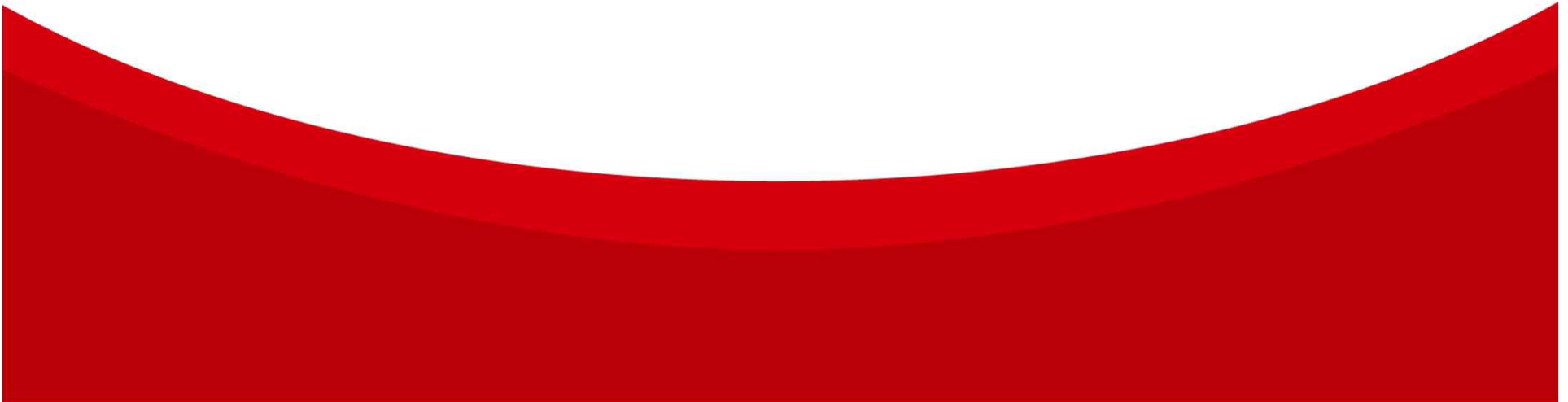


Chapter 10

배열 2



01

Question

1부터 10까지의 정수를 입력받다가 입력된 정수가 범위를 벗어나면 그 때까지 1번 이상 입력된 각 숫자의 개수를 작은 수부터 출력하는 프로그램을 작성하시오.

입력 예 5 6 10 9 3 5 3 5 0

출력 예 3 : 2개

5 : 3개

6 : 1개

9 : 1개

10 : 1개

```
#include <stdio.h>

int main()
{
    int num, i;
    int count[11] = {0};

    while (1) {
        scanf("%d", &num);
        if (num < 1 || num > 10) break;
        count[num]++;
    }

    for (i = 1 ; i <= 10; i++) {
        if (count[i] > 0) {
            printf("%d : %d개\n", i, count[i]);
        }
    }

    return 0;
}
```

설명

- `int count[11] = {0};`
각 숫자의 개수를 저장하기 위해 배열을 선언하고 0으로 초기화한다.
`count[1]`에는 1의 개수, `count[2]`에는 2의 개수, ..., `count[10]`에는 10의 개수를 저장하기 위해 만든 배열이다. 1부터 10까지의 숫자의 개수를 필요로 하므로 11개 선언을 한 것이다. (만약 `count[10]`으로 선언하면 배열은 `count[0]`부터 `count[9]`까지 만들어지므로 `count[10]`을 사용할 수 없다.)
- 배열을 초기화 할 때 선언한 개수보다 적게 넣어주면 배열의 0번부터 차례로 대입이 되다가 나머지는 모두 0으로 초기화된다.
- 초기화 하지 않으면 Random (임의의) 값이 들어감



초기화 하는 이유는?

- while (1) {
 scanf("%d", &num);
 if (num < 1 || num > 10) break;
 count[num]++;
}

숫자를 입력받아서 그 값이 범위를 벗어나면 입력 작업을 종료한다.
그렇지 않으면 배열에서 입력받은 숫자에 해당하는 위치의 값을 증가시킨다. (만약 3이 입력되면 count[3]을 1 증가시킨다.)

- if (count[i] > 0) {
 printf("%d : %d개\n", i, count[i]);
}

i번에 해당하는 배열의 위치의 값이 0보다 크면 그 개수를 출력한다. (만약 count[3]의 값이 2라면 3이 두 번 입력된 것이므로 3이 2개라고 출력한다.)

자가진단 1

영문 대문자를 입력받다가 대문자 이외의 문자가 입력되면 그 때까지 1번 이상 입력된 각 문자의 개수를 사전 순으로 출력하는 프로그램을 작성하시오.

입력 예 A B C F F F B Z !

출력 예 A : 1

 B : 2

 C : 1

 F : 3

 Z : 1

• Hint

'A'는 0번, 'B'는 1번, 'C'는 2번, ..., 'Z'는 25번에 count 한다. 'A'는 메모리에 65로 저장된다.

따라서 입력받은 문자에서 65 또는 'A'를 빼면 해당문자의 개수를 저장할 번호가 된다. ('B' - 65 = 1 또는 'B' - 'A' = 1)

1을 다시 'B'로 바꾸어 출력하려면 (char)(1+65) 또는 (char)(1+'A')를 출력하면 된다.

02

Question

정수를 입력받다가 0이 입력되면 마지막에 입력된 0을 제외하고 그 때까지 입력된 정수의 일의 자리 숫자가 각각 몇 개인지 작은 수부터 출력하는 프로그램을 작성하시오 (0개인 숫자는 출력하지 않는다.)

입력 예 10 55 123 63 85 61 125 0

출력 예 0 : 1개

1 : 1개

3 : 2개

5 : 3개

소스

```
#include <stdio.h>

int main()
{
    int num, i;
    int count[10]={0};

    while (1) {
        scanf("%d", &num);
        if (num == 0) break;
        count[num % 10]++;
    }

    for (i = 0; i < 10; i++) {
        if (count[i] > 0) {
            printf("%d : %d개\n", i, count[i]);
        }
    }

    return 0;
}
```


설명

- `int count[10]={0};`
일의 자리에 해당하는 개수를 저장하기 위해 배열을 선언하고 0으로 초기화한다.
일의 자리는 0부터 9까지이므로 `count[11]`로 하지 않아도 된다.
- `count[num % 10]++;`
일의 자리에 해당하는 배열의 위치를 1 증가시킨다.
(만약 35가 입력되면 `count[35 % 10]` 즉 `count[5]`를 증가시킨다.)

자가진단 2

100 미만의 정수를 입력받다가 0이 입력되면 마지막에 입력된 0을 제외하고 그 때까지 입력된 정수의 십의 자리 숫자가 각각 몇 개인지 작은 수부터 출력하는 프로그램을 작성하시오 (0개인 숫자는 출력하지 않는다.)

입력 예 10 55 3 63 85 61 85 0

출력 예 0 : 1

1 : 1

5 : 1

6 : 2

8 : 2

• Hint

10의 자리 숫자는 10으로 나눈 몫이다.

03

Question

피보나치 수열이란 첫 번째 항과 두 번째 항만 1이고 나머지 항은 바로 앞의 두 항의 합과 같은 수열을 말한다. 각 항의 값을 계산하는 다음의 예를 보고 피보나치 수열의 10, 20, 30, 40항의 값을 출력하는 프로그램을 작성하시오.

항	1	2	3	4	5	...
계산식			$1 + 1$	$1 + 2$	$2 + 3$...
수열	1	1	2	3	5	...

출력 예

피보나치 수열 10항 : 55
 피보나치 수열 20항 : 6765
 피보나치 수열 30항 : 832040
 피보나치 수열 40항 : 102334155

소스

```
#include <stdio.h>

int main()
{
    int i;
    int fibo[41]={0,1,1};

    for (i = 3; i <= 40; i++) {
        fibo[i] = fibo[i - 1] + fibo[i - 2];
    }

    for (i = 10; i <= 40; i += 10) {
        printf("피보나치 수열 %d항 : %d\n", i, fibo[i]);
    }

    return 0;
}
```

설명

$$F_0 = 0, F_1 = 1, F_{n+2} = F_{n+1} + F_n$$

- `int fibo[41]={0, 1, 1};`
피보나치 수열의 40번째 항까지 저장하기 위해 배열을 선언하고 1항과 2항을 1로 초기화하고 나머지는 모두 0으로 초기화한다. 배열의 첫 번째는 `fibo[0]`이므로 `fibo[1]`과 `fibo[2]`를 초기화하기 위해서는 반드시 첫 번째 `fibo[0]`에 해당하는 값을 먼저 넣어야 한다.
- `fibo[i] = fibo[i - 1] + fibo[i - 2];`
`i`항을 채우기 위해 바로 전항과 전전항의 합을 구하여 채운다. (3항이라면 `fibo[3] = fibo[2] + fibo[1]`이므로 2가 저장된다.)
- `for (i = 10; i <= 40; i += 10) {`
 `printf("피보나치 수열 %d항 : %d\\n", i, fibo[i]);`
 `}`
10항부터 10씩 증가하여 40항까지 배열에 저장된 값을 출력한다

자가진단 3

100 이하의 자연수를 입력받아 첫 번째 항은 100으로 두 번째 항은 입력받은 수로 초기화하고 다음 항부터는 전전항에서 전항을 뺀 수로 채워나가는 수열을 작성하여 그 수가 음수가 나올 때까지 출력하는 프로그램을 작성하시오

입력 예 62

출력 예 100 62 38 24 14 10 4 6 -2

04

Question

3행 3열의 배열을 선언하고 아래의 출력 예와 같이 초기화한 후 출력하는 프로그램을 작성하시오.

출력 예 3 5 4
 2 6 7
 8 10 1

소스

```
#include <stdio.h>

int main()
{
    int i, j;
    int arr[3][3] = {{3, 5, 4}, {2, 6, 7}, {8, 10, 1}};

    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            printf("%d ", arr[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```


설명

- `int arr[3][3] = {{3, 5, 4}, {2, 6, 7}, {8, 10, 1}};`
3 * 3의 2차원 배열을 선언하고 값을 초기화한다.
- 위와 같이 선언하면 다음과 같은 배열이 생성된다.

	0열	1열	2열
0행	arr[0][0] 3	arr[0][1] 5	arr[0][2] 4
1행	arr[1][0] 2	arr[1][1] 6	arr[1][2] 7
2행	arr[2][0] 8	arr[2][1] 10	arr[2][2] 1

2차원 배열을 행렬로 만들면 배열의 첫 번째 첨자는 행(줄)을 나타내고 두 번째 첨자는 열(칸)을 나타내게 된다.

2차원 배열을 아파트라고 생각하면 첫 번째 첨자는 동 번호, 두 번째 첨자는 호수를 구분해 주는 것과 같다. 즉 `arr[2][1]`은 아파트로 치면 2동 1호를 나타내고 행렬에서는 2행 1열이 된다.

```
• for (i = 0; i < 3; i++) {  
    for (j = 0; j < 3; j++) {  
        printf("%d ", arr[i][j]);  
    }  
    printf("\n");  
}
```

0행 0열부터 2행 2열까지 배열의 값을 차례로 출력한다.

큰 for문의 `i`는 행 번호를 나타내고 각 행마다 열을 나타내는 `j`가 0부터 2까지 실행되면서 `i`행 `j`열의 값을 출력한다. `i`행의 모든 열이 출력이 되면 줄을 바꾸고 다시 행 번호를 바꾸어 반복 실행한다.

자가진단 4

3행 5열의 배열을 아래 출력 예와 같이 초기화 한 후 형식에 맞추어 출력하는 프로그램을 작성하시오 (수 사이의 공백은 3칸으로 하고 각 수는 2칸으로 오른쪽에 맞추어 출력한다. "%2d ")

출력 예

5	8	10	6	4
11	20	1	13	2
7	9	14	22	3

05

Question

3행 3열의 배열 두 개를 만들어서 입력을 받고 두 배열의 합을 구하여 출력하는 프로그램을 작성하시오.

입력 예

첫 번째 배열 1행 1 2 3

첫 번째 배열 2행 4 5 6

첫 번째 배열 3행 7 8 9

두 번째 배열 1행 9 6 3

두 번째 배열 2행 8 5 2

두 번째 배열 3행 7 4 1

출력 예

10 8 6

12 10 8

14 12 10

Chapter 10

배열 2

소스

```
#include <stdio.h>

int main()
{
    int i, j;
    int a[3][3], b[3][3], hap[3][3];

    for (i = 0; i < 3; i++) {
        printf("첫번째 배열 %d행 ", i + 1);
        for (j = 0; j < 3; j++) {
            scanf("%d", &a[i][j]);
        }
    }

    for (i = 0; i < 3; i++) {
        printf("두번째 배열 %d행 ", i + 1);
        for (j = 0; j < 3; j++) {
            scanf("%d", &b[i][j]);
        }
    }

    for (i = 0; i < 3; i++) {
```

Chapter 10

배열 2

```
        for (j = 0; j < 3; j++) {
            hap[i][j] = a[i][j] + b[i][j];
        }
    }

    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            printf("%d ", hap[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

설명

- ```
for (i = 0; i < 3; i++) {
 printf("첫번째 배열 %d행 ", i + 1);
 for (j = 0; j < 3; j++) {
 scanf("%d", &a[i][j]);
 }
}
```

입력받을 행의 정보(i+1)를 출력하고 각 열의 값을 입력받는다.  
a[0][0]부터 a[2][2]까지 차례로 입력받게 된다.

- ```
for (i = 0; i < 3; i++) {  
    for (j = 0; j < 3; j++) {  
        hap[i][j] = a[i][j] + b[i][j];  
    }  
}
```

입력받은 두 개의 배열 a와 b의 같은 위치의 값을 합하여 hap배열의 해당위치에 저장한다.

hap[0][0] = a[0][0] + b[0][0]부터 hap[2][2] = a[2][2] + b[2][2]까지 실행된다.

06

Question

학생 3명의 국어, 영어, 수학 점수를 입력받아 학생별 총점과 과목별 합계를 아래와 같이 출력하는 프로그램을 작성하시오.

입력 예 1번째 학생의 점수 95 100 88

2번째 학생의 점수 85 60 93

3번째 학생의 점수 66 99 88

출력 예 국어 영어 수학 총점

1번 95 100 88 283

2번 85 60 93 238

3번 66 99 88 253

합계 246 259 269 774

Chapter 10

배열 2

소스

```
#include <stdio.h>

int main()
{
    int i, j;
    int score[4][4] = {0};

    for (i = 0; i < 3; i++) {
        printf("%d번째 학생의 점수 ", i + 1);
        for (j=0 ; j<3 ; j++) {
            scanf("%d", &score[i][j]);
            score[i][3] += score[i][j];
            score[3][j] += score[i][j];
            score[3][3] += score[i][j];
        }
    }

    printf("    국어 영어 수학 총점\n");
    for (i = 0; i < 4; i++) {
        if (i < 3) printf(" %d번", i + 1);
        else printf("합계");
        for (j = 0; j < 4; j++) {
            printf(" %4d", score[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

설명

- `int score[4][4] = {0};`

3명씩 3과목의 점수와 각각의 합계를 저장할 배열을 선언하고 0으로 초기화한다. 앞의 첨자는 0부터 2까지는 학생의 번호를 나타내고 3번에는 합계를 저장하기 위해 4개를 선언했다. 뒤의 첨자는 0부터 2까지는 각각 과목(국어, 영어, 수학)을 나타내고 마지막 3번에는 총점을 저장하기 위해 4개를 선언했다.

입력을 받는 변수나 배열은 초기화가 필요없지만 합계와 총점은 입력받는 것이 아니고 점수를 누적하여 더해 나갈 것이므로 처음에 0이 되어야 한다. 따라서 전체를 0으로 초기화 한 것이다. 배열에 저장되는 내용을 표로 나타내면 다음과 같다.

행 \ 열	0 (국어)	1 (영어)	2 (수학)	3 (합계)
0 (1번학생)				
1 (2번학생)				2번학생 총점
2 (3번학생)		3번학생 영어점수		
3 (합계)	국어점수 합계			전체합계

score[2][1]이라고 하면 세 번째 학생의 두 번째 과목(영어) 점수를 나타내는 것이고
score[1][3]은 두 번째 학생의 총점
score[3][0]은 첫 번째 과목(국어)의 합계
score[3][3]은 전체학생 총점의 합계를 나타내는 것이다.

- for (j = 0 ; j < 3; j++) {
 scanf("%d", &score[i][j]);
 score[i][3] += score[i][j];
 score[3][j] += score[i][j];
 score[3][3] += score[i][j];
}

i행에서 j가 0부터 2까지 바뀌면서 3과목의 점수를 입력받는다.
입력받은 점수를 i번 학생의 총점 score[i][3]에 누적하여 더한다.
" j번 과목의 합계 score[3][j] "
" 전체 학생 총점의 합계 score[3][3] "

```
• for (i = 0; i < 4; i++) {  
    if (i < 3) printf(" %d번 ", i + 1);  
    else printf("합계 ");  
    for (j = 0; j < 4; j++) {  
        printf("%4d ", score[i][j]);  
    }  
    printf("\n");  
}
```

i가 0이면 "1번", 1이면 "2번", 2이면 "3번"이라고 출력하고 i가 3일 경우는 "합계"라고 출력한 후 각각의 점수를 차례로 출력한다.

자가진단 6

5명 학생의 국어, 영어, 수학, 과학 4과목의 점수를 입력 받아서 각 개인별로 평균이 80 이상이면 “pass”, 그렇지 않으면 “fail”을 출력하고, 합격한 사람의 수를 출력하는 프로그램을 작성하시오.

입력 예

```
85 67 95 65
80 95 86 56
100 98 67 86
95 76 84 65
67 86 90 76
```

출력 예

```
fail
fail
pass
pass
fail
Successful : 2
```

07

Question

아래와 같은 파스칼 삼각형을 출력하는 프로그램을 작성하시오. 파스칼 삼각형은 첫 번째 줄 첫 번째 행을 1로 초기화한 후 다음 줄부터는 바로 위의 값과 바로 위 왼쪽의 값을 더한 값이 된다.

출력 예

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

Chapter 10

배열 2

소스

```
#include <stdio.h>

int main()
{
    int i, j;

    int pas[6][6] = {0};

    pas[1][1] = 1;
    for (i = 2; i < 6; i++) {
        for (j = 1; j <= i; j++) {
            pas[i][j] = pas[i - 1][j - 1] + pas[i - 1][j];
        }
    }

    for (i = 1; i < 6; i++) {
        for (j = 1; j <= i; j++) {
            printf("%d ", pas[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

설명

- `int pas[6][6] = {0};`
`pas[1][1] = 1;`

행렬을 구성하여 모든 값을 0으로 초기화하고 1행 1열을 1로 초기화한다.

`pas[0][0]`부터 사용해도 되지만 위와 같이 하는 것이 훨씬 편리하다. 그 이유는 아래에서 설명하기로 하고 일단 `pas[5][5]`까지 필요하므로 $6 * 6$ 의 행렬을 구성한 것이다.

- ```
for (i = 2; i < 6; i++) {
 for (j = 1; j <= i; j++) {
 pas[i][j] = pas[i - 1][j - 1] + pas[i - 1][j];
 }
}
```

1행은 이미 초기화되었으므로 행을 나타내는 `i`가 2행부터 5행까지 차례로 실행된다. 행이 정해지면 열을 나타내는 `j`가 1부터 `i`번까지 실행되면서 블록 내의 계산식을 실행한다. `i`행 `j`열의 값은 바로 위의 왼쪽 `i - 1`행 `j - 1`열의 값과, 바로 위 `i - 1`행 `j`열의 값을 더한 값이 된다.



## Chapter 10

### 배열 2

```

• for (i = 2; i < 6; i++) {
 for (j = 1; j <= i; j++) {
 pas[i][j] = pas[i - 1][j - 1] + pas[i - 1][j];
 }
}

```

• i=2

j=1 : pas[2][1] = pas[1][0] + pas [1][1];

j=2 : pas[2][2] = pas[1][1] + pas [1][2];

• i=3

j=1 : pas[3][1] = pas[2][0] + pas [2][1];

j=2 : pas[3][2] = pas[2][1] + pas [2][2];

j=3 : pas[3][3] = pas[2][2] + pas [2][3];

|    | 0열 | 1열 | 2열 | 3열 | 4열 |
|----|----|----|----|----|----|
| 0행 | 0  | 0  | 0  | 0  | 0  |
| 1행 | 0  | 1  | 0  | 0  | 0  |
| 2행 | 0  | 0  | 0  | 0  | 0  |
| 3행 | 0  | 0  | 0  | 0  | 0  |
| 4행 | 0  | 0  | 0  | 0  | 0  |

|    | 0열 | 1열 | 2열 | 3열 | 4열 |
|----|----|----|----|----|----|
| 0행 | 0  | 0  | 0  | 0  | 0  |
| 1행 | 0  | 1  | 0  | 0  | 0  |
| 2행 | 0  | 1  | 1  | 0  | 0  |
| 3행 | 0  | 0  | 0  | 0  | 0  |
| 4행 | 0  | 0  | 0  | 0  | 0  |

|    | 0열 | 1열 | 2열 | 3열 | 4열 |
|----|----|----|----|----|----|
| 0행 | 0  | 0  | 0  | 0  | 0  |
| 1행 | 0  | 1  | 0  | 0  | 0  |
| 2행 | 0  | 1  | 1  | 0  | 0  |
| 3행 | 0  | 1  | 2  | 1  | 0  |
| 4행 | 0  | 0  | 0  | 0  | 0  |

2행 1열은  $\text{pas}[2][1] = \text{pas}[1][0] + \text{pas}[1][1]$ 이 된다. 만약 위 파스칼 삼각형을  $\text{pas}[0][0]$ 을 1로 초기화하고 시작하였다면 어떻게 되었을까? 2행 1열은  $\text{pas}[1][0]$ 이 되고 계산식은  $\text{pas}[0][-1] + \text{pas}[0][0]$ 이 될 것이다. 첨자가 음수로 되거나 선언된 배열의 위치를 벗어나게 되면 메모리를 침범하는 심각한 에러가 발생하게 된다. 따라서 위와 같이 하려면 반드시 범위를 벗어나는지 확인하기 위해 조건문을 써야 한다. 그러한 복잡한 조건문을 쓰지 않고 간단하게 처리하기 위해서 0을 비워두고 1부터 사용하게 된 것이다.

## 자가진단 7

배열을 만들어서 반복문을 이용하여 아래와 같이 저장한 후 출력하는 프로그램을 작성하시오.

출력 예

```
1 1 1 1 1
1 2 3 4 5
1 3 6 10 15
1 4 10 20 35
1 5 15 35 70
```

• Hint

바로 위의 값과 바로 왼쪽의 값을 더한 것이다.

배열[i][1] 또는 배열[1][i]을 1로 초기화하고 배열[1][1]부터 배열[5][5]까지 식을 세워 구하면 된다.

첫 번째 행을 모두 1로 초기화하고 두 번째 행부터 채워나가도 된다.

# Thank You!!!

