Разработать:

·класс «обыкновенная дробь»,

·класс «массив обыкновенных дробей произвольной размерности»,

·конструкторы для инициализации,

·перегруженные операторы потокового ввода/вывода (<< и >>), (в операторе ввода спросить новую длину массива)

·перегруженную \* умножения обыкновенной дроби на число как метода (функцию-член класса),

·перегруженную операцию \*= перемножение элементов двух массивов обыкновенных дробей.

Организовать тестирование всех возможностей программы в функции main()

(Примечание: сокращение дробей не делать)

#include <iostream>

using namespace std;

class Fraction { // дробь

int m /\*числитель\*/, n /\*знаменатель\*/;

public:

Fraction() { // конструктор по умолчанию

m = 1; n = 1;

}

Fraction(int inputM, int inputN) { // конструктор с параметрами

m = inputM; n = inputN;

}

friend istream& operator>>(istream&, Fraction&);

friend ostream& operator<<(ostream&, Fraction&);

Fraction operator\*(int k) {

return Fraction(m\*k, n);

}

// далее - код для функции перемножения массивов

int getM() {

return m;

}

int getN() {

return n;

}

void setM(int inputM) {

m = inputM;

}

void setN(int inputN) {

n = inputN;

}

};

istream& operator>>(istream& is, Fraction& f) {

cout << "Введите числитель: ";

is >> f.m;

cout << "Введите знаменатель: ";

is >> f.n;

return is;

}

ostream& operator<<(ostream& os, Fraction& f) {

os << f.m << "/" << f.n;

return os;

}

class FractionArray { // массив дробей

int size;

Fraction\* elements;

public:

FractionArray() { // конструктор по умолчанию

size = 0;

elements = nullptr; // размер - 0, массив пустой

}

FractionArray(int inputSize) { // конструктор с параметрами

size = inputSize;

elements = new Fraction[size];

}

~FractionArray() {

delete[] elements;

}

friend istream& operator>>(istream&, FractionArray&);

friend ostream& operator<<(ostream&, FractionArray&);

//friend FractionArray& operator\*=(FractionArray&, FractionArray&);

};

/\*FractionArray& operator\*=(FractionArray& fa1, FractionArray& fa2) {

return fa1;

}\*/

istream& operator>>(istream& is, FractionArray& fa) {

for(int i = 0; i < fa.size; i++) {

cout << "Введите элемент №" << i+1 << ":" << endl;

is >> fa.elements[i];

}

return is;

}

ostream& operator<<(ostream& os, FractionArray& fa) {

os << "[ ";

for(int i = 0; i < fa.size; i++) {

os << fa.elements[i] << " ";

}

os << "]";

return os;

}

int main() {

// пункты 1, 2, 3, 4 задания

Fraction f;

cout << "Введите любую дробь: " << endl;

cin >> f;

cout << "Ваша дробь: " << f << endl << endl;

// пункт 5 задания

int temp;

cout << "Введите произвольное число, на которое умножим введённую дробь: ";

cin >> temp;

Fraction tempf = f\*temp;

cout << "Новая дробь: " << tempf << endl << endl;

// пункт 6 задания

FractionArray fa1, fa2, newfa;

int size1, size2;

cout << "Введите размер первого массива: ";

cin >> size1;

fa1 = FractionArray(size1);

cin >> fa1;

cout << "Первый массив: " << fa1 << endl << endl;

cout << "Введите размер второго массива: ";

cin >> size2;

fa2 = FractionArray(size2);

cin >> fa2;

cout << "Второй массив: " << fa2 << endl << endl;

// не работает

//operator\*=(fa1, fa2); // не работает

//cout << "Перемножение двух массивов: " << fa1 << endl;

return 0;

}



