

객체지향프로그래밍

Assignment Report 1-1

문제 1

이름, 학번, 전공을 입력받고 세 입력이 모두 입력되면 전공, 학번, 이름 순으로 출력하는 프로그램 만들기(이름은 char형으로 학번은 int형으로 자료형을 선언하면 좋을 듯 하다.)

```
이름 : 이강현  
학번 : 2019202050  
전공 : 컴퓨터정보공학부
```

입력

```
이름 : 이강현  
학번 : 2019202050  
전공 : 컴퓨터정보공학부
```

결과

고찰: 메인 함수내에서 모두 처리하지 않고 입력받고 출력해주는 함수를 선언 및 정의해두고 코드를 완성했다면 더 효율적이지 않았을까 생각한다.

문제 2

2개의 수를 입력받아 두 수의 합을 출력한다. 두 수의 범위는 0보다 크고 3×10^9 보다 작은 수 (long long 자료형을 사용하면 범위내의 값들을 모두 출력할 수 있을 것 같다.)이며 모든 수에 대하여 올바른 결과가 나와야 한다. 덧셈은 main함수가 아닌 직접 만든 함수 내에서 이루어진다.

(덧셈연산 후 반환값을 내놓는 함수를 선언하면 될 것 같다.)

```
Microsoft Visual Studio 디버그 콘솔  
Enter Two Numbers : 90 -90  
Enter Two Numbers : 09 3000000000  
Enter Two Numbers : 2999999999 90  
30000000089
```

실행결과

고찰: 문제에서 이미 입력값의 범위를 주었지만 그 외의 범위가 주어졌을 때 입력값을 다시 받도록 코드를 구성해보았다. Unsigned를 사용할 순 없을까라는 생각도 해보았지만 unsigned 변수를 사용하면 음수를 입력받았을 때 부호비트를 고려하지 않은 연산이 이루어져 이상한 값을 얻게 되었다.

문제 3

2개의 수를 입력 받아 두 수의 덧셈, 뺄셈, 곱셈, 나눗셈, 나머지 연산의 결과를 출력하는 프로그램을 구현한다. 단, 나눗셈의 결과는 소수점까지 출력되어야 한다.(float형 자료형을 이용하면 나눗셈을 했을 때 소수점까지 구할 수 있을 것 같다.)

<pre>Enter Two Numbers :10 11</pre>	<pre>Enter Two Numbers :10 11 10 + 11: 21 10 - 11: -1 10 * 11: 110 10 / 11: 0.909091 10 % 11: 10</pre>	입력	결과
-------------------------------------	--	----	----

고찰: 덧셈, 뺄셈, 곱셈, 나눗셈, 나머지를 출력하는 과정에서 단순 반복이 많이 사용되었는데 이를 함수를 이용했다면 조금 더 간단하고 명료하게 코드를 짤 수 있었을 것 같다.

문제 4

1개의 수를 입력 받아 그 수의 제곱과 세제곱을 출력한다, 이때 제곱과 세제곱의 연산은 직접 만든 return이 없는(반환값이 없다면 값의 참조를 통해 함수에서 접근해야 겠다.) 함수내에서 연산되어야 하며 결과는 main함수 내에서 출력되어야 한다.

<pre>Number :6</pre>	<pre>Number :6 Number^2 : 36 Number^3 : 216</pre>	입력	결과
----------------------	---	----	----

고찰: 함수 내에서 제공한 값을 다시 변수에 저장하니 세제곱을 할때도 입력받은 변수값과 달라져 있어 어려움을 겪었다. 하지만 변수 하나를 더 선언해 함수를 사용하고 출력까지 완료되었을 때 다시 처음 입력받은 값으로 초기화 시켜주는 과정을 거치니 쉽게 해결되었다.