

Задачи

Задача 1

Да се дефинира структура **Point**, описваща точка в евклидовата равнина.

Структурата **Point** трябва да съдържа следните **методи**:

- **read(std::istream& in)** – инициализира данните на структурата от потока за вход in
- **readBinary(std::istream& in)** – инициализира данните на структурата от потока за вход in, който ще представлява отворен двоичен файл
- **write(std::ostream& out)** – записва данните на структурата в потока за изход out
- **writeBinary(std::ofstream& out)** – записва данните на структурата в потока за изход out, който ще представлява отворен двоичен файл

Задача 2

Да се дефинира структура **Line**, описваща права в евклидовата равнина, зададена чрез две нейни точки.

Структурата **Line** трябва да съдържа следните **методи**:

- **bool isParallelWith (const Line& otherLine)** - проверява дали линията е успоредна с otherLine
- **bool contains (const Point& point)** - проверява дали дадена точка лежи на правата
- **Point intersection (const Line& otherLine)** - намира пресечната точка на две прави. Приемаме, че правите не са успоредни. Стойността на резултата може да е произволна в противен случай.
- **read(std::istream& in)** – инициализира данните на структурата от потока за вход in
- **readBinary(std::istream& in)** – инициализира данните на структурата от потока за вход in, който ще представлява отворен двоичен файл
- **write(std::ostream& out)** – записва данните на структурата в потока за изход out
- **writeBinary(std::ofstream& out)** – записва данните на структурата в потока за изход out, който ще представлява отворен двоичен файл

Задача 3

Да се дефинира структурата **Rational**, която реализира рационално число.

Структурата **Rational** да съдържа следните **методи**:

- **void makeRat(int, int)** - създава рационално число;
- **int numerator() const** - връща числителя на рационалното число;
- **int denominator() const** - връща знаменателя на рационалното число;
- **void printNumber() const** - извежда рационалното число;
- **read(std::istream& in)** – инициализира данните на структурата от потока за вход in;
- **readBinary(std::istream& in)** – инициализира данните на структурата от потока за вход in, който ще представлява отворен двоичен файл;

- **write(std::ostream& out)** – записва данните на структурата в потока за изход out;
- **writeBinary(std::ofstream& out)** – записва данните на структурата в потока за изход out, който ще представлява отворен двоичен файл.

Да се дефинират:

- булева функция **equal(const Rational& x, const Rational& y)**, която установява дали рационалните числа x и y са равни;
- булева функция **isGreaterThan(const Rational& x, const Rational& y)**, която установява дали рационалното число x е по-голямо от рац. число y;
- функция **maxRational(int n, Rational* numbers)**, която намира най-голямото от рационалните числа в редицата от рационални числа numbers.