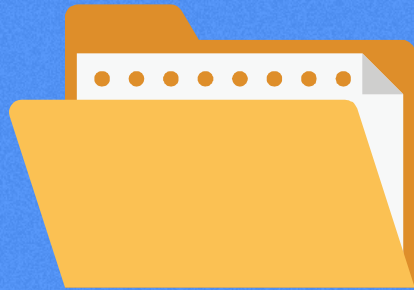


# 컴퓨터 프로그래밍 I



## Lab5(실습5)

조건



# Part I: 기본 프로그래밍 실 습



# 1. 다음에서 서술 내용이 맞으면 0, 틀리면 x 표시

- 1) 프로그램의 실행 흐름에서도 순차적인 실행뿐만 아니라 선택과 반복 등 순차적인 실행을 변형하여 프로그램의 실행 순서를 제어하는 제어문 (control statement)이 제공된다. ( )
- 2) if 문 if (cond) stmt; 에서 조건식 cond가 0이면 stmt를 실행하고, 0이 아니면 stmt를 실행하지 않는다. ( )
- 3) If 문의 else는 문법적으로 같은 블록 내에서 else가 없는 가장 근접한 상위의 if 문에 소속된 else로 해석한다. ( )
- 4) switch 문은 주어진 연산식이 (문자형 또는) 정수형이라면 그 값에 따라 case의 상수 값과 일치하는 부분의 문장들을 수행하는 선택 구문이다. ( )
- 5) switch 문에서 case 다음의 value 값은 변수가 올 수 없으며 그 결과가 정수(또는 문자 상수)여야 하고 중복될 수 있다. ( )
- 6) switch 문에서 case 문 내부에 break 문이 없다면 일치하는 case 문을 실행하고, break 문을 만나기 전까지 다음 case 내부 문장을 실행한다. ( )
- 7) switch 문에서 case 4, 5와 같은 표현이 가능하다. ( )



- 1) 프로그램의 실행 흐름에서도 순차적인 실행뿐만 아니라 선택과 반복 등 순차적인 실행을 변형하여 프로그램의 실행 순서를 제어하는 제어문(control statement)이 제공된다. ( O )
- 2) if 문 if (cond) stmt; 에서 조건식 cond가 0이면 stmt를 실행하고, 0이 아니면 stmt를 실행하지 않는다. ( X )
- 3) If 문의 else는 문법적으로 같은 블록 내에서 else가 없는 가장 근접한 상위의 if 문에 소속된 else로 해석한다. ( O )
- 4) 조건 문장 if는 이미 배운 조건연산자와 매우 유사해서 좀더 편리하게 활용될 수 있다. ( O )
- 5) switch 문은 주어진 연산식이 문자형 또는 정수형이라면 그 값에 따라 case의 상수 값과 일치하는 부분의 문장들을 수행하는 선택 구문이다. ( O )
- 6) switch 문에서 case 다음의 value 값은 변수가 올 수 없으며 그 결과가 정수 또는 문자 상수여야 하고 중복될 수 있다. ( X )
- 7) switch 문에서 case 문 내부에 break 문이 없다면 일치하는 case 문을 실행하고, break 문을 만나기 전까지 다음 case 내부 문장을 실행한다. ( O )
- 8) switch 문에서 case 4, 5와 같은 표현이 가능하다. ( X )

## 2. 비어 있는 부분을 적당히 채우기

- 1) 문장 \_\_\_\_는 조건에 따른 선택을 지원하는 구문이다.
- 2) 조건문 `if (cond) stmt1; else stmt2;`는 조건 `cond`를 만족하면 \_\_\_\_\_을 실행하고, 조건 `cond`를 만족하지 않으면 \_\_\_\_\_를 실행하는 문장이다.
- 3) 속도 `speed`가 40km와 60km 사이라는 조건식은 \_\_\_\_\_라고 표현해야 한다.
- 4) `switch` 문에서 연산식 `exp`의 결과 값은 반드시 문자 또는 \_\_\_\_\_여야 한다.
- 5) 문장 `if (cond1) if (cond2) s1; else s2;`에서 이해를 쉽게 하기 위해 블록을 삽입하면 문장 \_\_\_\_\_와 같다.
- 6) 다중선택 구문인 \_\_\_\_\_문을 사용하면, 문장 `if else`가 여러 번 계속 반복되는 구문을 간략하게 구현할 수 있다.
- 7) `switch` 문은 주어진 연산식이 (문자형 또는) \_\_\_\_\_이라면 그 값에 따라 `case`의 상수 값과 일치하는 부분의 문장들을 수행하는 선택 구문이다.
- 8) \_\_\_\_\_는 선택적으로 없거나 하나이며 어디에 위치해도 모든 `case` 처리를 하지 않은 경우 실행되며, 다른 `case`가 뒤에 있다면 `break`가 필요하다.
- 9) 두 수 중에서 최대 값을 반환하는 조건식은 \_\_\_\_\_ ? `x : y`이다.

- 1) 문장 if는 조건에 따른 선택을 지원하는 구문이다.
- 2) 조건문 if (cond) stmt1; else stmt2; 는 조건 cond를 만족하면 stmt1을 실행하고, 조건 cond를 만족하지 않으면 stmt2를 실행하는 문장이다.
- 3) 속도 speed가 40km와 60km 사이라는 조건식은 (40 <= speed && speed <= 60)라고 표현해야 한다.
- 4) switch 문에서 연산식 exp 의 결과 값은 반드시 문자 또는 정수여야 한다.
- 5) 조건식 (n % 2 )는 0이 아니면(참)이면 홀수이고 0이면(거짓)이면 짝수이다.
- 6) 문장 if (cond1) if (cond2) s1; else s2; 에서 이해를 쉽게 하기 위해 블록을 삽입하면 문장 if (cond1) { if (cond2) s1; else s2; } 와 같다.
- 7) 다중선택 구문인 switch 문을 사용하면, 문장 if else가 여러 번 계속 반복되는 구문을 간략하게 구현할 수 있다.
- 8) switch 문은 주어진 연산식이 문자형 또는 정수형이라면 그 값에 따라 case의 상수 값과 일치하는 부분의 문장들을 수행하는 선택 구문이다.
- 9) default 는 선택적으로 없거나 하나이며 어디에 위치해도 모든 case 처리를 하지 않은 경우 실행되며, 다른 case가 뒤에 있다면 break가 필요하다.
- 10) 두 수 중에서 최대 값을 반환하는 조건식은  $x > y ? x : y$  이다.

### 3. 다음 부분 소스에서 문법오류 및 논리오류를 찾아 수정

- 1) `if (grade >= 4.0);`  
    `printf("성적이 우수합니다.\n");`
- 2) `if (30 <= age < 50)`  
    `printf("중년입니다.\n");`
- 3) `if (type = 'v')`  
    `printf("VIP 회원입니다.\n");`
- 4) `if (type == 'm')`  
    `printf("남자입니다.\n");`  
    `else (type == 'w')`  
        `printf("여자입니다.\n");`
- 5) `if (point >= 80)`  
    `if (point < 90)`  
        `printf("점수가 80 이상이고 90 미만이다.\n");`  
    `else`  
        `printf("점수가 80 미만이다.\n");`

```
① if (grade >= 4.0)
    printf("성적이 우수합니다.\n");
② if (30 <= age && age < 50)
    printf("중년입니다.\n");
③ if (type == 'v')
    printf("VIP 회원입니다.\n");
④ if (type == 'm')
    printf("남자입니다.\n");
else if (type == 'w')
    printf("여자입니다.\n");
```

```
⑤ if (point >= 80) {
    if (point < 90)
        printf("점수가 80 이상이고 90 미만이다.\n");
} else
    printf("점수가 80 미만이다.\n");
```



## 4. 다음 프로그램의 실행 결과는?

실행해보지 않고 결과를 얻어야 함

```
#include <stdio.h>

int main(void)
{
    int point = 7;

    if (point >= 10) {
        printf("A\n");
    }
    else if (point >= 8)      {
        printf("B\n");
    }
    else if (point >= 6)      {
        printf("C\n");
    }
    else if (point >= 4)      {
        printf("D\n");
    }
    else {
        printf("F\n");
    }

    return 0;
}
```

## 5. 다음 프로그램의 실행 결과는?

```
#include <stdio.h>

int main(void)
{
    int op = 3, n = 0 , i = 5;
    switch (op) {
        case 1:
            i++;
        case 2:
            n++;
        case 3:
            i++;
        case 4:
            n++;

        default:
            i++;
            n++;
    }
    printf("%d %d\n", i, n);
    return 0;
}
```

## 6. 표준입력으로 받은 두 실수의 대소에 따라 다양한 연산을 수행하여 그 결과를 출력

- 두 실수를 입력 받아 두 실수의 연산값이 출력되는 프로그램
  - 만일  $x > y$  이면  $x / y$  연산값 출력
  - 만일  $x < y$  이면  $x + y$  연산값 출력
  - 만일  $x == y$  이면  $x * y$  연산값 출력
- 결과
  - 두 실수를 입력: 32.765 3.987
  - 연산 결과: 8.22
- 우측 코드의 빈칸을 채워서 소스코드를 작성하고 위에 제시된 결과가 나오는지 확인해 본다

```
tworeal.c
01 // file: tworeal.c
02 #define _CRT_SECURE_NO_WARNINGS //scanf() 오류를 방지하기 위한 상수 정의
03
04 #include <stdio.h>
05
06 int main(void)
07 {
08     double x = 0, y = 0, result = 0;
09
10     printf("두 실수를 입력: ");
11     scanf("%lf %lf", &x, &y);
12
13     -----
14     {
15         result = x / y;
16     }
17     else if (x == y)
18     {
19         result = x * y;
20     }
21     -----
22     {
23         result = x + y;
24     }
25
26     printf("연산 결과: %.2f\n", result);
27
28     return 0;
29 }
```

## 7. 표준입력으로 받은 세 정수의 최대값을 출력

- 표준입력으로 받은 세 정수에서 최대값이 출력되는 프로그램
  - 먼저 조건식  $x > y$  이 참이면  $x$ 와  $y$ 의 최대값, 거짓이면  $x$ 와  $y$ 의 최소값
- 결과
  - 세 정수를 입력: 10 30 20
  - 최대 수: 30
- 우측 코드의 빈칸을 채워서 소스코드를 작성하고 위에 제시된 결과가 나오는지 확인해 본다

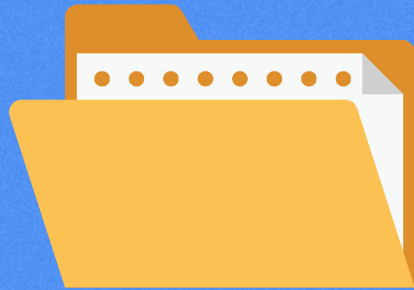
```
maxof3.c
61 // file: maxof3.c
62 #define _CRT_SECURE_NO_WARNINGS
63
64 #include <stdio.h>
65
66 int main(void)
67 {
68     int x, y, z;
69
70     printf("세 정수를 입력: ");
71     scanf("%d %d %d", &x, &y, &z);
72
73     if (x > y)
74     {
75         if (_____)
76             printf("최대수: %d\n", x);
77         else
78             printf("최대수: %d\n", z);
79     }
80     else
81     {
82         if (_____)
83             printf("최대수: %d\n", y);
84         else
85             printf("최대수: %d\n", z);
86     }
87
88     return 0;
89 }
```

## 8. break의 적절한 사용

- ◆ 표준입력으로 1년 중 해당하는 월을 입력 받아 그 달에 맞는 계절을 출력하는 프로그램
  - case 문 내부에 break 문이 없다면 실행순서가 어떻게 달라지는가?
- ◆ case 4: case 5:을 case 4, 5로 바꾸면 어떻게 되는가?

```
01 // file: seasonswitch.c
02 #define _CRT_SECURE_NO_WARNINGS //scanf() 오류
03
04 #include <stdio.h>
05
06
07 int main(void)
08 {
09     int month;
10
11     printf("년도의 월(month)을 입력: ");
12     scanf("%d", &month);
13
14     switch (month) {
15     case 4: case 5:
16         printf("%d월은 봄입니다.\n", month);
17         break;
18
19     case 6: case 7: case 8:
20         printf("%d월은 여름입니다.\n", month);
21         break;
22
23     case 9: case 10: case 11:
24         printf("%d월은 가을입니다.\n", month);
25         break;
26
27     case 12: case 1: case 2: case 3:
28         printf("%d월은 겨울입니다.\n", month);
29         break;
30
31     default:
32         printf("월(month)을 잘못 입력하셨습니다.\n");
33     }
34
35     return 0;
36 }
```



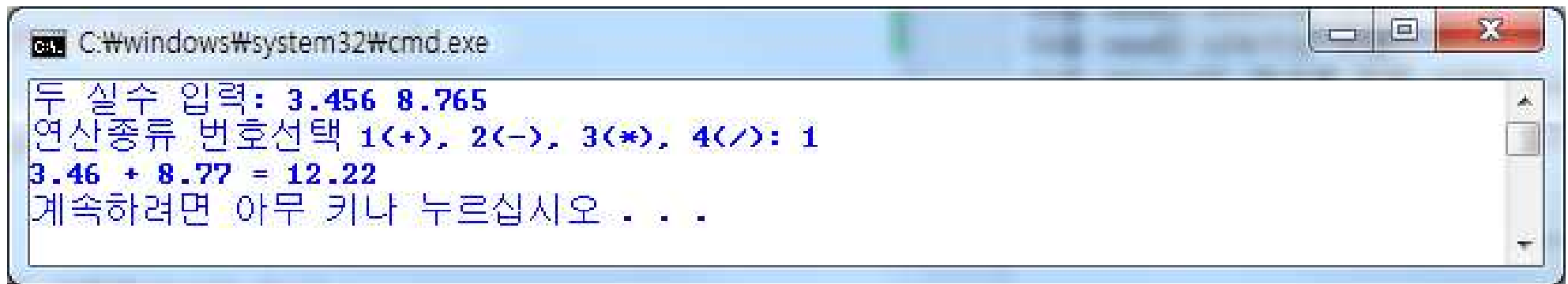


## Part II : 응용 프로그래밍 실습



## 9.

- 다음을 참고로 표준입력으로 받은 두 실수의 연산을 수행하는 프로그램을 작성.
  - 조건선택 문을 사용하며, 다음과 같은 입출력이 되도록



```
C:\windows\system32\cmd.exe
두 실수 입력: 3.456 8.765
연산종류 번호선택 1(<+>), 2(<->), 3(<*>), 4(</>): 1
3.46 + 8.77 = 12.22
계속하려면 아무 키나 누르십시오 . . .
```

## 10.

- 전기 요금을 계산하는 프로그램을 작성한다. 전기 요금은 다음의 3가지의 분류가 존재한다. 먼저 사용자로부터 분류 번호를 받은 후에 각 분류에 따라서 전력량을 입력받아서 전기 요금을 계산한다. 또한 계절에 따라서도 전기 요금이 달라지므로 사용자로부터 몇 월인지도 입력받도록 한다.

구분	기본요금	전력량요금(원/kw)		
		여름(7월 ~ 8월)	봄,가을(4월 ~ 6,9월)	겨울(10 ~ 3월)
1.교육용	4,090	73.0	46.9	50.5
2.농사용	1,070	36.4		
3.일반용	5,320	94.2	62.7	66.8

예: 분류번호=3, 7월, 사용 전력량 10kw → 전기요금= 5320+94.2x10

## 11.

- 표준입력으로 입력받은 두 실수  $x, y$ 에 대해 다음과 같은 연산 값을 출력하는 프로그램을 작성하라.
  - $x$ : 양수  $y$ : 양수 :  $x + y$
  - $x$ : 양수  $y$ : 0 또는 음수 :  $x - y$
  - $x$ : 0 또는 음수  $y$ : 양수 :  $-x + y$
  - $x$ : 0 또는 음수  $y$ : 0 또는 음수 :  $-x - y$

## 12.

- 다음을 참고로 표준입력으로 입력 받은 년도의 윤년을 판단하는 프로그램을 if 문을 사용하여 작성.
  1. 연수가 4로 나누어 떨어지는 해는 우선 윤년으로 하고,
  2. 1번 중에서 100으로 나누어 떨어지는 해는 평년으로 하며,
  3. 다만 400으로 나누어 떨어지는 해는 윤년으로 정한다



## 13.

- 앞의 실습문제를 참고로 표준입력으로 입력 받은 년도와 달에 대해 해당 달의 말일을 출력하는 프로그램을 switch 문을 사용하여 작성하시오.