

```
Microsoft Visual Studio 디버그 콘솔
10개점수입력:10 20 30 40 50 60 70 80 90 100
60점 이상은4명
C:\c++vs workspace\c++HomeWork#1\hw\hw#3-(1)\x64\Debug\hw#3-(1).exe(프로세스 4060)이(가) 0 코드(0x0)와 함께 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요...
```

```
Microsoft Visual Studio 디버그 콘솔
10개점수입력:20 30 40 50 60 70 80 90 100 110
60점 이상은 5명
C:\vs\workspace\C++\HomeWork#1\HW\hw#3-(3)\x64\Debug\hw#3-(3).exe(프로세스 23280)이(가) 0 코드(0x0)와 함께 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요...
```



소스 구현 설명

-문제 정의 : Dept 클래스, countPass, main()이 정의되어 있으며 (1)번 문제는 Dept 클래스멤버주 복사 생성자, 소멸자, void read() 함수가 정의되어 있지 않기 때문에 이를 (1)번 문제에 나온 실행 결과에 맞게끔 Dept 클래스 멤버를 정의하는 것이다.
(2)번 문제는 Dept클래스에 있는 복사 생성자를 지운다음 실행하면 오류가 뜨게 되는데 countPass(com) 호출시 com객체는 값을 전달하는 것이기 때문에 복사 생성자가 호출이 되는데 이 때 기본 복사생성자가 실행되기에 scores의 포인터 주소가 기존 com의 주소와 동일하기에 countPass 호출이 끝난후 소멸자가 호출되어 scores메모리가 해제되며 기존 com객체의 scoers 메모리도 같이 해제되기 때문에 이를 해결하는 문제이다]

-문제 해결방법 :
(1)문제-1 : 복사생성자, 소멸자, void read()함수를 정의하면된다.
(1)문제-2 : 복사생성자에서 동적할당을 받는 배열 scores에 대해서 복사를 진행하면 되며 원본 객체 size는 dept.size를 통해 가져오며 scores 는 새로운 메모리할당을 통해 새로운 메모리를 지정해 주고 for문을 통해 새로운 scores에 원본 객체의 성적을 복사해주면된다.
(1)문제-3 : 소멸자는 동적할당된 메모리를 해제하면 되기에 동적할당된 메모리는 배열인 scores 하나 밖에 없기에 delete[] scores 를통해서 제거한다.
(1)문제-4 : read()는 호출되었을때 size 만큼 키보드에서 정수를 읽어 scores 배열에 저장을 해야되기에 cout << "10개점수입력:.";를 사용하여 한번에 10개의 점수를 입력하도록 유도한뒤 for문과 cin을 통해 입력을 받으면 된다.

(2)문제-1: 복사 생성자를 삭제하게 되면 새로운 com이 아닌 기존 com의 scores메모리가 소멸자에 의해 삭제 되기 때문에 기존 com의 scores 메모리가 없어질때 이미 삭제된 상태여서 오류가 뜬다.
(2)문제-2: 객체의 주소가 현재 값으로 전달되기에 복사생성자가 실행되는데 이를 참조로 전달하여 해결하면된다.
(2)문제-3: countPass 함수의 매개변수 타입을 const Dept& dept 로 변경하며 참조로 전달한다. const로 만들어진 객체가 만들어 지는 것이기 때문에 이를 통해 호출되는 dept.getSize dept.isOver60 또한 const 를 추가하여 함수를 만들어야 한다.
처음에는 const Dept& dept 만 변경하여 만들어 실행해 보았으나 다른 함수가 호출이 안되는 문제가 발생하여 찾아보니 const로 만들어진 객체는 const로 만들어진 함수만 호출 할 수 있으며 일반 함수는 const함수도 호출 할 수 있다는것을 알아서 수정했습니다.

-아이디어 평가: (1)번 문제를 풀때 복사 생성자를 제대로 이해하여 잘 완성 하였으며 소멸자를 이용해 동적할당된 메모리를 해제하였다. read()함수는 문제에 제시한 대로 작동하게끔 잘 만들어서 (1)번 문제는 복사생성자의 이해, 소멸자에 대한 이해를 필요로 했던 문제였으며 이를 잘 해결 하였다.
(2) 문제는 깊은 복사 생성자가 실행되는 조건에 대해서 알아야 했으며 이를 잘 파악하였고 const의 특성에 대해 몰랐던 것을 찾아 잘 해결 하였습니다.

-문제를 해결한 키 아이디어 또는 알고리즘 설명:
(1)번 문제는 복사생성자에 동적할당을 받는 배열에 대해 새로운 메모리 할당을 통하여 복사를 진행 하였으며 원본 객체 size는 그대로 가져왔습니다.
(2)번 문제는 복사 생성자의 조건에 대해 사전 지식이 있어야 했으며 const의 특성에 const로 만들어진 객체는 const로 만들어진 함수만 호출 할 수 있으며 일반 함수는 const함수도 호출 할 수 있다 라는 특성을 활용하여 나머지 함수에도 const를 붙여야 했습니다.