

# 프로그래밍을 통한 논리적 사유 연습





# 9장. 반복 문

(while, do while)

# while문

- while문

- 횟수를 정확하게 알지는 못하지만 반복의 조건을 알고 있을 때 사용

반복 횟수가 정해지지 않은 경우 for문과 while문 비교

for문을 사용할 때(반복이 모호함)	while문을 사용할 때(간단하게 처리할 수 있음)
<pre>char c; int i; c = getchar();  for(i=0; c != 'q'; i++){     putchar(c);     c = getchar(); }</pre>	<pre>char c; while((c=getchar()) != 'q')     putchar(c);</pre>

# while문

- while문의 기본 형식

```
while(조건식)
{
    명령문 1;
    명령문 2;
    ... ..
    증감식
}
```

- while문과 for문의 비교

- while문은 조건식을 검사하여 참이면 명령을 수행하고, for문과 같이 증감 식으로 변수값을 증감시킴. 그리고 다시 조건식과 비교해 특정한 명령을 반복 수행

## [while문]

```
초기화; ①
while(조건식) ②
{
    ③ 명령문 1;
    ④ 증감식
}
```

## [for문]

```
for(초기화; 조건식; 증감식)
{
    ①    ②    ④
    ③ 명령문 1;
    명령문 2;
    ... ..
}
```

# while문

- while문과 for문의 초기화 위치 비교
  - 일반적으로 for문은 괄호 안에서 초기화, while문에서는 외부에서 초기화함

for문에서 초기화하는 위치

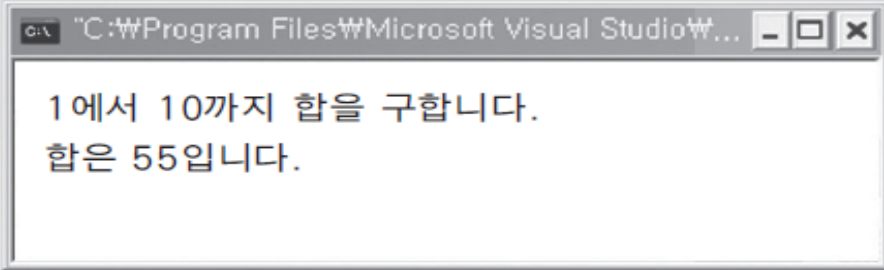
일반적인 for문	외부에서 처리한 경우
<pre>int i, sum = 0; for(i=1; i&lt;=10; i++);     sum += i;</pre>	<pre>int i, sum = 0; i=1; for( ; i&lt;=10; i++)     sum += i;</pre>

- while문과 같이 외부에서 초기화해도 관계없으나 소스코드에서 알 수 있듯 불편해 보임

# while문

예제) 1에서 10까지 합을 출력

```
01 #include <stdio.h>
02 int main(void)
03 {
04     int sum = 0, number = 1;
05     printf("1에서 10까지 합을 구합니다.\n");
06
07     while(number <= 10)
08     {
09         sum += number;
10         number++;
11     }
12
13     printf("합은 %d입니다.\n", sum);
14     return 0;
15 }
```



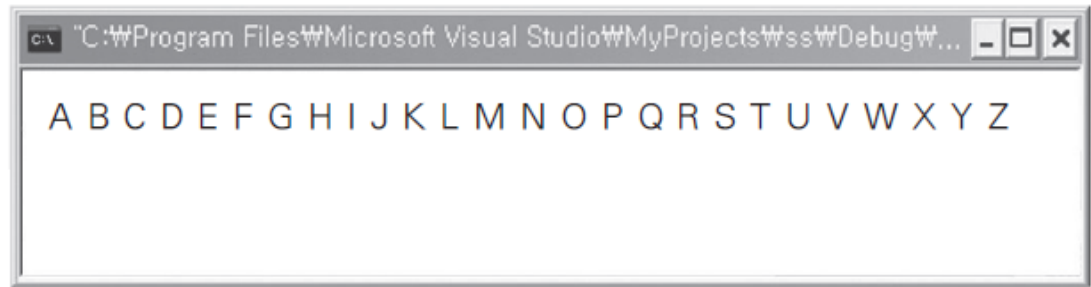
"C:\Program Files\Microsoft Visual Studio\... - [ ] X

1에서 10까지 합을 구합니다.  
합은 55입니다.

# while문

예제) 알파벳을 순서대로 보여주는 예제

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int i = 65;
06
07     while(i <= 90)
08     {
09         printf("%c ", i);
10         i++;
11     }
12
13     printf("\n");
14     return 0;
15 }
```





# while문

예제) 1부터 더한 값이 최초로 300 이상이 되는 a를 구하는 예제

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int a = 1, sum = 0;
06
07     while(sum < 300)
08     {
09         sum += a;
10         a++;
11     }
12
13     printf("합이 300이 넘는 a의 최솟값은 %d다. \n", a-1);
14     printf("sum = %d\n", sum);
15
16     return 0;
17 }
```

C:\Program Files\Microsoft Visual Studio\Debug\...

합이 300이 넘는 a의 최솟값은 24다.  
sum = 300



# while문

- 무한반복

종료되지 않는 반복문을 무한반복(Infinite Loop)이라고 한다. 일반적으로 어떤 조건을 사용하여 조건에 맞으면 반복을 시키지만 경우에 따라서는 무한반복을 사용하기도 한다. for문과 while문에서 무한반복을 사용하는 방법은 다음과 같다.

## [for문]

```
for( ; ; ){  
    문장들...  
}
```

## [while문]

```
while(1){  
    문장들...  
}
```

while문에서 while(1)처럼 조건식에 값 1을 주어 조건이 항상 참이라는 것을 표현한다. 이와 같이 프로그래밍에서는 값이 0인 경우에 거짓(FALSE)이고 그 외에 값이 있는 경우는 참(TRUE)이다. 위의 경우는 while(2)라고 해도 무한반복한다. 그러나 일반적으로 while(1)이라는 표현을 더 많이 쓴다.

무한반복에서 빠져 나오려면, break문을 이용하거나 혹은 **[Shift] + [F5]** 키로 디버깅을 강제로 종료한다.