7 C 제어 Statements: Branching과 Jumps

■ Keywords

if , else , switch , continue

break , case , default , goto

■ Operators && || ?:

■ Functions getchar() , putchar() , ctype.h

■ if, if else statements 사용법

■ logical operators를 사용하여 relational expressions을 작성

■ C’s conditional operator

■ switch statement

■ break , continue , goto jumps

■ C의 character I/O functions 사용 - getchar(), putchar()

**7.1 if Statement**

* if statement을 branching statement 또는 selection statement 라 부름

if ( expression )

statement

if (score > big)

printf("Jackpot!\n"); // simple statement

if (joe > ron)

{ // compound statement

joecash++;

printf("You lose, Ron.\n");

}

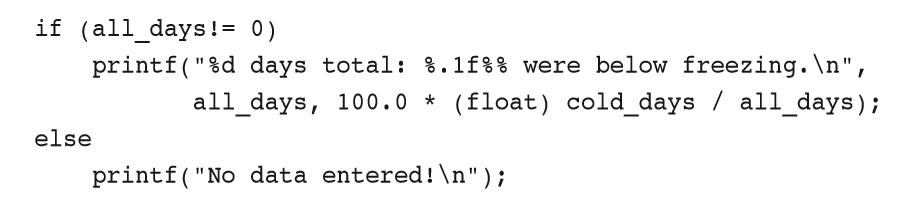
**7.2 if Statemen의 else**

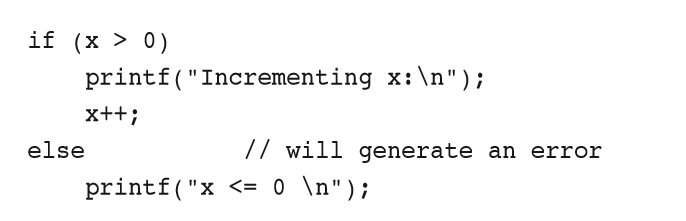
if ( expression )

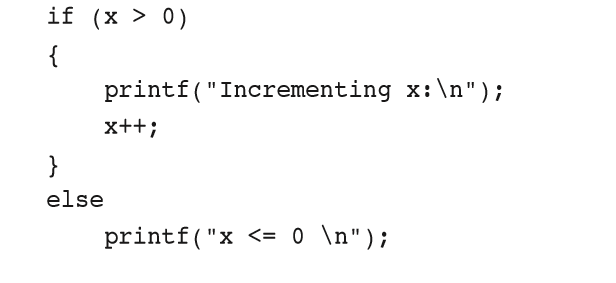
statement1

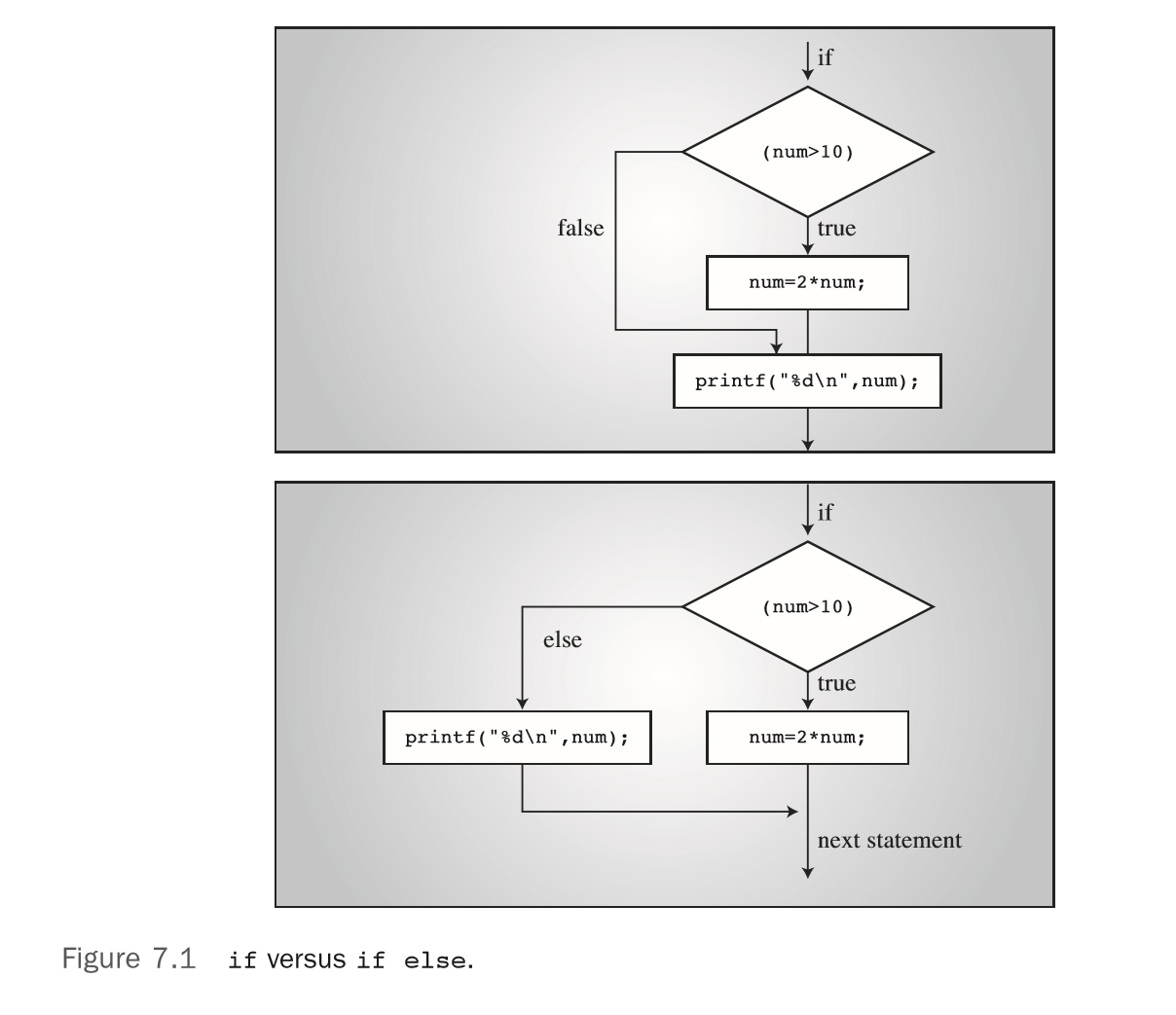
else

statement2





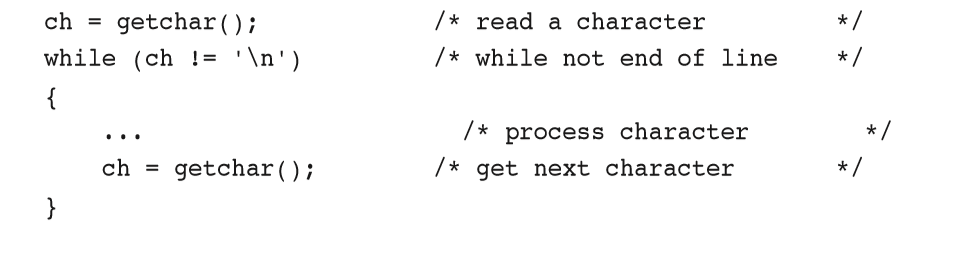


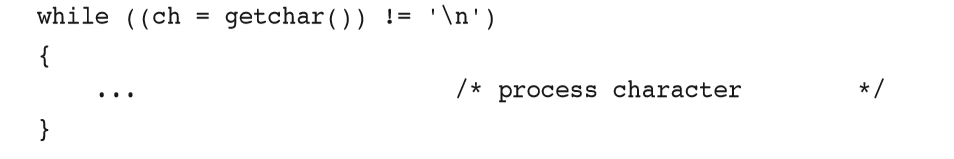


**7.2.1 getchar()와 putchar()**

ch = getchar();

scanf("%c", &ch);

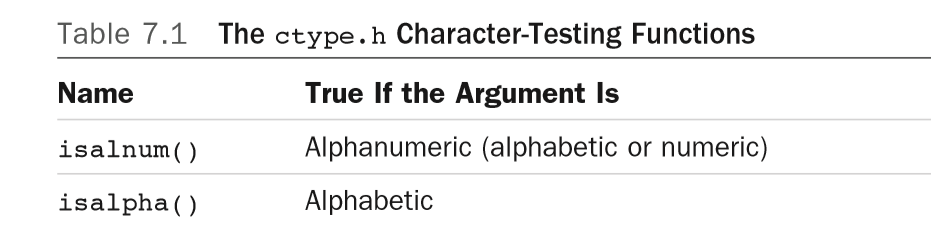


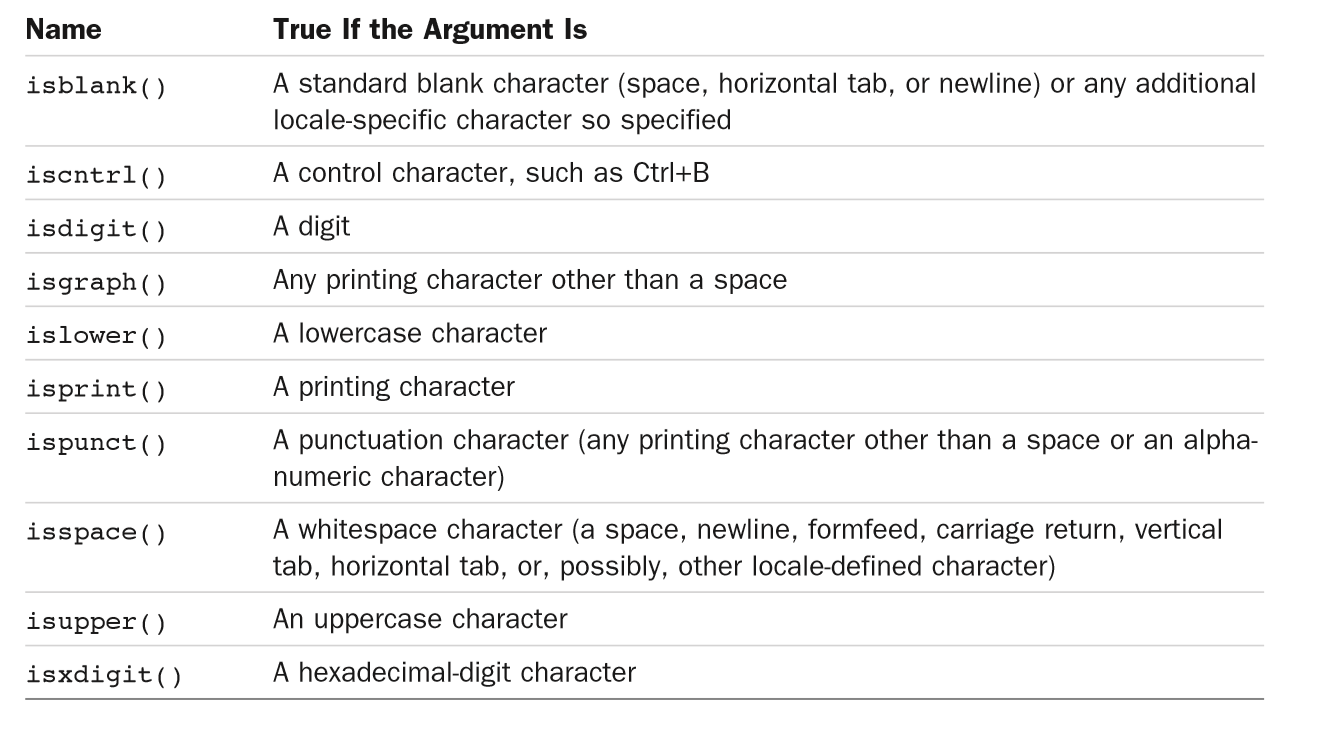


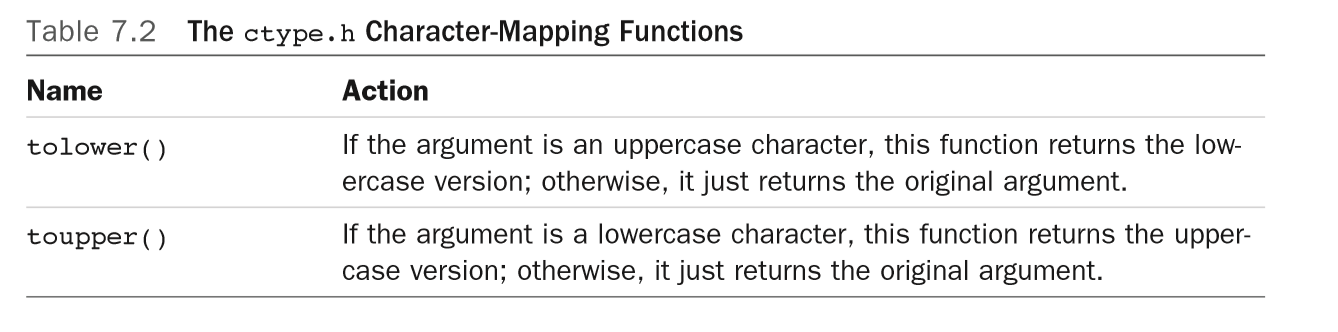
문제: while (ch = getchar() != '\n')

**7.2.2 ctype.h의 Character Functions**

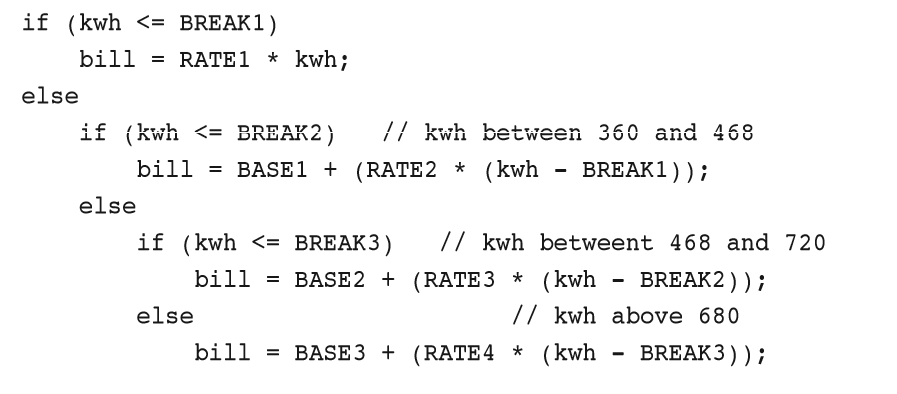
* isalpha() function 은 argument가 letter이면 nonzero 값을 return한다

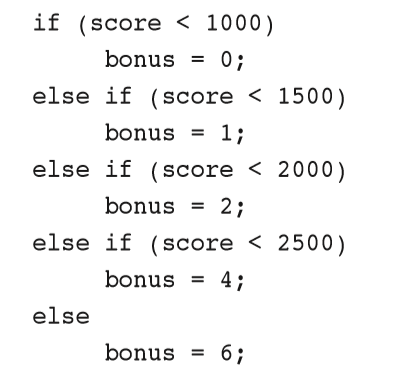




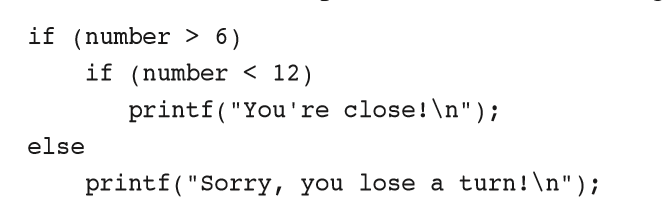


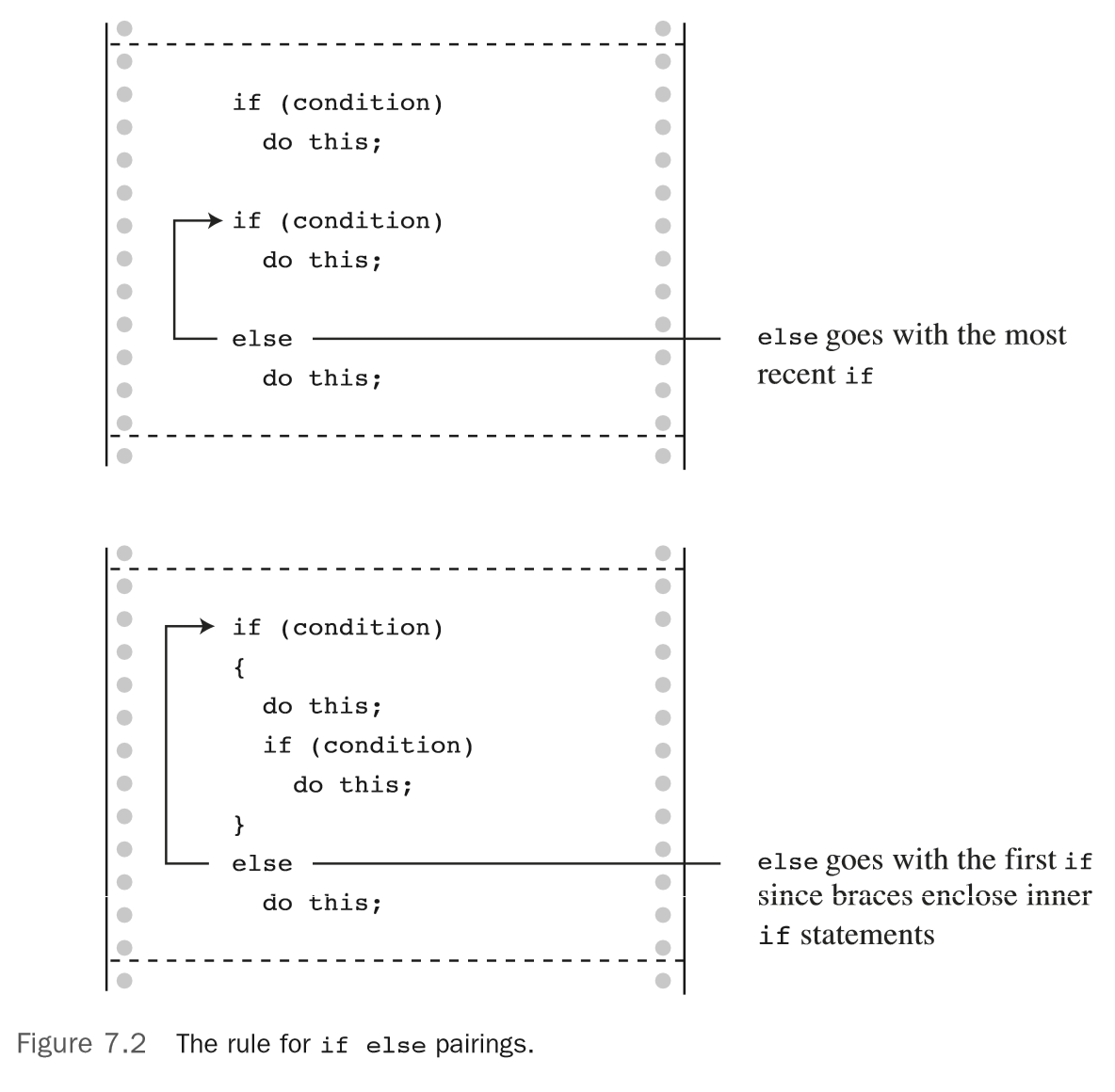
**7.2.3 여러개의 else if**

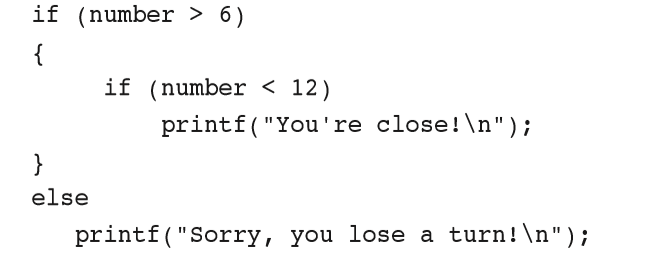




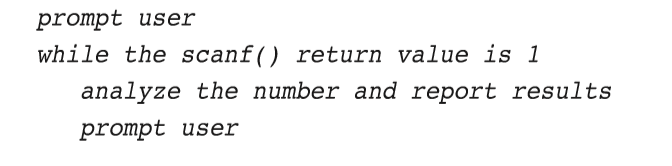
**7.2.4 else와 if의 대응 관계**

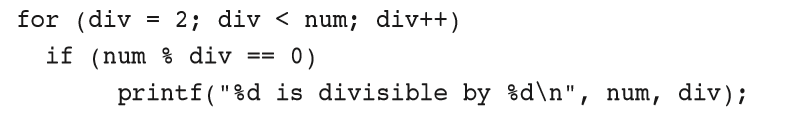


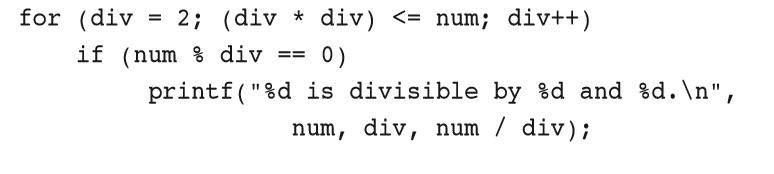


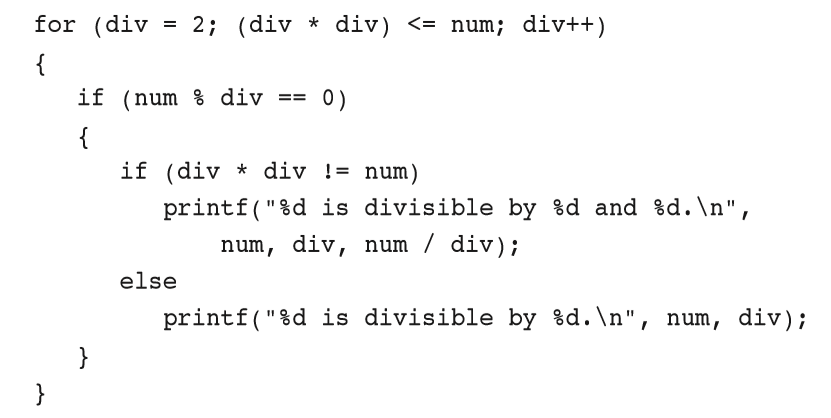


**7.2.5 Nested if**

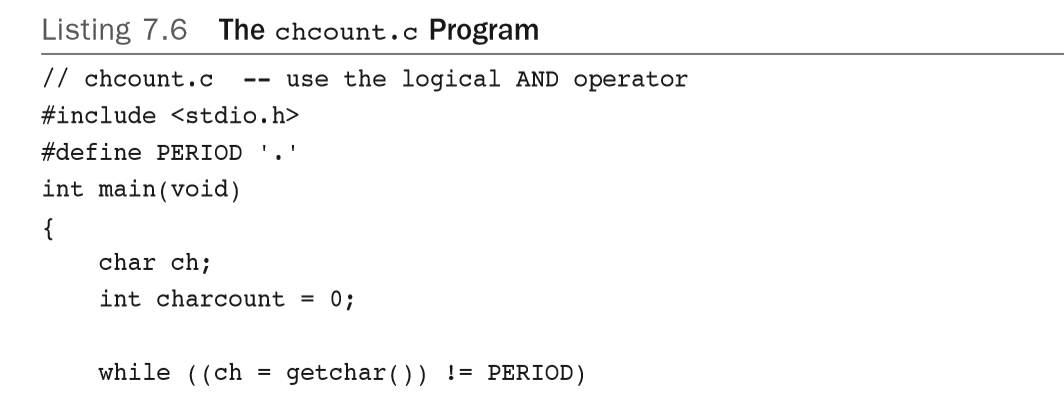


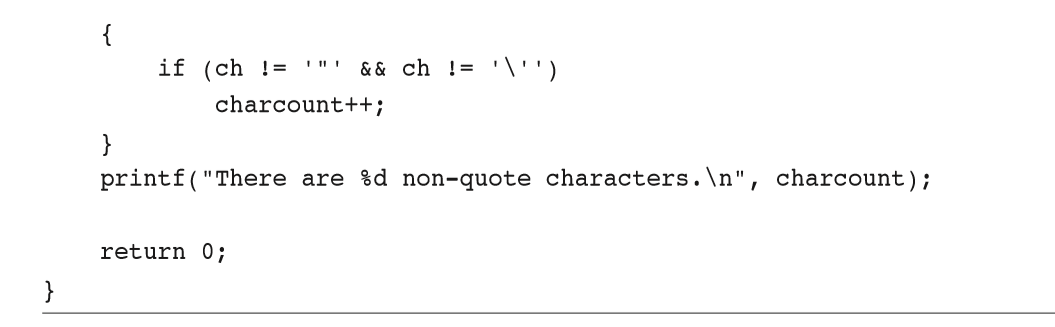


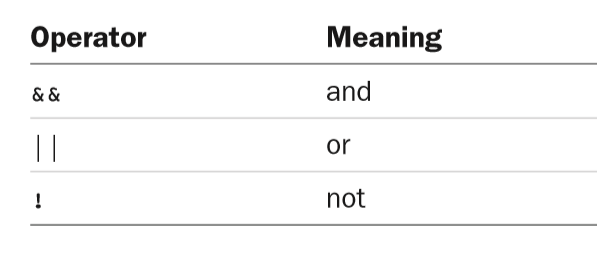




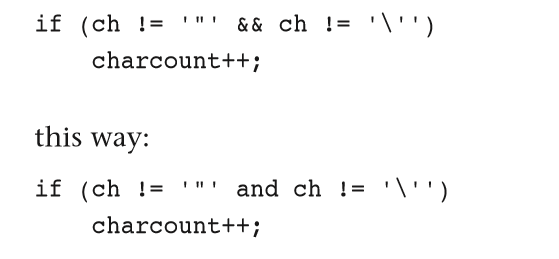
**7.3 Logical operator**

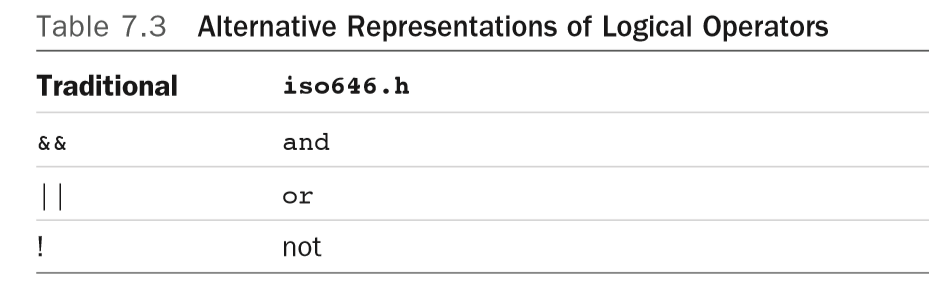






**7.3.1 iso646.h 사용시 and or 사용**





**7.3.2 Precedence**

a > b && b > c || b > d

((a > b) && (b > c)) || (b > d)

**7.3.3 Evaluation 순서**

while ((c = getchar()) != ' ' && c != '\n')

if (number != 0 && 12/number == 2)

printf("The number is 5 or 6.\n");

while ( x++ < 10 && x + y < 20)

**7.3.4 최소 및 최대 범위 지정 방법**

if (range >= 90 && range <= 100)

printf("Good show!\n");

if (90 <= range <= 100) // NO! Don't do it!

printf("Good show!\n");

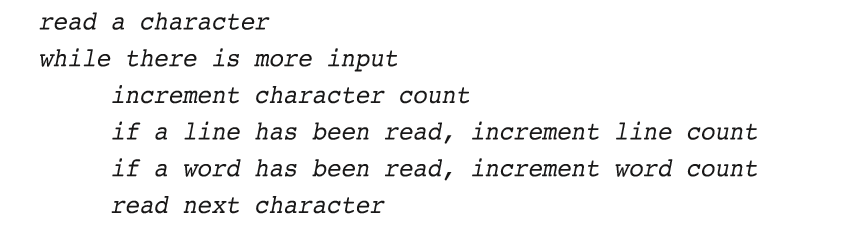
if (ch >= 'a' && ch <= 'z')

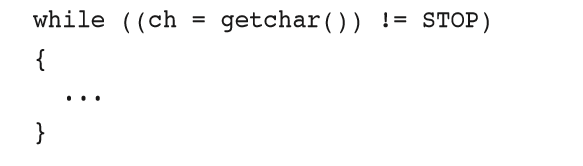
printf("That's a lowercase character.\n");

if (islower(ch))

printf("That's a lowercase character.\n");

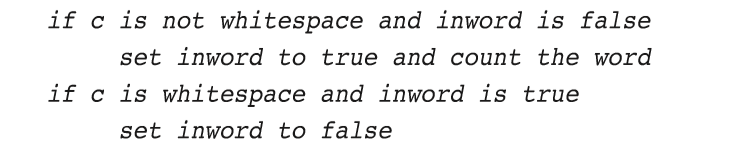
**7.4 Word-Count Program**





c != ' ' && c != '\n' && c != '\t' /\* true if c is not whitespace \*/

c == ' ' || c == '\n' || c == '\t' /\* true if c is whitespace \*/

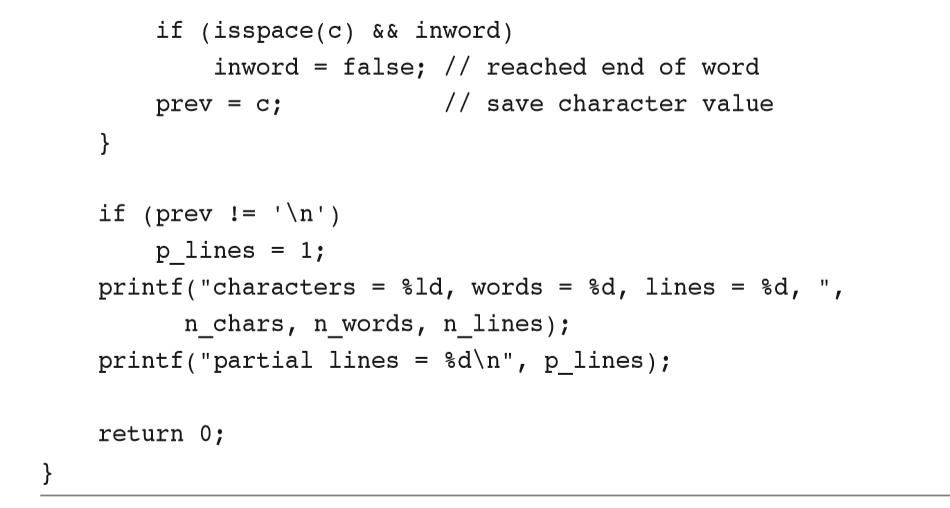


if (inword)

instead of if (inword == true)

if (!inword)

instead of if (inword == false)



**7.5 Conditional Operator: ?:**

x = (y < 0) ? -y : y;

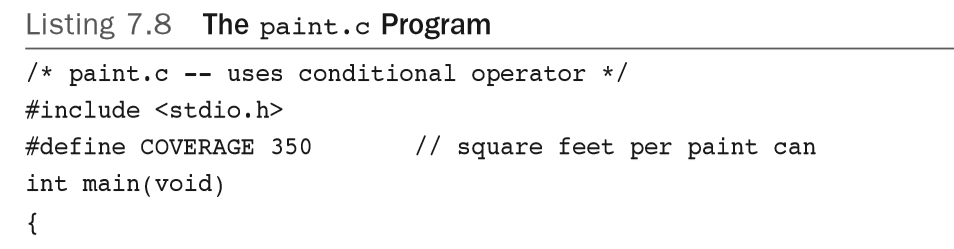
* if else 문도 expression이다
* expression은 evaluate 되고 값을 계산
* x = -y 또는 x = y가 되는 것이다.

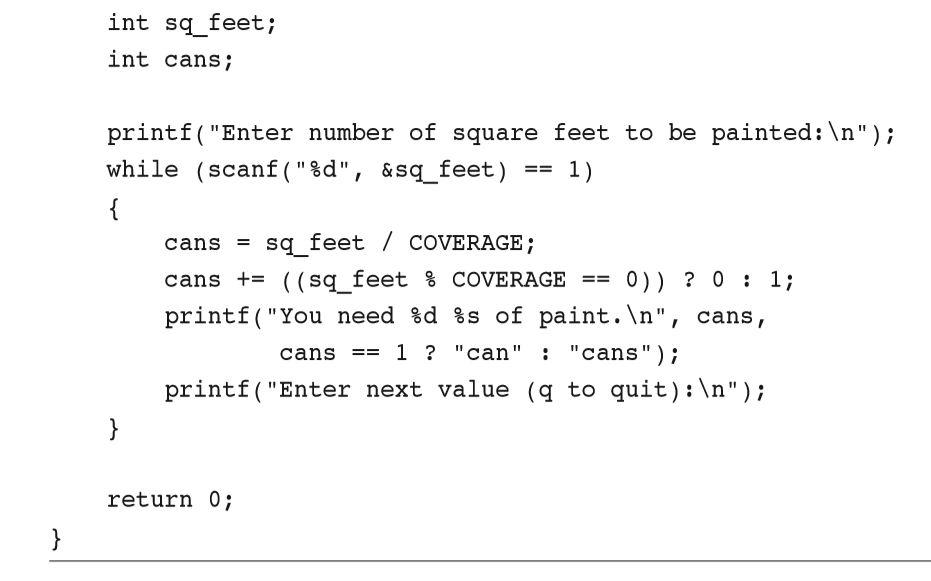
“If y is less than zero, x = -y; otherwise, x = y.”

if (y < 0) x = -y; else x = y;

expression1 ? expression2 : expression3

//x = if (y<0) –y; else y; 오류 – 두개의 statement이다



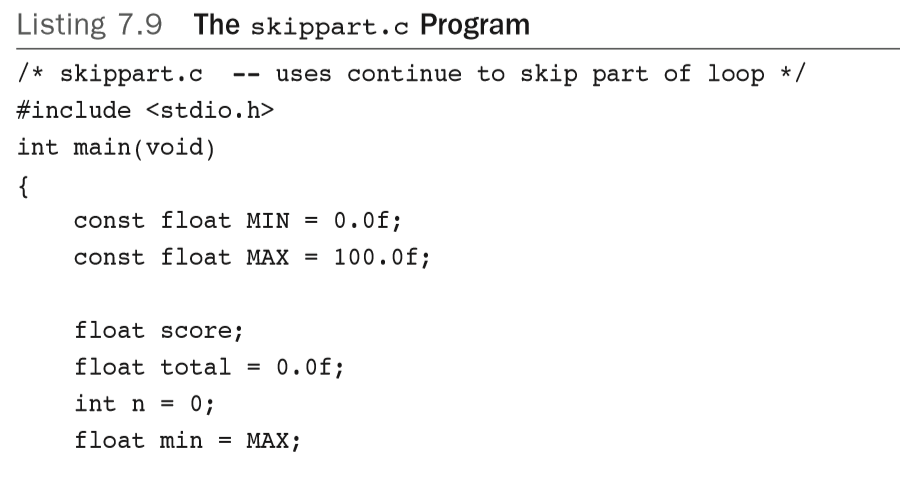


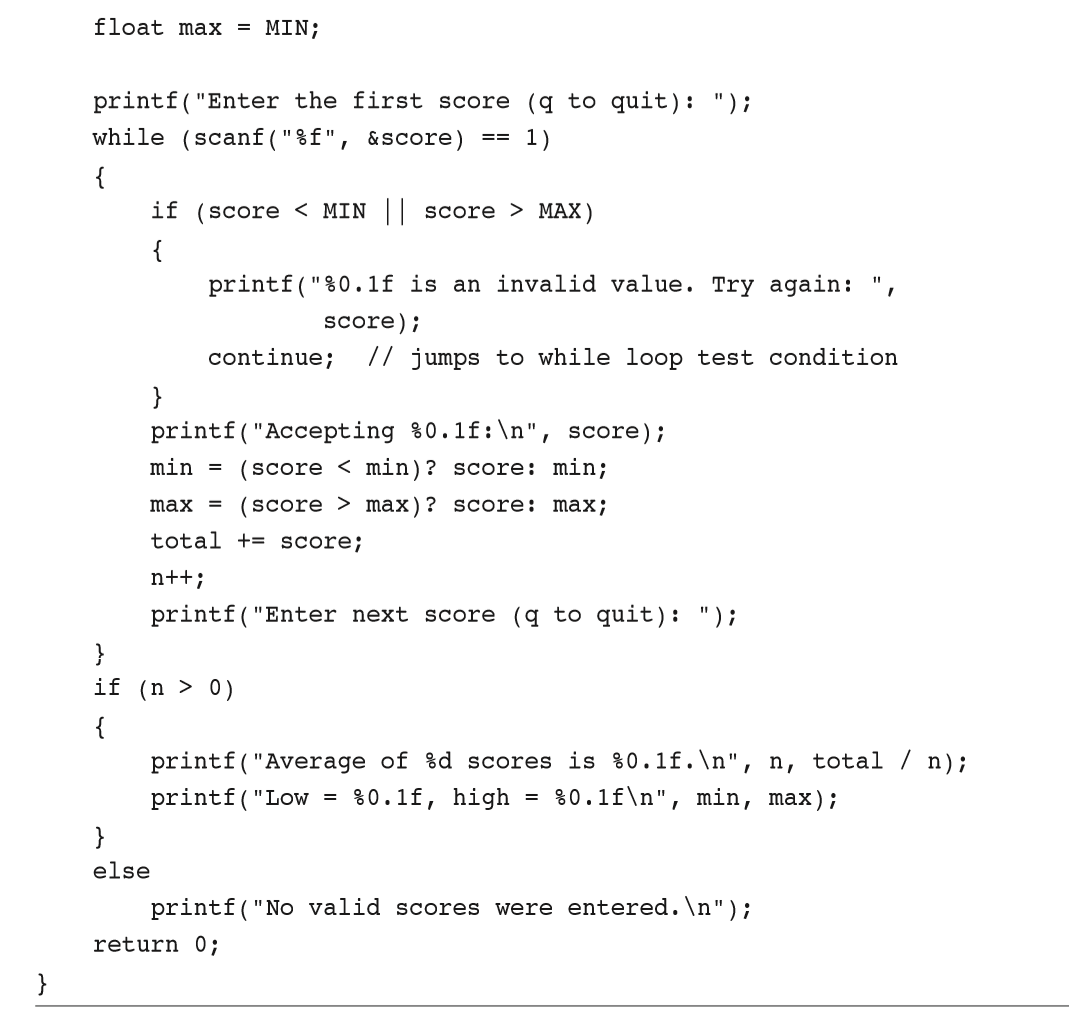
cans += ((sq\_feet % COVERAGE == 0)) ? 0 : 1;

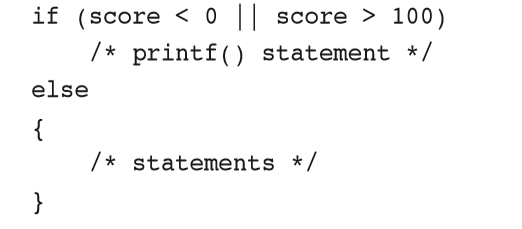
cans == 1 ? "can" : "cans");

**7.6 Loop 사용시 추가로 사용되는 keyword: continue and break**

**7.6.1 continue Statement**







if (score >= 0 && score <= 100)

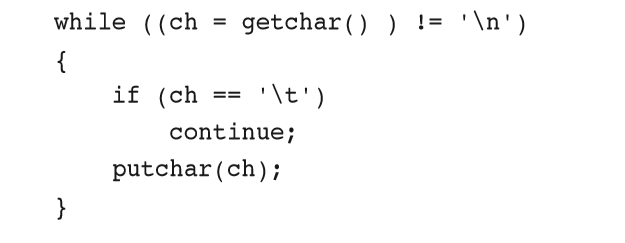
{

/\* statements \*/

}

while (getchar() != '\n')

continue;



while ((ch = getchar()) != '\n')

if (ch != '\t')

putchar(ch);

count = 0;

while (count < 10)

{

ch = getchar();

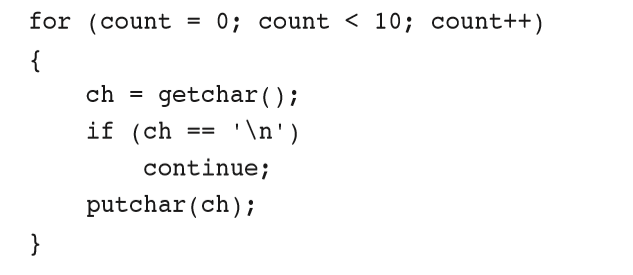
if (ch == '\n')

continue;

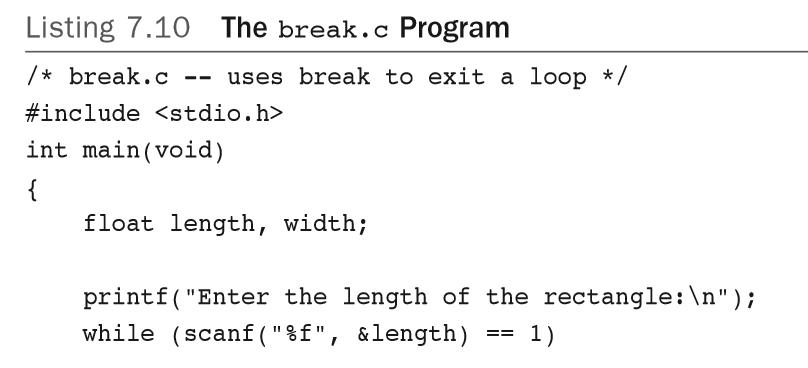
putchar(ch);

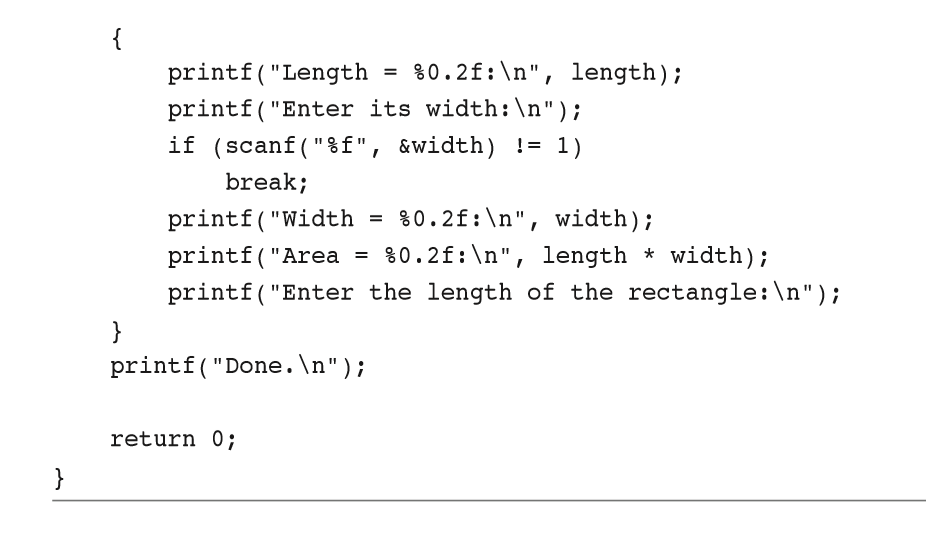
count++;

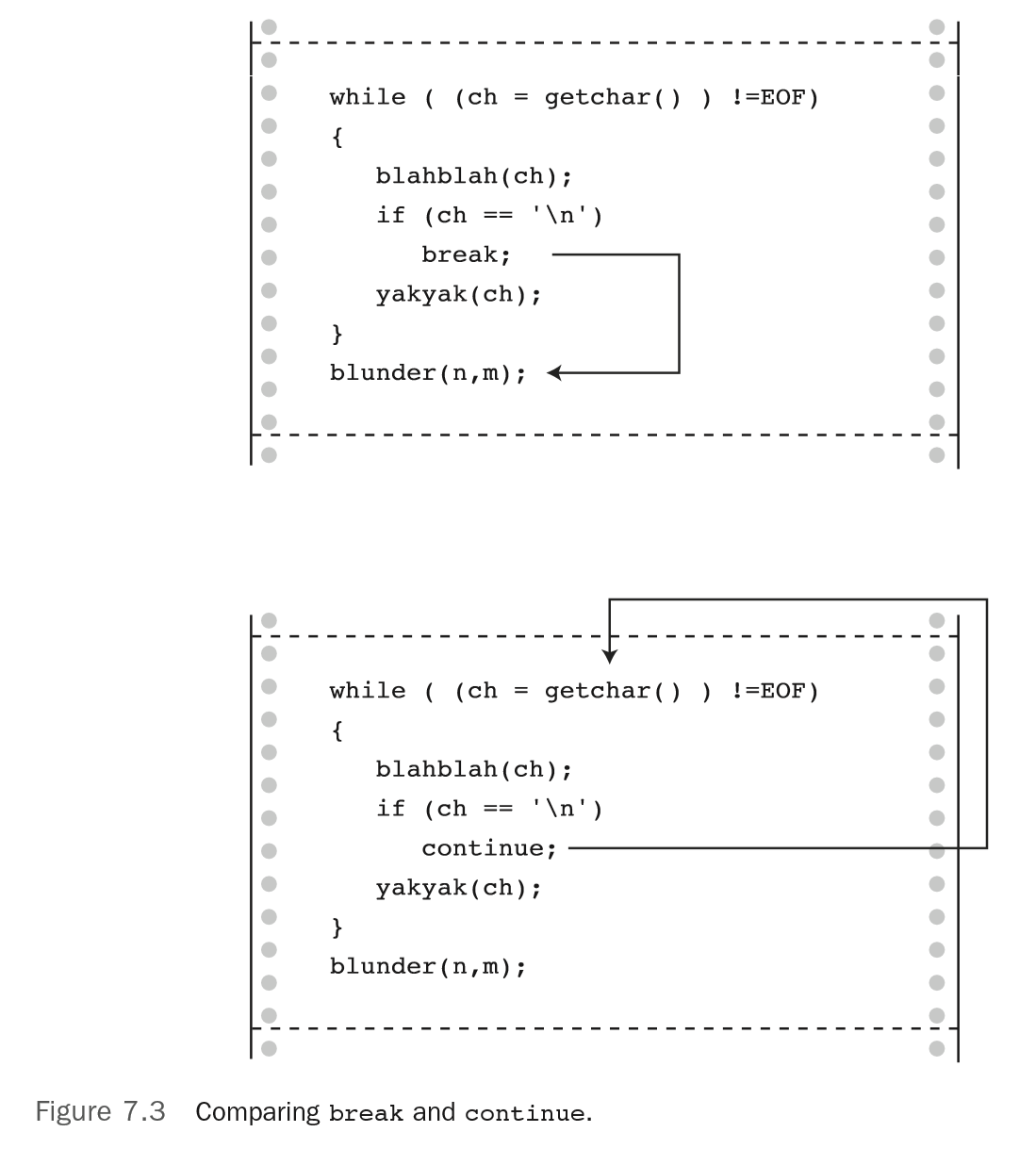
}

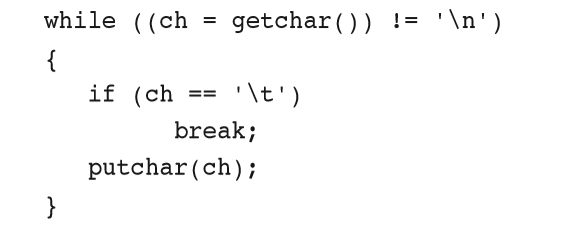


**7.6.2 break Statement**



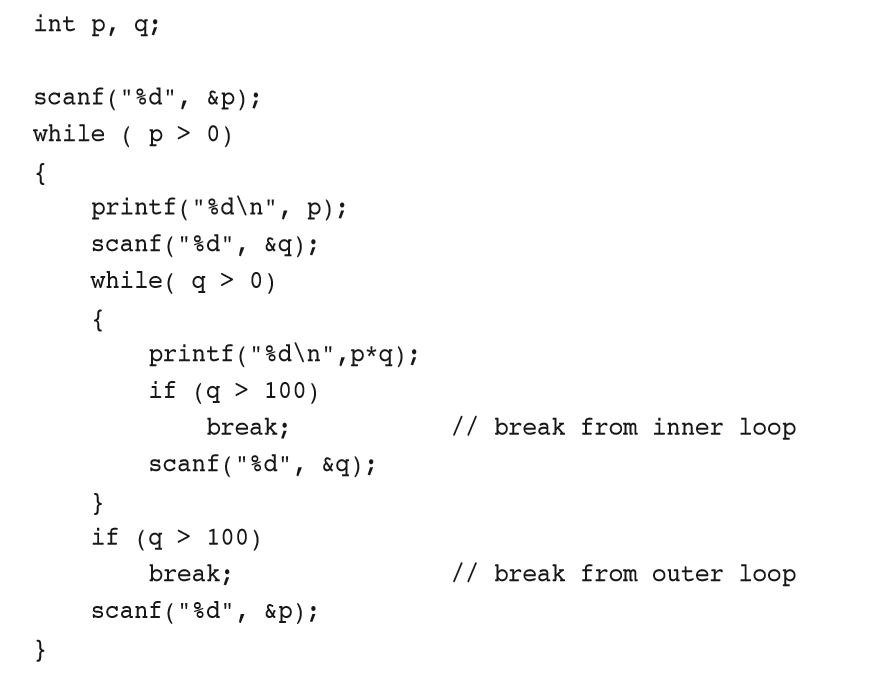






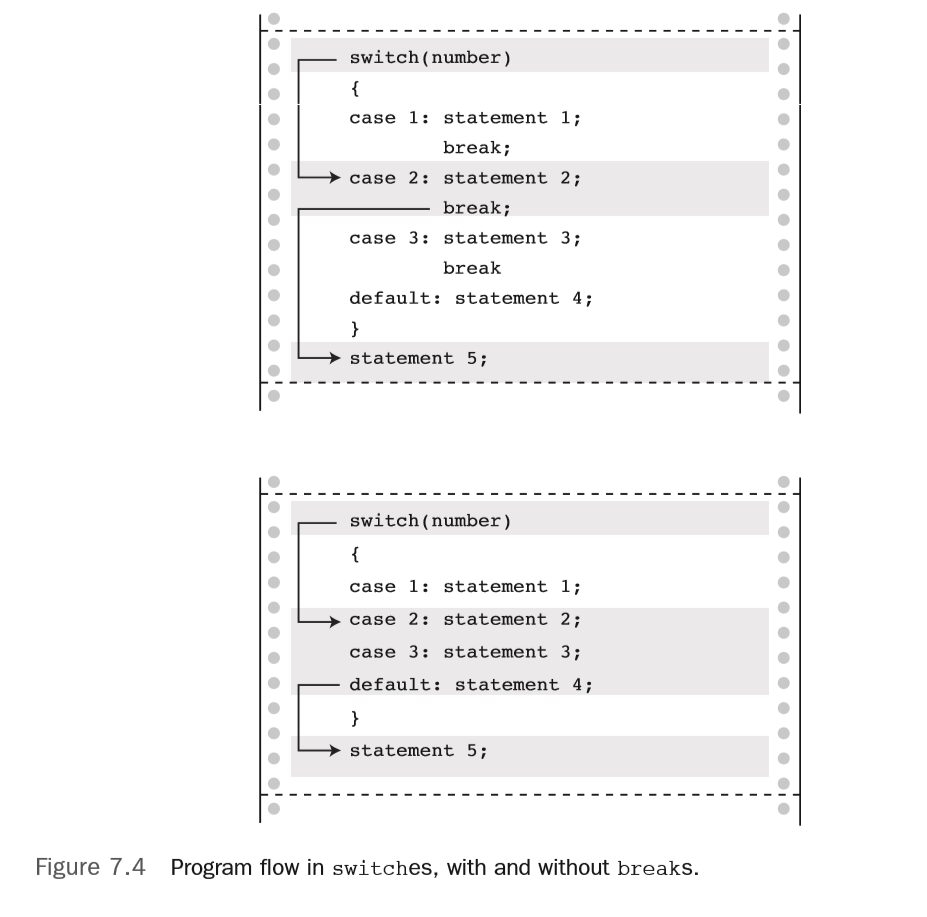
while ((ch = getchar() ) != '\n' && ch != '\t')

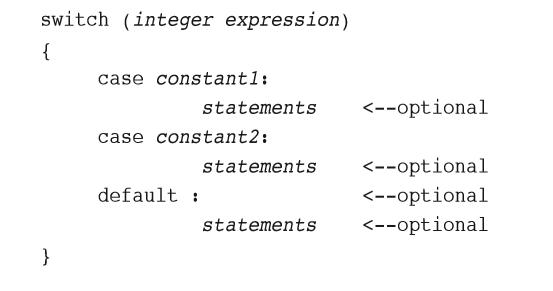
putchar(ch);



**7.7 Multiple Choice: switch와 break**

**7.7.1 switch Statement의 사용**





**7.7.2 getchar() 사용**

while (getchar() != '\n')

continue; /\* skip rest of input line \*/

if (ch == '\n')

continue;

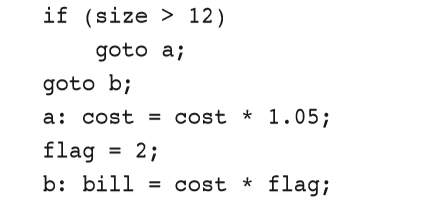
**7.7.3 Multiple Labels**

**7.8 goto Statement**

goto part2;

part2: printf("Refined analysis:\n");

**7.8.1 goto 사용 회피가 좋다**



if (size > 12)

{

cost = cost \* 1.05;

flag = 2;

}

bill = cost \* flag;

