

```
Microsoft Visual Studio 디버그 콘솔
102 번지의 값 = 50
메모리 제거됨
C:\#c++vs_workspace#C++HomeWork#1#x64#Debug#C++HomeWork#1.exe(프로세스 29888)이(가) 0 코드(0x0)와 함께 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요...
```



소스 구현 설명

-문제 정의 : 간단한 RAM동작을 만드는것이며 문제에는 Ram.h와 main클래스만 나와있으며 Ram.cpp이 구현되어 있지않아 실질적인 Ram에 읽기와 쓰기 동작을 구현하는 문제이다.  
주의할 점으로는 Ram.cpp 동작 구현을 만들때 문제에서 정의된 메모리 이상의 값이나 이하의 값을 넣었을 경우 오류가 뜰 수 있다.

-문제 해결방법 : Ram.cpp에 Ram.h에 정의되어 있는 메소드 Ram(), ~Ram(), char read, void write 을 구현 하면되며  
1. 생성자인 Ram::Ram() 에는 size를 메모리 크기인 100 \* 1024로 정의하여 유효값 검사때 사용할것이다.  
2. Ram::~~Ram() 에는 cout << "메모리 제거됨" << endl; 을 넣어 메모리가 끝날 시 메모리 제거됨을 출력  
3. char Ram::read(int address) 에는 유효값 검사인 조건식 if (address < 0 || address >= size) 을 넣어 주소값 보다 크거나 주소의 최소값 0 보다 작으면 유효하지 않은 주소입니다라고 출력할 것이며 '0'을 반환할 것이다.  
주소값 유효하여 if문이 동작하지 않았을 경우 return mem[address]; 으로 주소값에 맞는 값을 반환할 것이다.  
4. void Ram::write(int address, char value) 에서는 if else 문으로 if (address >= 0 && address < size) 을 통과하여 유효한 값이 경우 mem[address] = value; 을 할당할 것이며 유효하지 않을경우 else문으로 처리하며 "유효하지 않은 값 입니다."를 출력한다.

-아이디어 평가: 간단한 문제이기에 유효값 검사 조건만 잘 적용 한다면 다른 값을 넣어도 오류 없이 정상 작동할것이다.  
실제로 -1 , -2번지 주소에 값을 저장하게 되면 유효하지 않은 값입니다가 나오며 저장되지 않으며 102번지 주소에는 빈값이나온다.  
즉 이번에 문제를 풀며 생각해낸 아이디어는 알맞은거 같으며 결과값 또한 이상적이다|

-문제를 해결한 키 아이디어 또는 알고리즘 설명  
size값에 있는 최대 메모리 값을 이용하여 if와 else문으로 문제를 해결하였으며 if (address < 0 || address >= size) 과 if (address >= 0 && address < size) 을 이용하여 메모리의 최대,최소값을 벗어 나면 오류라고 나오게 해결하였습니다.