06\_반복문

학습목표

While 반복 구조를 이해하고 사용할 수 있다.

Do-while 반복 구조를 이해하고 사용할 수 있다.

For 반복 구조를 이해하고 사용할 수 있다.

반복문에서의 break와 continue 사용법을 이해한다.

1. 반복이란?  
   반복은 어떤 단계를 반복하게 하는 것으로 반복 구조를 사용하면 프로그램이 간단하고 빠르게 된다. 반복 처리는 동일한 처리 과정을 정해진 횟수만큼 반복하거나 조건에 따라서 반복하게 할 수 있다. 따라서 반복 처리는 프로그래머가 반드시 잘 알아야 할 긴요한 기법이다.

-반복문의 종류

반복문에는 while문과 for문이 있다.

조건 제어 반복(while문): 특정한 조건이 만족되면 계속 반복한다.

횟수 제어 반복(for문): 정해진 횟수만큼 반복한다.

1. While문

형식:

While(조건식) {

//문장들;

}

조건식이 참이면 문장을 반복 실행한다. 반복되는 문장이 하나이면 중괄호를 생략할 수 있다. 조건식의 값이 거짓이 되면 반복을 중단한다. 문장은 단일 명령어 문장이거나 블록이 될 수 있다.

1. Do-while문

Do-while문은 while문과 비슷하나 반복 조건을 루프의 처음이 아니라 루프의 끝에서 검사한다는 것이 다르다.

Do {  
//문장들;

}while(조건식);

일단 문장을 한 번 실행한 후에 조건을 검사하여 반복 여부를 결정한다.

Do-while문은 적어도 한번은 실행되어야 하는 경우나 사용자로부터 메뉴나 데이터를 입력을 받아야 하는 경우에 많이 사용된다.

1. For문

for문은 일정한 횟수만큼 반복할 때 유용하게 사용된다. 흔히 for 루프(loop)라 부른다.

For(초기식; 조건식; 증감식) {  
//문장들;

}

초기식을 실행한 후에 조건식의 값이 참인 동안, 문장을 반복한다. 한번 반복이 끝날 때마다 증감식이 실행된다.

-초기식은 루프를 초기화한다. 루프가 시작할 때 딱 한번 실행된다.

-만약 초기식이 거짓으로 계산되면 루프는 종료하게 된다.

-증감식은 한 번의 반복 뒤에 실행된다. 대게 증감식은 루프 제어 변수를 증가하거나 감소하게 된다. 루프 제어 변수란 루프를 제어할 목적으로만 사용하는 변수이다.

for문에서 주의할 점

-조건식에는 참, 거짓을 판별할 수 있는 수식이면 어떤 것이던지 가능하다. 다음과 같이 논리 연산자를 이용하여 여러 가지 조건을 결합할 수도 있다.

-for문의 3부분 중에서 어떤 부분은 비어 있을 수도 있다. 만약 이들 3부분이 전부 비어 있으면 루프가 무한히 반복된다.

-증감식에는 주로 증가 수식이 많이 이용되지만 때에 따라서는 감소 수식도 사용된다.

1. 중첩 반복문

반복문은 중첩되어 사용될 수 있다. 반복문 안에 반복문이 실행될 수 있는데, 이러한 형태를 중첩 반복문(nested loop)이라고 한다. 중첩 반복문에서 가장 주의할 점은 각각의 반복문을 제어하는 변수가 달라야 한다는 점이다. 만약 같은 변수가 사용되면 논리적인 오류가 발생할 가능성이 높다.

1. Break와 continue

break문은 반복 루프에서 벗어나기 위해 사용된다. 반복 루프 안에서 break문이 실행되면 반복 루프는 즉시 중단되고 반복 루프 다음에 있는 문장이 실행된다.

break문은 원하는 만큼 얼마든지 둘 수 있지만 너무 많이 사용하면 코드가 난해해진다. 따라서 특수하게 반복 루프를 중단할 경우에만 break문은 사용하는 것이 좋다.

continue문은 현재 수행하고 있는 반복 과정의 나머지를 건너뛰고 다음 반복을 시작하게 만든다. 반복 루프에서 continue문을 만나게 되면 continue문 다음에 있는 후속 코드들은 실행되지 않는다.