



실 습 문 제

제6장

1. [if-else 문제]

성적을 입력받아서 60점 이상이면 합격이라고 출력하고 60점 미만이면 불합격이라고 출력하는 프로그램은 다음과 같다.

```
#include <stdio.h>

int main(void)
{
    int score;

    printf("성적을 입력하십시오:");
    scanf("%d", &score);

    if( score >= 60 )
        printf("합격입니다.\n");
    else
        printf("불합격입니다.\n");

    return 0;
}
```

(1) 위의 프로그램을 컴파일하고 실행하여 보라. 다음의 각각의 입력에 대하여 출력을 기록하라.

	실행 #1	실행 #2	실행 #3	실행 #4	실행 #5
입력	50	59	60	61	70
출력					

(2) 사용자가 입력한 점수가 60점 이상이면 printf(“축하합니다”); 문장을 추가 실행하도록 하라.

```
성적을 입력하십시오: 60
축하합니다.
합격입니다.
```

(3) 중첩 if 문을 사용하여 성적에 따라서 메시지를 다양하게 출력하도록 위의 프로그램을 수정하여 보라. 다양한 입력값을 이용하여 프로그램을 테스트하라.

50점 미만 → 불합격입니다.

50점 이상 ~ 60점 미만 → 안타깝게 불합격입니다.

60점 이상 ~ 70점 미만 → 합격입니다.



70점 이상 ~ 80점 미만 → 우수하게 합격하였습니다.

80점 이상 → 아주 우수하게 합격하였습니다.

- (4) 연속적인 if 문을 사용하여 (3)번 문제를 다시 작성하여 보라.
 (5) “60점 이상 ~ 70점 미만”을 처리하기 위하여 if 문을 다음과 같이 작성하여 실행하여 보라. 이 if 문의 조건식은 어떤 경우에 참이 되는가?

```
if(score >= 60) && (score < 70)
    printf("합격입니다.\n");
```

- (6) 조건 연산자 ?를 사용하여 위의 프로그램을 다시 작성하여 보라.

2. [dangling else 문제]

```
#include <stdio.h>

int main(void)
{
    int x, y;

    printf("2개의 정수를 입력하시오.");
    scanf("%d %d", &x, &y);

    if( x == 0 )                // ①
    if( y == 0 )                // ②
    printf("두수 모두 0입니다.\n");
    else                        // ③
    printf("둘 중에서 하나는 0입니다.0\n"); // ④

    return 0;
}
```

- (1) 위의 프로그램을 컴파일하고 실행하여 보라. 다음과 같은 입력에 대하여 출력을 기록하라.

	실행 #1	실행 #2	실행 #3	실행 #4
x	0	0	1	1
y	0	1	0	1
출력				

- (2) ④번 문장은 어떤 경우에 실행되는가? ③번의 else 절은 어떤 if와 대응되는가?
 (3) 만약 ③번의 else 절을 ①의 if에 대응시키려면 어떻게 하여야 하는가?
 (4) 다음의 표와 같이 메시지를 출력할 수 있도록 위의 프로그램을 수정하라.

입력 값	메시지
x=0, y=0	두수 모두 0입니다.
x=0, y≠0	x는 0이고, y는 0이 아닙니다.
x≠0, y=0	x는 0이 아니고, y는 0입니다.
x≠0, y≠0	x, y는 모두 0이 아닙니다.

3. [switch 문제]

키보드에서 문자 하나를 읽어서 모음과 비모음을 구분하는 프로그램을 작성하여 보자. 단 함수를 사용하지 않고 switch 문만을 사용하여 문자를 구분한다.

```
#include <stdio.h>

int main(void)
{
    char c;

    printf("문자를 입력하십시오:");
    c = getchar();

    switch(c)
    {
        case 'a':
            printf("입력된 문자 %c은 모음입니다.", c);
            break
            // 여기에 적절한 case 문을 삽입한다.
    }
    return 0;
}
```

- (1) 모음 'e', 'i', 'u', 'o'에 대한 case 문을 추가하여서 테스트하여 보라.
- (2) 대문자로 모음을 입력한 경우에도 처리할 수 있도록 'A', 'I', 'U', 'E', 'O'에 대한 case 문도 작성하여 보라.
- (3) 비모음을 처리하기 위하여 위의 5가지의 모음을 제외한 나머지 문자들을 default 문을 이용하여 다음과 같은 메시지를 출력하도록 작성하라.
“입력된 문자는 모음이 아닙니다.”
- (4) if-else 문을 이용하여 위의 프로그램을 다시 작성하여 보라.

4. [메뉴 구현]

실제 프로그램에서 자주 나타나는 문제가 사용자를 위한 메뉴 구현이다. 본문에서 설명한 계산기 프로그램을 기본으로 하여, 먼저 메뉴를 화면에 표시하고 사용자가 메뉴를 선택하면 선택한 메뉴를 실행하도록 하여 보자.



```
#include <stdio.h>

int main(void)
{
    char command;
    int x, y;

    printf("===== 메뉴=====\n");
    printf("+ : 더하기\n");
    printf("- : 빼기\n");
    printf("* : 곱하기\n");
    printf("/ : 나누기\n");
    printf("===== 메뉴=====\n");

    printf("메뉴에서 명령어를 선택하시오:");
    scanf("%c", &command);

    printf("피연산자를 입력하시오:");
    scanf("%d %d", &x, &y);

    switch(command)
    {
        case '+':
            printf("연산의 결과: %d\n", x + y);
            break // ①
        case '-':
            printf("연산의 결과: %d\n", x - y);
            break
        case '*':
            printf("연산의 결과: %d\n", x * y);
            break
        case '/':
            printf("연산의 결과: %d\n", x / y);
            break

        default:
            printf("지원되지 않는 연산자입니다. \n");
            break
    }

    return 0;
}
```

- (1) ①번 문장을 생략하면 어떻게 되는지를 관찰하라.
- (2) 메뉴를 다음과 같이 변경하고 여기에 따라 위의 프로그램도 변경하시오.

```
===== 메뉴 =====  
+ , a : 더하기  
- , s : 빼기  
* , m : 곱하기  
/ , d : 나누기  
===== 메뉴 =====
```

- (3) 연산자 %, <, >, &&, ||도 지원할 수 있도록 코드를 추가하라. 메뉴에도 추가하여야 한다.
- (4) 위의 프로그램을 연속적인 if-else 문을 이용하여 다시 작성하여 보라.