개요 | 제출 | 편집 | 코딩 결과

실습 086 - MyList Excercise (p35)

제출 마감일: 2023-05-17 23:59
업로드 가능한 파일 수: 2
제출 방식: 개인

목적

이 실습은 template 함수를 정의하는 연습을 합니다.

문제

Implement a generic list using templates

//MyListTest.cpp

```
#include "MyList.h"

using namespace std::complex_literals;

int main() {

    MyList<std::complex<double>, 5> complexList;

    std::complex<double> z1 = {0. + 0i}, z2 = {1. + 1i}, z3 = {2. + 2i};
    complexList.add(z1); complexList.add(z2); complexList.add(z3);

    std::vector<std::complex<double>> cTargets = {{0. + 0i}, {1. + 0i}};
    for(auto target : cTargets) {
        auto found = find(complexList.begin(), complexList.end(), target);
        if (found != complexList.end()) std::cout << "Found: " << *found << std::endl;
    }

    MyList<std::string, 5> stringList;
    std::string s1 = "abc", s2 = "def", s3 = "ghi";

    stringList.add(s1); stringList.add(s2); stringList.add(s3);

    std::vector<std::string> sTargets = {"ccc", "abc"};
    for(auto target : sTargets) {
        auto found = find(stringList.begin(), stringList.end(), target);
        if (found != stringList.end()) std::cout << "Found: " << *found << std::endl;
```

```
    }

    complexList.remove(z1);
    for (auto z : complexList) std::cout << z << std::endl;

    stringList.remove(s1);
    for (auto s : stringList) std::cout << s << std::endl;

    MyList<std::string, 5> stringList2(stringList);  // copy constructor
    stringList2.add(s1);
    for (auto s : stringList) std::cout << s << std::endl;
    for (auto s : stringList2) std::cout << s << std::endl;

    MyList<std::string, 5> stringList3;
    stringList3 = stringList2;              // copy assignment
    stringList3.remove(s2);

    for (auto s : stringList3) std::cout << s << std::endl;
    for (auto s : stringList2) std::cout << s << std::endl;
}
```

```
// MyList.h

#include <cstddef>

template<typename T, size_t N>
class MyList {
public:
    MyList();
    MyList(const MyList& rhs) noexcept;
    MyList& operator=(const MyList& rhs);
    ~MyList() noexcept;

    void add(T& data);
    void remove(T& data);

    const T* begin() const {return _data;}
    const T* end() const {return _data + N;}

    T* begin() {return _data;}
    T* end() {return _data + N;}

private:
    T* _data= nullptr;
    int pos = 0;
};
```

입력

없음

출력

```
Found: (0,0)
Found: abc
```

(1,1)
(2,2)
(0,0)
(0,0)
(0,0)
def
ghi

def
ghi

def
ghi
abc

ghi
abc

def
ghi
abc

제출파일

86.csv

MyList.h