = { 3,7,12,24,45 } ;`    `int numbers [ ] = { 3,7,12,24,45 } ;`

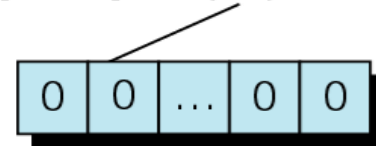


`int numbers [5] = { 3,7 } ;`



The rest are  
filled with 0s

`int lotsOfNumbers [1000] = { 0 } ;`



All filled with 0s



# 1차원 배열

- 1차원 배열 : 대괄호([ ]) 1개로 선언하는 배열
- 배열을 반복문과 함께 사용하면 유용.  
반복문의 증감 변수를 배열의 첨자와 동일하게 여기면 됨.
- 반복문의 초기값을 설정할 때, 배열의 원소는 0부터 시작함.

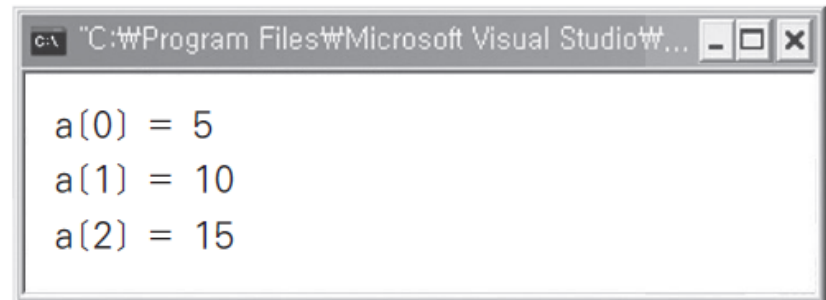
```
for(i=0; i<10; i++)  
    printf("score[%d] = %d\n", i, score[i]);
```



# 1차원 배열

예제) 배열을 초기화하고 원소를 출력

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i;
06     int a[ ] = {5, 10, 15};
07
08     for(i = 0; i < 3; i++)
09     {
10         printf("a[%d] = %d\n", i, a[i]);
11     }
12
13     return 0;
14 }
```



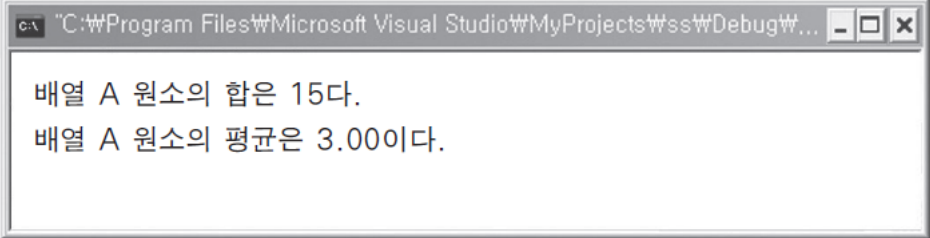
The screenshot shows a Windows command prompt window with the title bar "C:\ Program Files\Microsoft Visual Studio\...". The window contains the following output:

```
a[0] = 5
a[1] = 10
a[2] = 15
```

# 1차원 배열

예제) 배열 원소의 합과 평균을 구하는 예제

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int A[5] = {1, 2, 3, 4, 5};
06     int i, sum = 0;
07     double average;
08
09     for(i = 0; i < 5; i++)
10         sum += A[i];
11
12     average = (double)sum / i;
13
14     printf("배열 A 원소의 합은 %d다.\n", sum);
15     printf("배열 A 원소의 평균은 %4.21f이다.\n", average);
16
17     return 0;
18 }
```



배열 A 원소의 합은 15다.  
배열 A 원소의 평균은 3.00이다.