

과제: 객체지향프로그래밍실습 (목 1,2) Assignment 1-2

학부: 컴퓨터정보공학부

학번: 2019202021

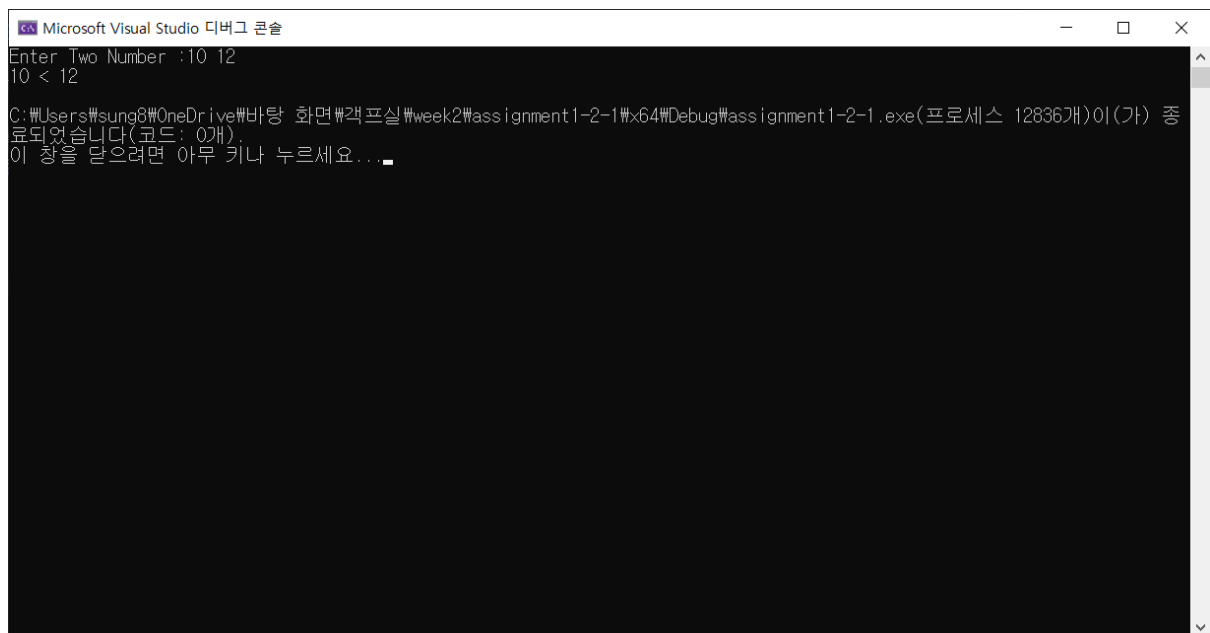
이름 정성엽

<1번 문제>

문제 설명

- 2개의 수를 입력 받고 두 수의 크기를 구분하는 프로그램으로 if문, if else문을 이용하여 조건에 크기 비교를 위해 '<', '>' 비교연산자를 사용한다. 두 수가 같은 경우는 else문을 이용하여 처리한다. A가 B보다 클 경우 'A>B'를, 같을 경우 'A=B'를, B가 A보다 클 경우 'A<B'를 출력해야 한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
Enter Two Number :10 12
10 < 12
C:\Users\sung8\OneDrive\바탕 화면\객체지향프로그래밍실습\week2\assignment1-2\Debug\assignment1-2-1.exe(프로세스 12836개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요....
```

두 수 10과 12를 입력했을 때 결과

고찰

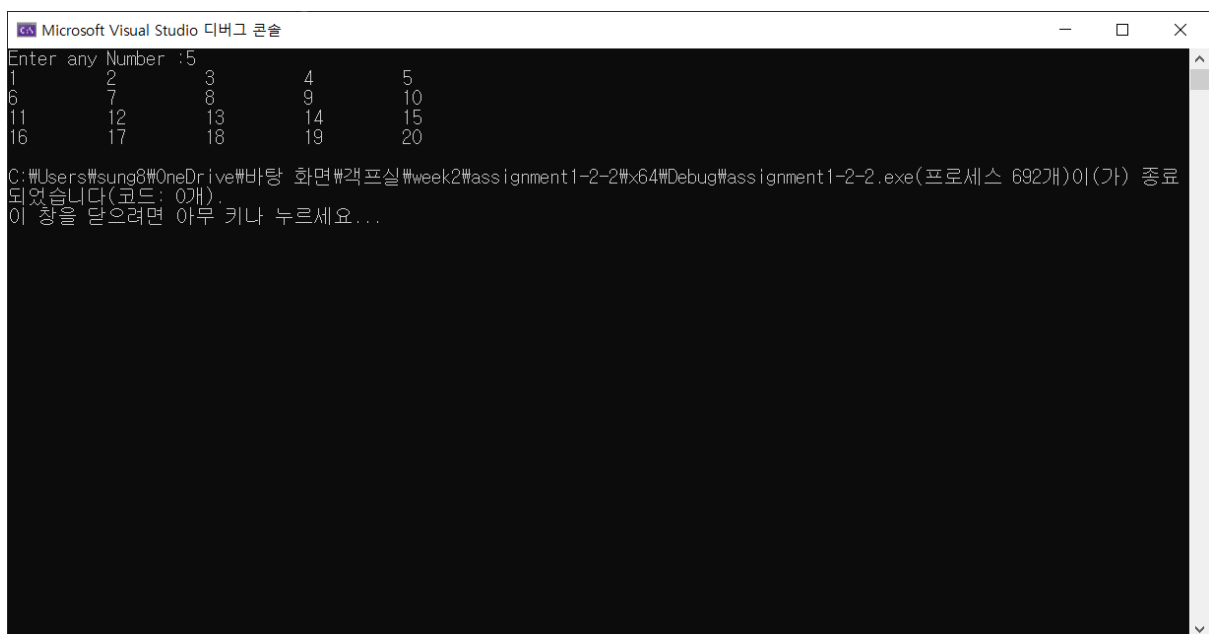
- 비교연산자 '<', '>' 외에도 '=='를 이용하여 비교를 하려 생각했으나 다시 생각해보니 결과는 무조건 크다, 작다, 같다 3가지로 2가지만 if와 if else 문을 이용하면 나머지는 무조건 else로 처리 가능하다고 생각되어 '=='연산자를 사용하지 않았다.

<2번문제>

문제설명

- 숫자 N을 입력 받아 1부터 20까지 N번 째 수에서 줄을 바꾸며 출력하는 프로그램으로 일정 간격을 유지해야 하므로 'iomanip' 헤더를 이용하며 'setw'를 이용하고, 일정 간격을 std::left를 이용하여 왼쪽 들여쓰기를 한다. N개마다 다음줄로 가기 위해 if문을 이용하여 해당 숫자를 N로 나누었을 때 나머지가 0인 경우에 "\n"이 출력되도록 한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
Enter any Number :5
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20

C:\Users\#sung8\OneDrive\바탕 화면\#객프실\week2\assignment1-2-2\#x64\Debug\assignment1-2-2.exe(프로세스 692개)이(가) 종료
되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

양의 정수 5를 넣었을 때 결과

고찰

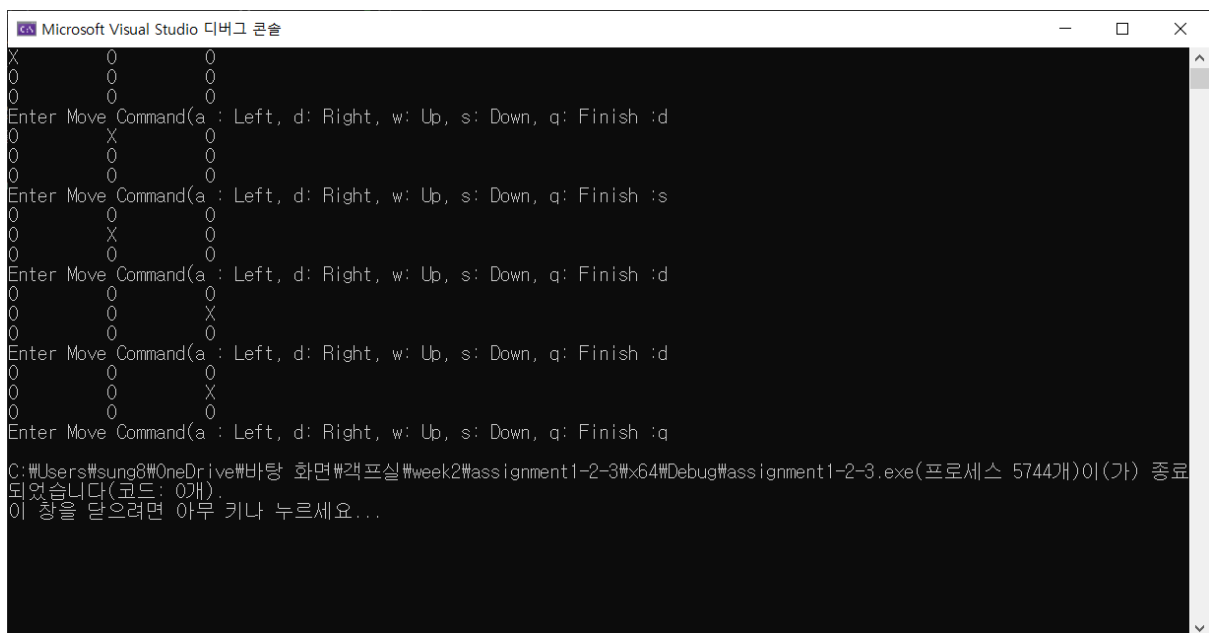
- 처음엔 일정 간격을 유지하는 것을 'wt'으로 처리하려 했으나 다른 헤더 파일을 사용해 보기 위해 원하는 간격만큼 유지할 수 있는 'iomanip'헤더의 'setw'를 이용해 보았다. 그리고 다른 변수로 카운트를 세서 일정 카운트마다 'wn'을 출력하기 보다 입력 받은 N을 나눴을 때 나머지가 0이 될 때 'wn'을 출력하도록 했다.

<문제 3번>

문제설명

- 3x3 보드에서 command를 입력 받아 node가 움직이는 프로그램으로 보드는 'O'로 채워져 있고 node는 'x'로 표기한다. 2차원 배열을 이용하여 마치 3x3보드를 구성한다. 출력할 때 일정 간격을 위해 iomanip 헤더의 setw를 이용하고, 왼쪽 들여쓰기를 위해 std::left를 이용한다. while문을 이용하여 무한 루프를 이루며 이를 탈출하는 것은 'q' 입력이다. 나머지 'w', 'a', 's', 'd'를 입력 받을 때 각각 명령어에 맞춰 좌표값을 더하거나 빼서 이동하며 3x3을 넘어갈 경우 보드를 다시 출력하고 다시 입력 받을 수 있도록 if문을 이용하여 특정 조건을 설계한다.

결과화면



```
Microsoft Visual Studio 디버깅 콘솔
X      O      O
O      O      O
O      O      O
Enter Move Command(a : Left, d: Right, w: Up, s: Down, q: Finish :d
O      X      O
O      O      O
O      O      O
Enter Move Command(a : Left, d: Right, w: Up, s: Down, q: Finish :s
O      O      O
O      X      O
O      O      O
Enter Move Command(a : Left, d: Right, w: Up, s: Down, q: Finish :d
O      O      X
O      O      O
Enter Move Command(a : Left, d: Right, w: Up, s: Down, q: Finish :d
O      O      X
O      O      O
Enter Move Command(a : Left, d: Right, w: Up, s: Down, q: Finish :q
C:\Users\sung8\OneDrive\바탕 화면\각프실\week2\assignment1-2-3\x64\Debug\assignment1-2-3.exe(프로세스 5744개)이(가) 종료
되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

명령어를 통해 node 이동 결과 및 보드를 넘었을 때 출력 결과

고찰

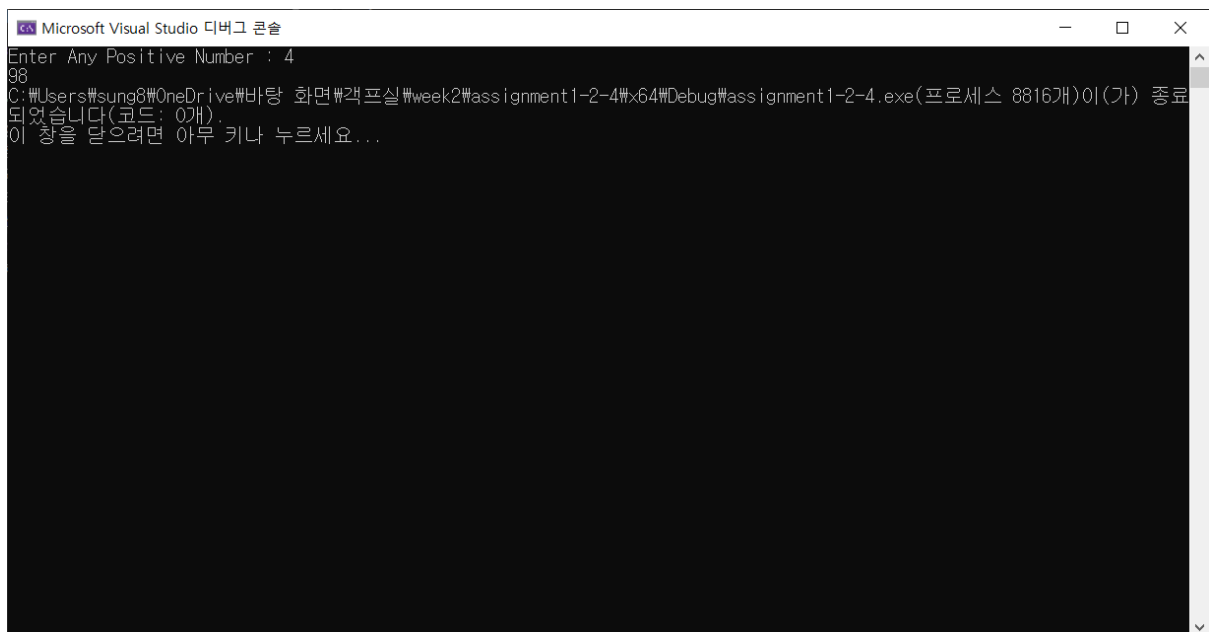
- 2번 문제와 마찬가지로 'setw'를 이용하여 일정 간격을 유지했으며, 3x3보드를 넘어간다고 판단하는 것은 일정 숫자를 넘었을 때 다시 빼거나 더하는 것이 아닌 해당 위치 정보에 따라서 명령어가 실행되는지 안 되는지를 정하여 프로그래밍하였다. 추가적으로 'q', 'w', 'a', 's', 'd' 외에 다른 명령어가 들어 갈 때는 보드를 다시 출력하고 다시 입력 받도록 하였다.

<문제 4번>

문제설명

- 양의 정수 N을 입력 받고 $\sum_{i=0}^N (i * 2^i)$ 를 계산하는 프로그램으로 for문과 while문과 같이 반복문을 사용하지 않아야 하므로 Recursion을 활용하여 하나의 함수 안에서 본인을 반복 호출하여 문제를 풀게 한다. 일정 조건이 만족되면 Recursion을 탈출하여 원하는 값을 return하여 값을 출력한다.

결과화면



```
Microsoft Visual Studio 디버그 콘솔
Enter Any Positive Number : 4
98
C:\Users\sung8\OneDrive\바탕 화면\각 프실\week2\assignment1-2-4\64\Debug\assignment1-2-4.exe(프로세스 8816개)이(가) 종료
되었습니다(코드: 0x#).
이 창을 닫으려면 아무 키나 누르세요...
```

양의 정수 4를 넣었을 때 결과

고찰

- 제공은 *를 여러 번 하여 표현할 수도 있지만 'cmath' 헤더의 'pow'를 이용하여 계산하여 표현하였다. Recursion을 이용할 땐 함수에 입력 받은 값을 1씩 줄이다 0이 될 때 return에서 함수를 호출하지 않고 0을 반환하게 하여 탈출하도록 했다.