

다음은 Visual Studio 2010의 MSDN에 나와 있는 키워드 friend에 대한 설명이다.

*In some circumstances, it is more convenient to grant member-level access to functions that are not members of a class or to all functions in a separate class. The friend keyword allows a function or class to gain access to the private and protected members of a class. You can declare friend functions or friend classes to access not only public members but also protected and private class members.*

friend를 꼭 사용하여야 하는 예로는 연산자 오버로딩을 들 수 있다. 연산자 오버로딩이란 연산자의 의미를 재정의하는 것이다. 연산자를 재정의 하기 위해서는 정해진 형식을 이용하여야 한다. 그 형식은 함수의 형식과 유사하며 다음과 같다.

*type operator operator-symbol ( parameter-list )*

예를 들어 자료형 T에 대한 +연산자를 재정의하기 위해서는 다음과 같이 쓴다.

```
T operator+( T a, T b );    // 선언
```

```
T operator+( T a, T b ) {    // 정의
    return T타입 객체;
};
```

프로그램에서는 일반 연산자를 이용하듯 프로그램하면 된다.

**[문제]** 다음의 class Point가 main()과 같이 실행될 수 있도록 friend를 사용하여 연산자를 오버로딩하라. (20)

```
#include <iostream>
using namespace std;
```

```
class Point {
    int x, y;
public:
    Point( int x, int y ) : x( x ), y( y ) { }

    void show() {
        cout << "[" << x << ", "
              << y << "]" << endl;
    }
};

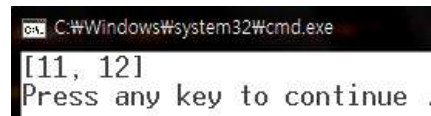
//-----
int main()
//-----
{
    Point a( 1, 2 );

    Point c = 10 + a; // a의 값에 상수를 더 함

    c.show();
}
```

**[힌트]** `Point c = 10 + a;`  
이 문장의 오른쪽 표현식 `10 + a`는 `operator+( 10, a )`라는 함수 호출문장임.

**[실행결과]**



```
C:\Windows\system32\cmd.exe
[11, 12]
Press any key to continue
```

**[답]**