

06. 접근지정자, const가 붙은 변수, static 멤버

## [실습 6-1]

- 컴퓨터의 주기억장치를 모델링하는 클래스 Ram을 구현하려고 한다. Ram 클래스는 데이터가 기록될 메모리 공간과 크기 정보를 가지고, 주어진 주소에 데이터를 기록하고(write), 주어진 주소로부터 데이터를 읽어온다(read). Ram 클래스는 다음과 같이 선언된다.

```
class Ram {  
    char mem[100*1024]; // 100KB 메모리  
    int size;  
public:  
    Ram();    // mem을 0으로 초기화하고 size를 100*1024로 초기화  
    ~Ram();   // "메모리 제거됨" 문자열 출력  
    char read(int address); // address 주소의 메모리를 읽어 리턴  
    void write(int address, char value); // address 주소에 value 저장  
        // 주소가 범위를 벗어나는 오류 발생하면 에러메시지 출력함.  
};
```

## [실습 6-1] 계속

- 다음 main() 함수는 100번지에 20을 저장하고, 101번지에 30을 저장한 후, 100번지와 101번지의 값을 읽고 더하여 102번지에 저장하는 코드이다.

```
int main() {  
    Ram ram;  
  
    ram.write(100, 20); // 100 번지에 20 저장  
    ram.write(101, 30); // 101 번지에 30 저장  
    char res = ram.read(100) + ram.read(101); // 20 + 30 = 50  
    ram.write(102, res); // 102 번지에 50 저장  
  
    cout << "102 번지의 값 = " << (int)ram.read(102) << endl; // 102 번지 메모리 값 출력  
}
```

102번지의 값 = 50  
메모리 제거됨

- 실행결과를 참고하여 Ram.h, Ram.cpp, mail.cpp로 헤더파일과 cpp파일을 분리하여 프로그램을 완성하라.

## [실습 6-2]

- 왼쪽 코드를 static 멤버를 가진 Math 클래스로 작성하고 멤버 함수를 호출하라.

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

int abs(int a) { return a>0?a:-a; }
int max(int a, int b) { return a>b?a:b; }
int min(int a, int b) { return (a>b)?b:a; }

int main() {
    cout << abs(-5) << endl;
    cout << max(10, 8) << endl;
    cout << min(-3, -8) << endl;
}
```

(a) 전역 함수들을 가진 좋지 않음 코딩 사례

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

class Math {
    
};

int main() {
    cout << _____ << endl;
    cout << _____ << endl;
    cout << _____ << endl;
}
```

(b) Math 클래스를 만들고 전역 함수들을 static 멤버로 캡슐화한 프로그램

5  
10  
-8