

2020059525 김유태

기본 설정

int _size_ : vector 원소들이 사용하고 있는 크기. 기본값은 0이다.

int _cap_ : steam의 크기로 _size_가 클 경우에는 _cap_을 _size_보다 크게 설정하고 steam을 temp에 복사후 동적할당 뒤 복사하는 방법으로 활용한다. 기본값은 1이다.

T* memory : 배열을 받는 역할을 한다.(원소 저장)

Vector() : 기본

Vector(int size) : 벡터의 크기를 설정한다.

Vector(int size, T element) : vector의 크기만큼 초기화할 요소를 정한다.

위 vector는 위의 _size_와 _cap_의 기본값을 가진다.

~vector() : steam을 삭제한다. 본 헤더파일에서는 Steam[]을 기본으로 사용한다.

구현

Capacity() : _cap_을 리턴한다.

Size() : _size_를 리턴한다.

Empty() : size()가 0이 아니면 false, 0이면 True 를 리턴한다.

Clear() : vector의 _size_를 0으로 설정한다.

Insert() : _cap_이 부족하면 _cap_을 여유있게 설정하여 temp라는 이름으로 동적할당을 한다. steam에서 temp로 임시로 복사후 steam을 새로운 _cap_에 맞게 동적할당을 하고 temp를 복사한다. Steam[index] = num을 넣고 전후로는 기존의 temp에서 값을 가져온다.

Erase() : steam을 temp에 복사하고 steam[del_num]을 제외한 요소(for문 이용해서 앞부분 과 뒷부분을 각각)를 steam에 복사한다. 그리고 _size_를 1 감소시킨다.

Push_Back() : _size_가 _cap_보다 크거나 같으면 _cap_을 2배 늘려주고(_size_보다 여유 있게 설정한다면 다른 방식도 무관하다.) temp에 임시로 만든후 값을 집어넣고 이를 다시 넓어진 _cap_을 가진 steam에 넣은뒤 steam[_size_]에 입력받은 값을 집어넣는다. 결과적으로 원소 하나가 추가 되었으므로 _size_를 1만큼 증가시켜준다.

Pop_back() : _size_ - 1을 시행한다. Size가 작아져 마지막 원소가 없어진다. Operator[] (int number) : steam[number]를 반환한다.

Resize() : size를 재설정한다.

Operator<< : 입력받은 벡터의 요소들을 순서대로 "/"로 구분하여 출력한다.(std::cin >> vector)

Operator>> : 값을 입력받아 벡터에 push_back한다.(std::cin << vector) (operator <>는 friend를 사용해 class 밖에서 사용가능하다.)

Iterator : 주소값을 받으며 vector::iterator it으로 선언할 수 있으며 포인터를 이용해 주소에 있는 값을 얻을 수 있다.

Operator= same 벡터의 값을 복사해서 steam에 저장한다. _size_와 _cap_도 넘겨받는다.

Assign(int count, const T& newone) : 원소 newone을 count개 가지는 벡터로 만든다. (재할당)

Begin/end : 각각 steam 과 steam+_size_를 리턴한다. 이는 main함수를 통해 간단히 확인해볼 수 있다.