



# 프로그래밍을 통한 논리적 사유 연습





# 10장. 배열 (2차원, 다차원)

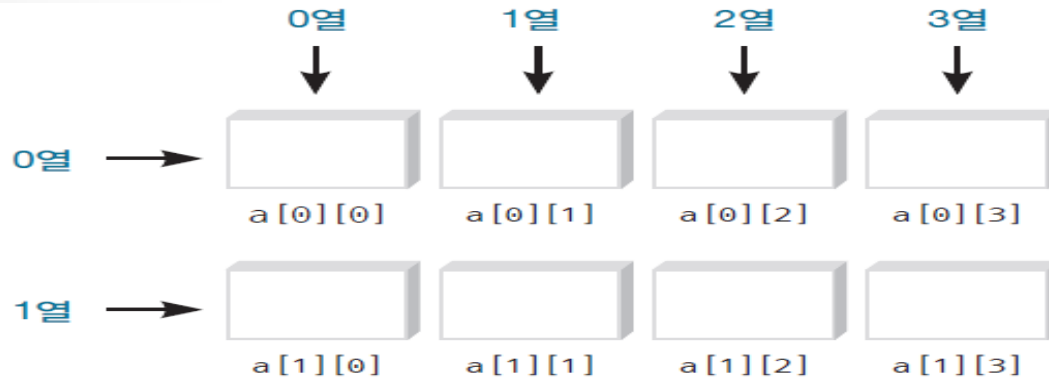
## 2차원 배열

- 연속적인 저장 공간을 선언하는 배열에서 1차원뿐만 아니라 2차원 이상도 가능.
- 2차원 배열의 형태 : 행과 열로 표현된 행렬

```
int a[2][4];
```



- 대괄호([ ]) 2개를 이용해서 정의함.
- a[2][4] : 행이 2개이고 열이 4개인 2차원 행렬



2차원 배열 선언

## 2차원 배열

First dimension  
(rows)

0				
1				
2				
3				
4				
	0	1	2	3

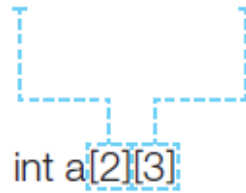
Second dimension  
(columns)

<table><tr><td></td><td></td><td></td><td></td></tr><tr><td>table [0][0]</td><td>table [0][1]</td><td>table [0][2]</td><td>table [0][3]</td></tr></table> table [0]					table [0][0]	table [0][1]	table [0][2]	table [0][3]	<table><tr><td></td><td></td><td></td><td></td></tr><tr><td>table [1][0]</td><td>table [1][1]</td><td>table [1][2]</td><td>table [1][3]</td></tr></table> table [1]					table [1][0]	table [1][1]	table [1][2]	table [1][3]	<table><tr><td></td><td></td><td></td><td></td></tr><tr><td>table [2][0]</td><td>table [2][1]</td><td>table [2][2]</td><td>table [2][3]</td></tr></table> table [2]					table [2][0]	table [2][1]	table [2][2]	table [2][3]	<table><tr><td></td><td></td><td></td><td></td></tr><tr><td>table [3][0]</td><td>table [3][1]</td><td>table [3][2]</td><td>table [3][3]</td></tr></table> table [3]					table [3][0]	table [3][1]	table [3][2]	table [3][3]	<table><tr><td></td><td></td><td></td><td></td></tr><tr><td>table [4][0]</td><td>table [4][1]</td><td>table [4][2]</td><td>table [4][3]</td></tr></table> table [4]					table [4][0]	table [4][1]	table [4][2]	table [4][3]
table [0][0]	table [0][1]	table [0][2]	table [0][3]																																									
table [1][0]	table [1][1]	table [1][2]	table [1][3]																																									
table [2][0]	table [2][1]	table [2][2]	table [2][3]																																									
table [3][0]	table [3][1]	table [3][2]	table [3][3]																																									
table [4][0]	table [4][1]	table [4][2]	table [4][3]																																									
<i>table</i>																																												

## 2차원 배열

- 2차원 배열 초기화 방법

세로줄 원소    가로줄 원소

  
int a[2][3]

```
a[0][0] = 10;  
a[0][1] = 20;  
a[0][2] = 30;  
a[1][0] = 40;  
a[1][1] = 50;  
a[1][2] = 60;
```

a[0][0]    a[0][1]    a[0][2]

10	20	30
----	----	----

40	50	60
----	----	----

a[1][0]    a[1][1]    a[1][2]

### 2차원 배열의 선언과 초기값 할당

- 열을 초기화시키는 방법
  - 배열을 선언한 후, 배열의 각 원소값을 할당
  - 선언과 동시에 초기화

## 2차원 배열

- 2차원 배열의 선언

```
int a [2][3] = {{10, 20, 30}, {40, 50, 60}};  
int a [2][3] = {10, 20, 30, 40, 50, 60};
```

- 중괄호({ })로 배열의 원소를 행 단위로 묶어 초기화하거나, 전체 원소를 묶어 초기화
- 2차원 배열을 선언할 때, 행의 개수는 생략 (O). 열의 개수는 생략(X)
- 올바른 예: `int a [ ][3] = {10, 20, 30, 40, 50, 60};`
- 틀린 예 1: `int a [2][ ] = {10, 20, 30, 40, 50, 60};`
- 틀린 예 2: `int a [ ][ ] = {10, 20, 30, 40, 50, 60};`
- 틀린 예 3: `int a [ ][3];`
- 올바른 예: `int a[ ][3] = {10, 20, 30, 40};`

## 2차원 배열

- 2차원 배열의 초기화
  - 행의 순서에 따라 2차원 배열이 순서대로 초기화

```
int num[2][3] = {1, 2, 3,  
                4, 5, 6};
```

1	2	3
4	5	6

```
int num[2][3] = {{1, 2},  
                {3, 4, 5}};
```

1	2	0
3	4	5

```
int num[2][3] = {1, 2, 3, 4, 5};  
int num[ ][3] = {1, 2, 3, 4, 5};
```

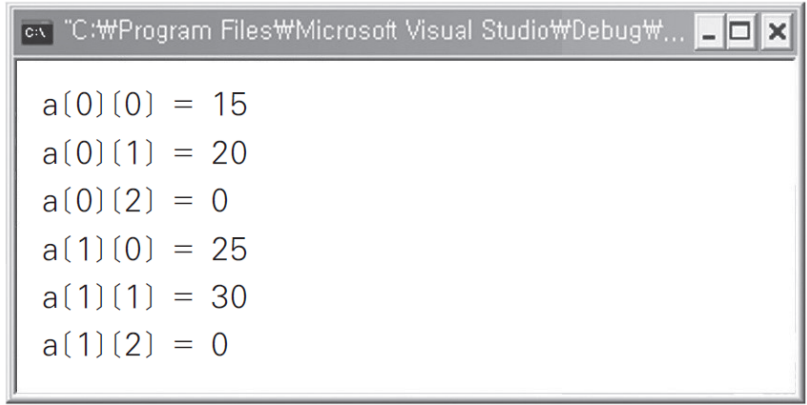
1	2	3
4	5	0



## 2차원 배열

예제) 2차원 배열의 초기화

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i, j;
06     int a[2][3] = {{15, 20}, {25, 30}};
07
08     for(i = 0; i < 2; i++)
09     {
10         for(j = 0; j < 3; j++)
11         {
12             printf("a[%d][%d] = %d\n", i, j, a[i][j]);
13         }
14     }
15
16     return 0;
17 }
```



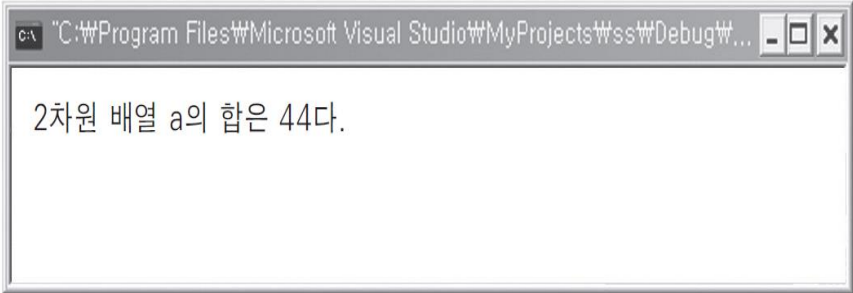
```
a[0][0] = 15
a[0][1] = 20
a[0][2] = 0
a[1][0] = 25
a[1][1] = 30
a[1][2] = 0
```



## 2차원 배열

예제) 2차원 배열의 합을 구하는 예제

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i, j, sum = 0;
06     int a[2][2] = {5, 20, 12, 7};
07
08     for(i = 0; i < 2; i++)
09     {
10         for(j = 0; j < 2; j++)
11         {
12             sum += a[i][j];
13         }
14     }
15
16     printf("2차원 배열 a의 합은 %d다.\n", sum);
17
18     return 0;
19 }
```



2차원 배열 a의 합은 44다.

## 2차원 배열

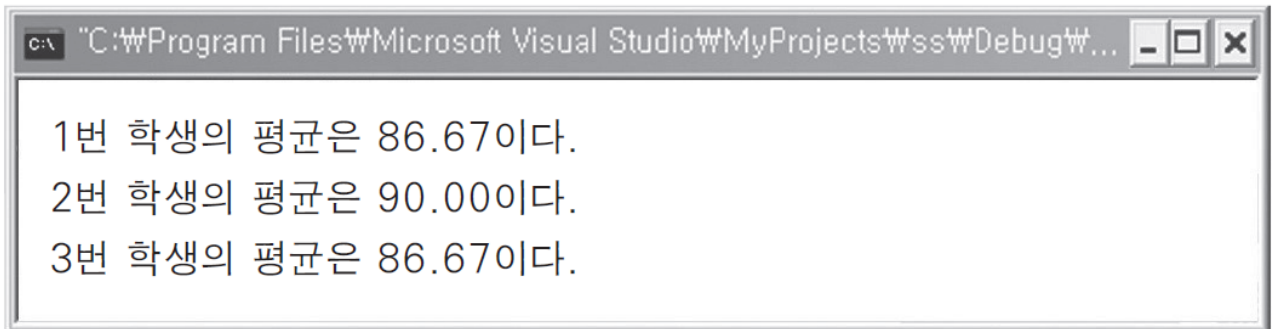
예제) 학생 별 평균을 구하는 프로그램

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i, j, sum = 0;
06     int a[ ][3] = {{90, 80, 90}, // 영희의 국어, 영어, 수학 점수
07                   {70, 100, 100}, // 철수의 국어, 영어, 수학 점수
08                   {80, 90, 90}}; // 미나의 국어, 영어, 수학 점수
09
10     double average[3];
11
12     for(i = 0; i < 3; i++)
13     {
14         for(j = 0; j < 3; j++)
15         {
16             sum += a[i][j];
17         }
18     }
```

## 2차원 배열

예제) 학생 별 평균을 구하는 프로그램

```
19     average[i] = (double)sum / j;  
20     printf("%d번 학생의 평균은 %.21f이다.\n", i+1,average[i]);  
21  
22     // 다음 학생의 평균을 구하기 위해 sum을 초기화한다.  
23     sum = 0;  
24 }  
25  
26 return 0;  
27 }
```



A screenshot of a Windows command prompt window. The title bar shows the path "C:\Program Files\Microsoft Visual Studio\MyProjects\ss\Debug\...". The window contains three lines of text, which are the outputs of the program: "1번 학생의 평균은 86.67이다.", "2번 학생의 평균은 90.00이다.", and "3번 학생의 평균은 86.67이다.".

```
c:\ "C:\Program Files\Microsoft Visual Studio\MyProjects\ss\Debug\...  
1번 학생의 평균은 86.67이다.  
2번 학생의 평균은 90.00이다.  
3번 학생의 평균은 86.67이다.
```

## 2차원 배열

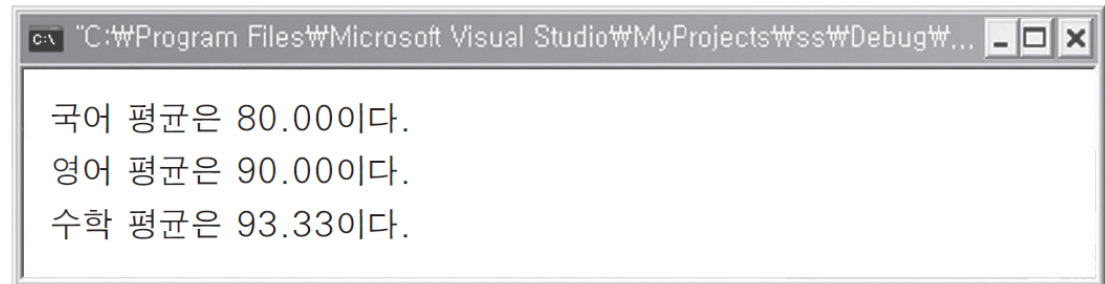
예제) 국어, 영어, 수학의 평균을 구하는 예제

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i, j, sum = 0;
06     int a[ ][3] = {{90, 80, 90}, // 영희의 국어, 영어, 수학 점수
07                   {70, 100, 100}, // 철수의 국어, 영어, 수학 점수
08                   {80, 90, 90}}; // 미나의 국어, 영어, 수학 점수
09
10     double aveClass[3];
11
12     for(j = 0; j < 3; j++)
13     {
14         for(i = 0; i < 3; i++)
15         {
16             sum += a[i][j];
17         }
18
19         aveClass[j] = (double)sum / i;
20     }
```

## 2차원 배열

예제) 국어, 영어, 수학의 평균을 구하는 예제

```
21     if(j == 0)
22         printf("국어 평균은 %.21f이다.\n", aveClass[j]);
23     else if(j == 1)
24         printf("영어 평균은 %.21f이다.\n", aveClass[j]);
25     else if(j == 2)
26         printf("수학 평균은 %.21f이다.\n", aveClass[j]);
27
28     // 다음 과목의 평균을 구하기 위해서 sum을 초기화한다.
29     sum = 0;
30 }
31
32 return 0;
33 }
```



A screenshot of a Windows command prompt window. The title bar shows the path "C:\Program Files\Microsoft Visual Studio\MyProjects\ss\Debug\...". The window contains the following text output:

```
국어 평균은 80.00이다.
영어 평균은 90.00이다.
수학 평균은 93.33이다.
```

# 다차원 배열

- 다차원 배열

```
int a[2][2][2];  
int b[2][2][2][2];
```

- 2차원 배열 -> 2차원 영상
- 3차원 배열 -> 3차원 영상

```
int a[256][256];
```

y ↓	0	1	2	...	254	255 ← x
0	33	43	43	...	63	60
1	32	42	43	...	62	60
2	40	44	44	...	60	63
3	42	39	39	...	50	51
⋮	...	...	...	...	...	...
255	72	67	72	...	88	80

2차원 영상의 좌표와 화소값

# 다차원 배열

예제) 3차원 배열을 초기화하고 각 원소를 출력

```
01 #include <stdio.h>
02 int main(void)
03 {
04     int x[2][2][5] = {0,1,2,3,4,
05                       5,6,7,8,9,
06                       10,11,12,13,14,
07                       15,16,17,18,19};
08     int i, j, k;
09
10     for(i = 0; i < 2; i++)
11         for(j = 0; j < 2; j++)
12             for(k = 0; k < 5; k++)
13                 printf("x[%d][%d][%d] = %d\n", i, j, k, x[i][j][k]);
14
15     return 0;
16 }
```



# 다차원 배열

예제) 3차원 배열을 초기화하고 각 원소를 출력

```
C:\ "C:\Program Files\Microsoft Visual Studio\MyProjects\ss\Debug\... - □ ×  
x[0][0][0] = 0  
x[0][0][1] = 1  
x[0][0][2] = 2  
x[0][0][3] = 3  
x[0][0][4] = 4  
x[0][1][0] = 5  
x[0][1][1] = 6  
x[0][1][2] = 7  
x[0][1][3] = 8  
x[0][1][4] = 9  
x[1][0][0] = 10  
x[1][0][1] = 11  
x[1][0][2] = 12  
x[1][0][3] = 13  
x[1][0][4] = 14  
x[1][1][0] = 15  
x[1][1][1] = 16  
x[1][1][2] = 17  
x[1][1][3] = 18  
x[1][1][4] = 19
```