



실 습 문 제

제10장

1. [배열의 선언 및 사용] 학생들의 성적을 저장하고 평균을 구해보자.

```
#include <stdio.h>

int main(void)
{
    int grade1, grade2, grade3;
    int average, i;

    printf("3명의 성적을 입력하시오\n");
    scanf("%d", &grade1);
    scanf("%d", &grade2);
    scanf("%d", &grade3);

    average = grade1 + grade2 + grade3;

    printf("평균 = %d\n", average);
    return 0;
}
```

- (a) 위의 프로그램을 컴파일하여 실행하고 그 결과를 기록하라.

실행결과



- (b) 5명의 성적을 처리할 수 있도록 변경하라.
 (c) 위의 프로그램을 배열 `grade[]`를 사용하여 평균을 구하도록 변경하라. 이 때 배열의 크기는 `SIZE` 라는 기호 상수로 정의하여 사용하라.
 (d) 배열을 사용한 프로그램에서 10명의 성적을 처리하도록 변경하여 보라. 일반 변수를 사용했을 때와 비교하라
 (e) 다음과 같은 10점 단위의 점수대 분포표도 출력하도록 코드를 추가하라. 분포표도 배열로 구현하라.

점수대	인원수
0-10	1
11-20	3
...	

2. [배열의 초기화]

```
#include <stdio.h>
#define SIZE 3

int main(void)
{
    int i;
    int grade[SIZE] = { 10, 20, 30 };    // ①

    for(i = 0; i < SIZE; i++)
        printf("%d\n", grade[i]);

    return 0;
}
```

(a) 위의 프로그램을 컴파일하여 실행하고 그 결과를 기록하라.
실행결과

- (b) 배열의 크기보다 더 많은 초기화 값을 주면 어떻게 되는가? SIZE는 3으로 유지한 채 초기화 값을 10, 20, 30, 40으로 주어보라.
- (c) 배열의 크기보다 더 적은 초기화 값을 주면 어떻게 되는가? SIZE는 3으로 유지한 채 초기화 값을 10, 20으로 주어보라. 초기화 되지 않은 배열 원소에는 어떤 값이 저장되어 있는가?
- (d) 초기값만 주어지고 배열의 크기가 지정되지 않으면 어떻게 되는지 알기 위하여 SIZE를 삭제하고 컴파일, 실행하여 보라.

3. [다차원 배열] 학생들의 국어, 영어, 수학 성적을 받아서 각 과목별로 평균을 구하는 프로그램을 작성하여 보자. 여러 명의 학생들이 있고 학생마다 3과목의 성적을 저장하여야 하므로 2차원 배열을 사용하는 것이 바람직하다.

```
#include <stdio.h>
#define STUDENTS 5
#define SUBJECTS 3

int main(void)
{
    int grade[STUDENTS][SUBJECTS];
    int i, j;

    for(i = 0; i < STUDENTS; i++)
        for(j = 0; j < SUBJECTS; j++)
            scanf("%d", &grade[i][j]);
}
```



```
// ①

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

return 0;
}
```

(a) 위의 프로그램을 컴파일하여 실행하고 그 결과를 기록하라.

실행결과

- (b) 위의 프로그램은 단순히 각 학생들의 성적을 받아서 출력한다. 각 과목별로 평균을 구하여 출력하도록 ①의 위치에 적절한 문장들을 추가하여 보라.
- (c) 각각의 학생별로 평균을 구하여 출력하도록 ①의 위치에 적절한 문장들을 추가하여 보라. 즉 각 학생의 국어, 영어, 수학 성적의 평균을 구하여 출력한다.
- (d) 성적을 사용자로부터 받지 말고 배열의 초기값으로 입력하라.
- (e) 학생별로 평균 성적을 구할 때 가중치를 두어서 가중 평균을 계산하여 보라. 가중치는 별도의 배열 `weight[]`에 0.3, 0.4, 0.3과 같이 저장되어 있다고 가정하라.

4. [배열과 함수] 1번 문제를 다음과 같이 함수를 이용하여 다시 작성하여 보자.

```
#include <stdio.h>

int cal_average(int grade[]);

int main(void)
{
    int grade[] = { 10, 20, 30 };
    int average;

    average = cal_average(grade);
    printf("평균 = %d\n", average);

    return 0;
}

int cal_average(int grade[])
{
    // ①
}
```

(a) 위의 프로그램에서 `cal_average()` 내부를 적절한 문장(들)로 채운 후에 컴파일 하여 실행하고 그 결과를 기록하라.

실행결과

(b) 프로그램에서 성적을 `int`형이 아닌 `double`형으로 처리하도록 코드를 변경하여 보라.

(c) `int cal_average(int grade[5])`와 같이 함수를 정의할 때 매개 변수에서 배열의 크기를 정의하면 어떻게 되는가?

(d) 보통 배열을 인수로 받는 함수의 경우, 배열의 크기를 알아야 하기 때문에 배열의 크기를 별도의 인수로 전달하는 것이 좋다. `cal_average(int grade[], int n)`과 같이, `cal_average()` 함수의 두 번째 인수로 배열의 크기를 전달하도록 코드를 변경하여 보라.

(e) 3번 실습문제와 같이, 여러 과목에 대한 성적을 2차원 배열로 표현하고, 이것을 `cal_average()`에 전달하여, 모든 학생에 대한 전과목 평균을 계산하여 반환하여 보라.

(f) 성적 배열과 배열의 크기를 인수로 받아서, 최고 성적 값을 지닌 배열 원소의 첨자를 반환하는 함수를 작성하고, 이 함수를 테스트하시오.

```
int get_min_index(int grade[], int n);
```

5. [배열 인수] 정수형이나 실수형 인수는 변수값이 복사되어서 함수의 매개 변수로 전달되므로, 함수 안에서 매개 변수의 값을 바꾸더라도 원래의 변수의 값이 변경되지 않는다. 하지만 배열을 함수의 인수로 전달할 때는 복사되는 것이 아니라 원본이 전달된다. 따라서 매개 변수를 변경하면 원래의 배열도 변경된다.

```
#include <stdio.h>

#define SIZE 5
void get_grade(int s[]);
void print_grade(int s[]);

int main(void)
{
    int grade[SIZE] = { 0, 0, 0, 0, 0 };

    print_grade(grade);
    get_grade(grade);
    print_grade(grade);

    return 0;
}
```



```
// 사용자로부터 성적을 입력받아서 배열에 저장한다.
void get_grade(int grade[])
{
    // ①
}
// 배열 원소들을 화면에 출력한다.
void print_grade(int grade[])
{
    // ②
}
```

- (a) 위의 프로그램에서 `get_grade()`와 `print_grade()`의 내부를 작성한 후에 실행하고 그 결과를 기록하라.

실행결과



- (b) 성적이 저장된 배열을 인수로 받아서, 성적이 60점 이하이면 60점으로 변경하는 함수를 작성하고 실행하여 보라. 함수 안에서 배열 원소의 값을 변경하면 원래의 배열 원소의 값도 변경되는가?
- (c) 2개의 배열을 인수로 받아서, 첫 번째 배열을 두 번째 배열로 복사하는 함수 `copy_grade()`를 작성하여, 수정된 프로그램을 컴파일하고 실행하여 보라. 참고로, 이 프로그램의 `main()` 함수 안에 또 하나의 배열을 선언하여야만 한다.
- (d) 배열을 인수로 받아서, 배열 원소의 값을 2배로 하는 `get_double()` 함수를 작성하고 테스트하시오. 단 배열 원소의 값은 100이하로 한정된다.