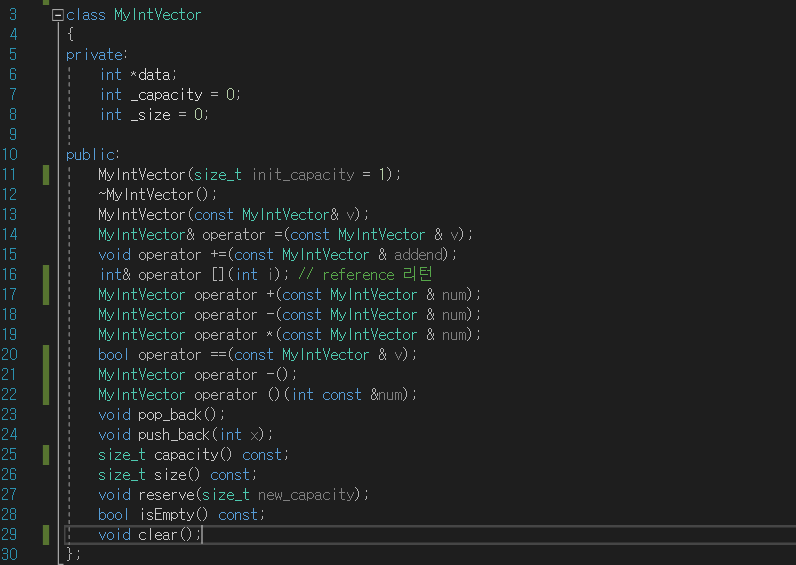
자료구조 1차 과제

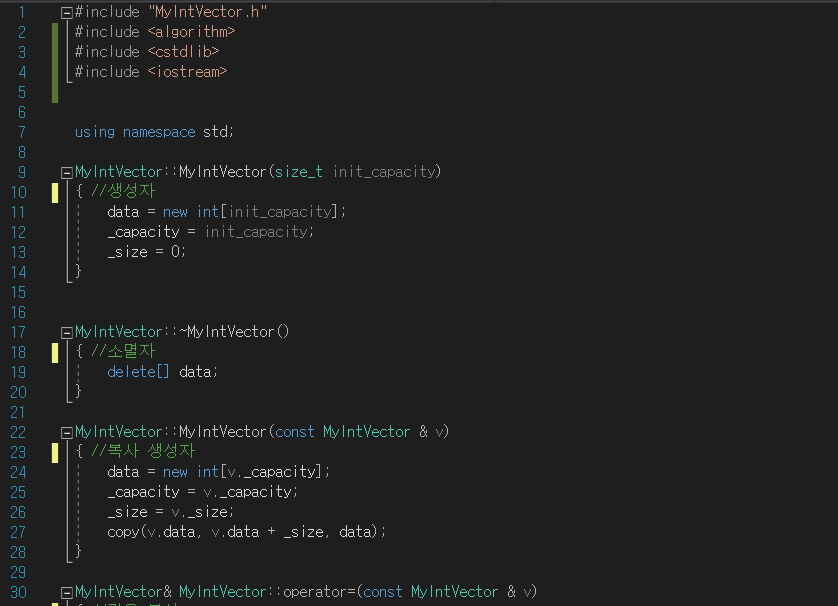
김용혁 교수님 (월5, 수6 수업)

소프트웨어학부

2017803094

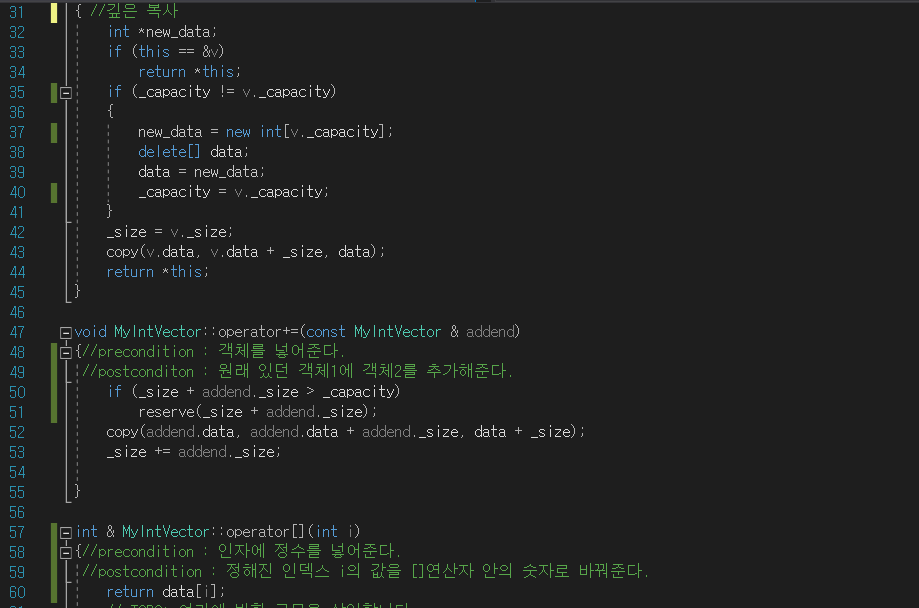
최지훈

우선 MyIntVector.h를 정의했습니다. 멤버함수와 멤버변수는 교수님께서 정의하신 그대로 이용하여 작성했습니다.

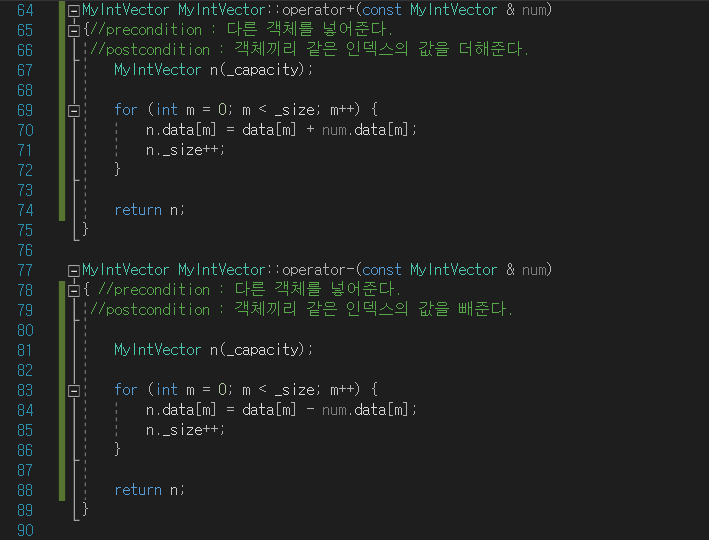


MyIntVector.cpp 파일입니다.

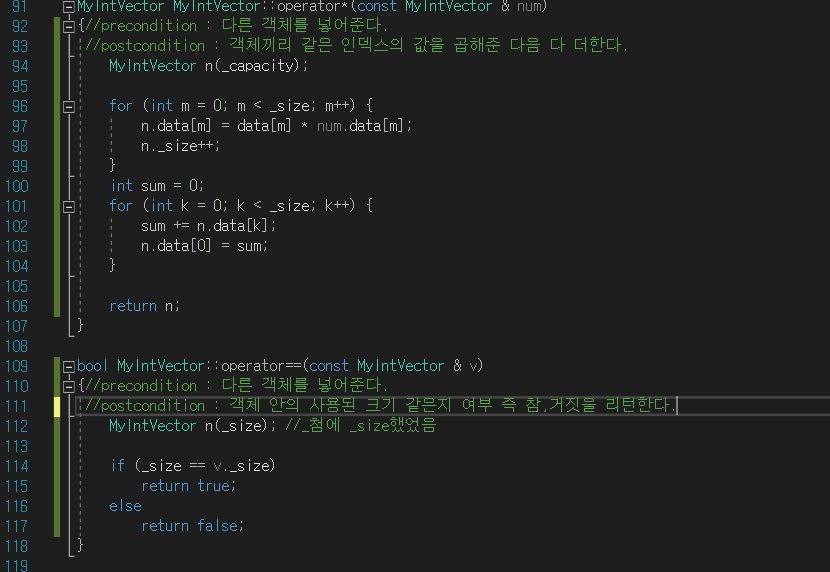
생성자와 소멸자 그리고 복사생성자를 정의해 주었습니다.



깊은 복사를 위해 = 연산자 오버로딩을 하였습니다. 강의자료를 토대로 했습니다.

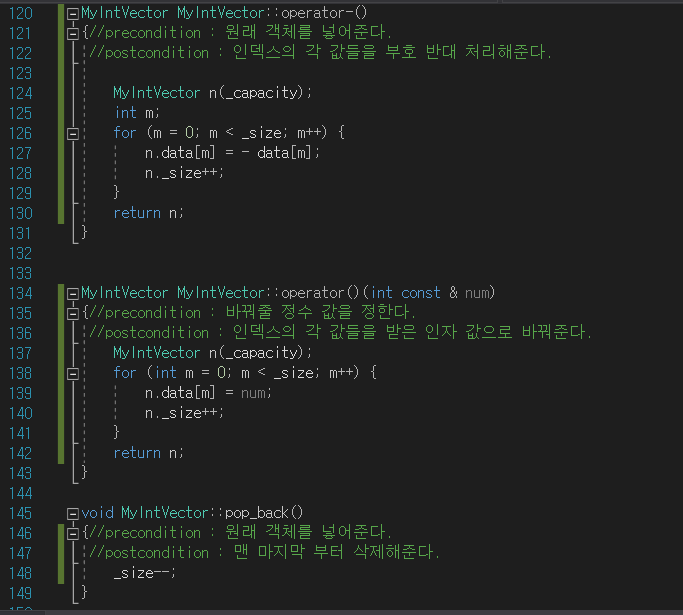


이항 연산자 +와 -를 오버로딩 했습니다. Size를 하나 더 늘리기 전에는 값이 저장되지 않고 쓰레기 값이 호출되었습니다. 거의 하루 종일 고민을 하면서 다양한 시도를 했다가 결국에는 해결을 했습니다. Size를 하나 더 늘려야 하는 것이었습니다. 혼자서 곰곰이 생각해보니 size를 늘려야 더한 결괏값을 저장해준다고 생각을 했습니다. 실제로 맞는지는 모르겠습니다. 질문해보겠습니다.



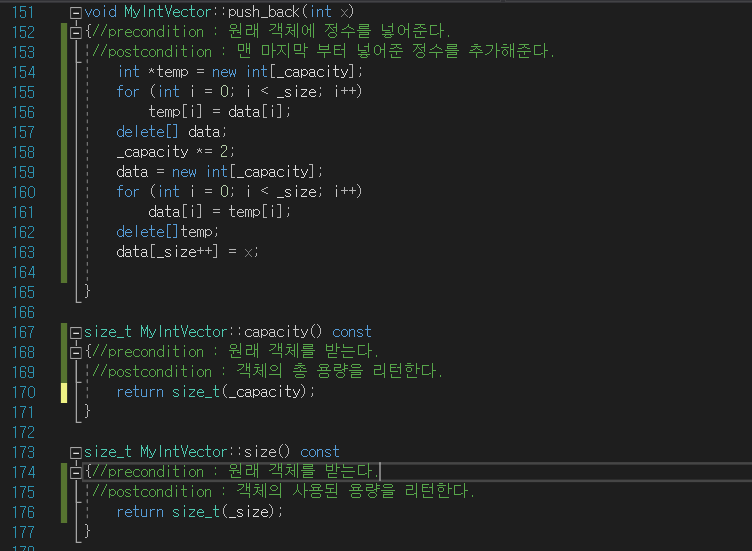
\*연산자와 == 연산자 오버로딩 했습니다. \* 연산자는 각각 벡터 인덱스가 같은 것들을 곱해줘서 마지막에 다 더하기 때문에 for문을 두 번 썼습니다. 인덱스가 같은 것들끼리 한 번 곱해주는 연산을 하기 위해 첫 번째 for문을, 그 다음에 새로운 객체에 저장해 주었기 때문에 곱해진 값들을 다 더해주는 두 번째 for문을 사용했습니다.

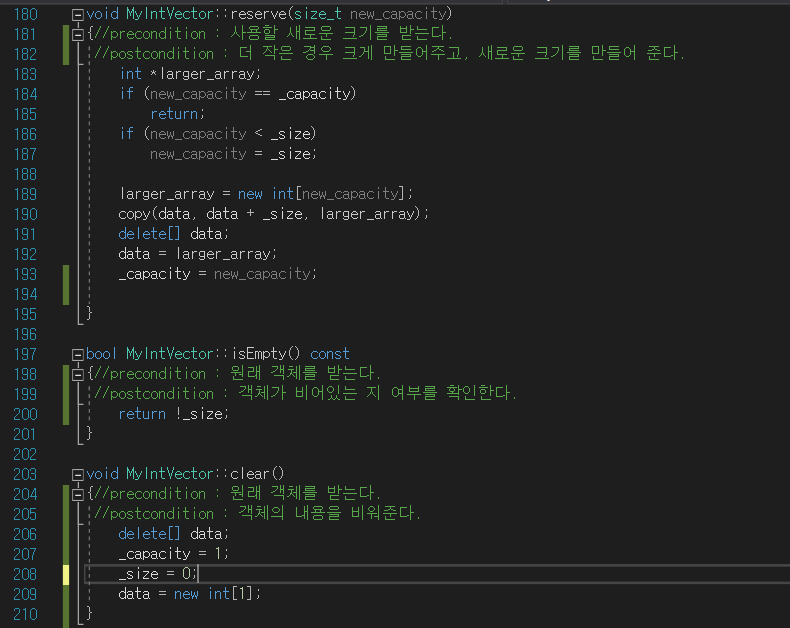
그 다음으로 == 연산자는 두 객체의 사용된 크기가 같은지를 확인해주는 연산자였습니다.



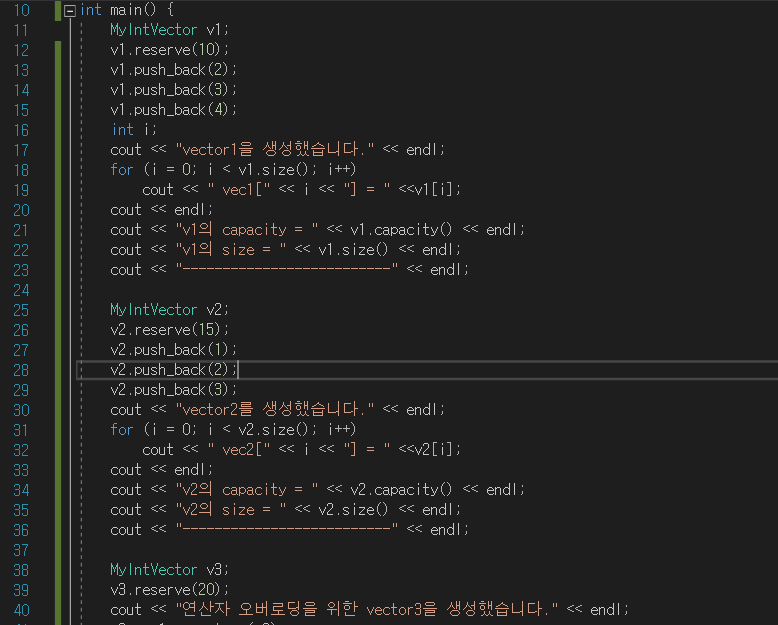
단항 연산자 -를 만들어 주었습니다. 위 이항 연산자들과는 다르게 자신의 값들의 부호를 바꾸어 주기에 인자를 받지 않습니다.

()연산자는 인자 안에 넣어준 숫자로 값으로 벡터 안의 값들을 다 바꾸어 주는 연산자 입니다.

push\_back과 capacity 그리고 size함수를 만들었습니다.

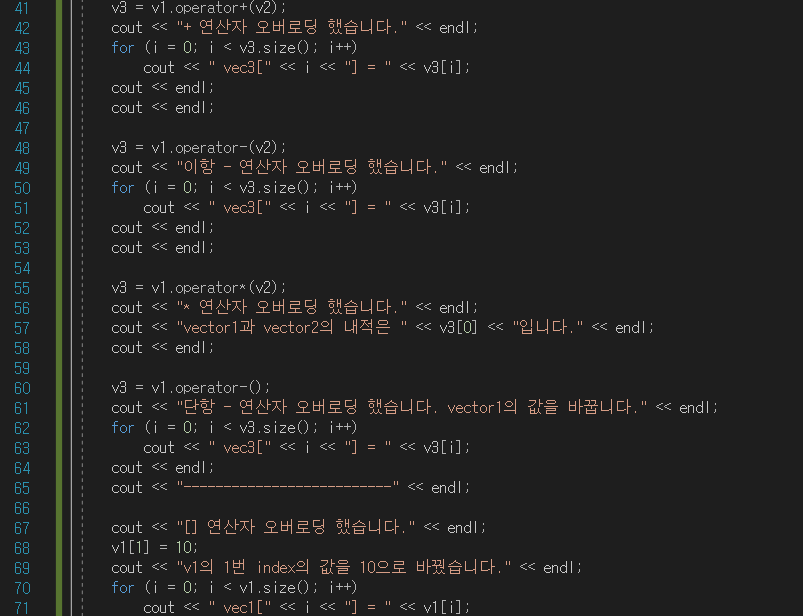


새로 만들어 주는 reserve 함수와 비어있는지 여부를 확인해주는 isEmpty 함수와 비어지게 만드는 clear 함수를 만들었습니다.

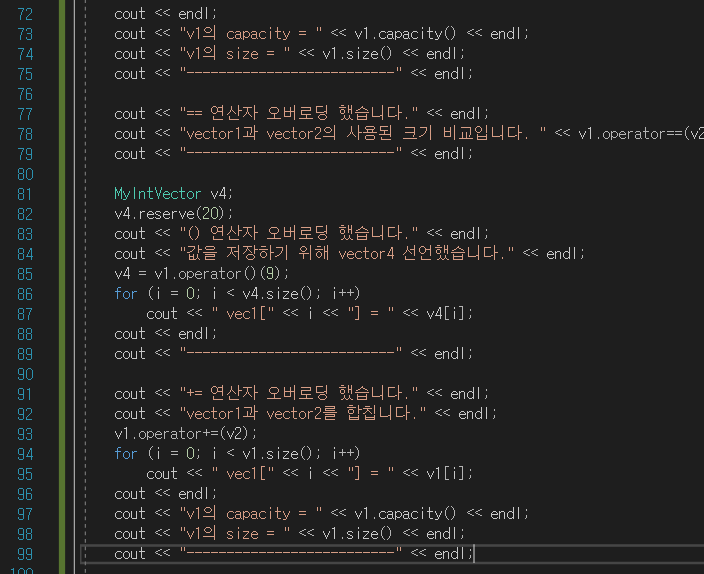


Main 함수입니다.

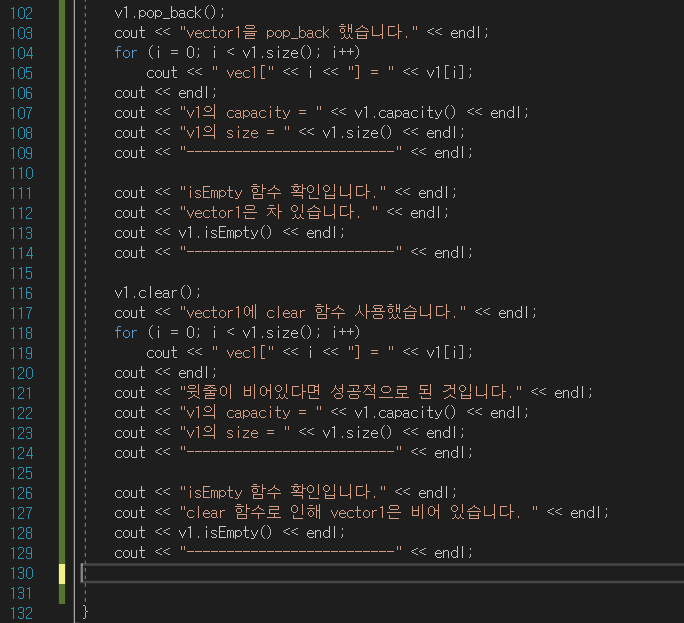
v1 벡터를 만들고 값을 push\_back으로 넣어 주었습니다. v2 벡터도 만들고 같은 방법을 사용해 주었습니다. 그 후 연산자 오버로딩을 검사 및 실행하기 위해 v3 벡터를 새로 만들어 주었습니다.



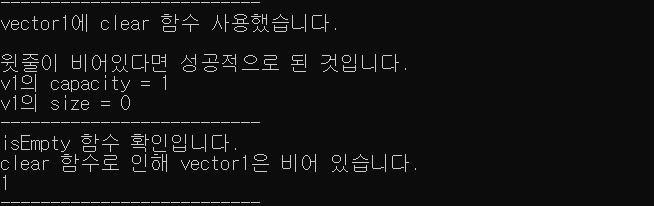
이항 연산자 +, -, \* 를 오버로딩 해주고 단항 연산자 – 도 오버로딩해주고 []로 해당 인덱스의 값이 바뀜을 보여주었습니다.



== 연산자 오버로딩으로 참, 거짓 값을 받아왔습니다. () 연산자 오버로딩도 했습니다. 아마 결괏값으로 모두 9로 바뀌겠죠? 그리고 += 연산자로 v1에 v1과 v2를 합쳤습니다.



합친 v1 객체를 pop\_back 했습니다. 하나가 줄어들 것입니다. 그리고 isEmpty로 객체가 비어있는지 여부를 확인했습니다. 현재는 비어있지 않으므로 false 값인 0이 호출될 것입니다. 그 다음에 clear 함수를 사용해 객체를 비웠습니다. 비우고 다시 한 번 확인하기 위해 isEmpty 함수를 사용했습니다. 비어있기에 이번엔 true인 1이 호출될 것입니다.



실행창입니다.

고찰

우선 하면서 든 생각은 정말 어려웠다 입니다. 개인적으로 C++에 대한 개념이 확립이 잘 안 되어 있어서 특히 생성자나 연산자 오버로딩에서 많은 난항을 겪었습니다. 자료구조 수업 때 배운 얄팍한 C++ 지식과 개인적으로 가지고 있던 C++ 책 그리고 인터넷 검색을 통하여 해결해 나갔습니다. 앞으로는 개인적으로 C++ 공부를 조금은 해야 할 것 같다는 생각이 듭니다. 그 다음으로 든 생각은 미리미리 하자였습니다. 물론 정말 어려운 과제긴 했지만 그리고 또 제가 완벽하게 구현은 한 것은 아니지만 어느정도 얼추 구현은 한 것 같습니다. 조금만 더 신경써서 미리미리 시도하고 질문했다면 나름 완벽에 가까운 결과물을 만들어내지 않을까라는 생각을 했습니다. 하지만 한편으로는 나름 뿌듯하기도 합니다. 저는 전과를 했습니다. 1학기 때 고급프로그래밍 과목이나 리눅스활용 등의 과목에서 많이 힘들어했고 과제를 하는 과정에서 도움을 많이 받았었는데 이번 자료구조 때는 남들의 도움을 받아보지 않으려 혼자서 해봤습니다. 물론 다시 한 번 말하지만 완벽하지는 않습니다. 앞으로 더 깔끔하게 만들 수 있도록 노력하겠습니다.