**<윤성우의 열혈 C 프로그래밍>**

(chap 14 : 포인터와 함수에 대한 이해)

* 간단 개념

Call-by-value : 함수를 호출할 때 단순히 값을 전달하는 형태의 함수호출.

Call-by-reference : 메모리의 접근에 사용되는 주소 값을 전달하는 형태의 함수호출.

.

1. 문제1

변수 num에 값을 저장하고 저장된 값의 제곱을 계산하는 함수를 정의하고, 이를 호출하는 main 함수를 작성해보자. 단, 여기서는 다음 두 가지 형태로 함수를 정의해야 한다.

* Call-by-value 기반의 SquareByValue 함수.
* Call-by-reference 기반의 SquareByReference 함수.

SquareByValue 함수는 인자로 전달된 값의 제곱을 반환해야 하며, SquareByReference 함수는 정수가 저장되어 있는 변수의 주소 값을 인자로 받아서 해당 변수에 저장된 값의 제곱을 그 변수에 다시 저장해야 한다.

풀이 : Call-by-value 기반의 함수는 단순하게 return 값에 num\*num을 하는 방식으로 해결 할 수 있고, Call-by-Reference 기반의 함수는 포인터 함수를 이용해서 값 자체를 바꾸는 방식을 사용해서 해결하면 된다. 여기서 주의할 점은 포인터 함수를 사용하기 위해 main에서 함수를 호출할 때 &을 사용해야 한다. 즉, SquareByReference(&num); 이라고 작성해야 한다.

1. 문제2

세 변수에 값을 저장하고 저장된 값을 서로 뒤바꾸는 함수를 정의해보자. 예를 들어서 함수의 이름이 Swap3라하면, 다음의 형태로 함수가 호출되어야 한다.

* Swap3(&num1, &num2, &num3);

그리고 함수호출의 결과로 num1에 저장된 값은 num2에, num2에 저장된 값은 num3에, 그리고 num3에 저장된 값은 num1에 저장되어야 한다.

풀이 : 이 문제도 포인터 함수를 이용해서 입력된 세 변수 num1, num2, num3를 포인터 함수를 이용해서 불러와야 한다. 그 이후에는 Swap3 함수 내에 임시로 값을 저장할 temp 변수를 선언한 후 값을 바꿔주는 형식으로 풀이하였다.