

char mem[100*1024]; 은 100KB 메모리를 나타냅니다.

Ram() 함수는 생성자로서 mem 배열을 0으로 초기화하고 size를 100*1024로 초기화하는 함수를 작성합니다.

```
size = 100*1024;
for (int i = 0; i < size; i++) {
    mem[i] = 0;
}
```

~Ram() 함수는 소멸자로서 "메모리 제거"를 출력 하도록 함수를 작성합니다.

```
cout << "메모리 제거" << endl;
```

char read(int address) 함수는 주어진 주소의 값을 읽어올 수 있도록 address 주소의 메모리 바이트 리턴을 할 수 있도록 리턴을 mem[address]로 작성합니다.

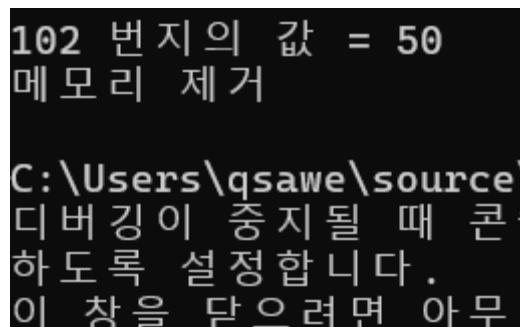
void write(int address, char value) 함수는 address 주소에 한 바이트로 value를 저장할 수 있도록 함수를 작성합니다.

main함수 같은 경우에는 문제에 작성된 코드를 그대로 사용합니다.

코드를 헤더파일, Ram.cpp, main.cpp로 분리하기 위해 선언부는 헤더파일로, 클래스 구현은 Ram.cpp파일로, 메인함수는 main.cpp로 분리하였습니다.

또한, 헤더파일에 쓰인 #ifndef Mem_H, #define Mem_H, #endif의 경우에는 헤더파일의 중복 include 문제를 해결하기 위해 사용하였습니다.

출력



```
102 번째의 값 = 50
메모리 제거

C:\Users\qsawe\source
디버깅이 중지될 때 콘
하도록 설정합니다.
이 창을 닫으려면 아무
```