

Санкт-Петербургский Государственный Электротехнический Университет
«ЛЭТИ» им. В. И. Ульянова (Ленина)

Кафедра информационных систем

Отчет

По практической работе №5

По дисциплине «Объектно-ориентированное программирование»

Студенты группы 2372 _____ Тубшинов В. Т., Алексеев Г.

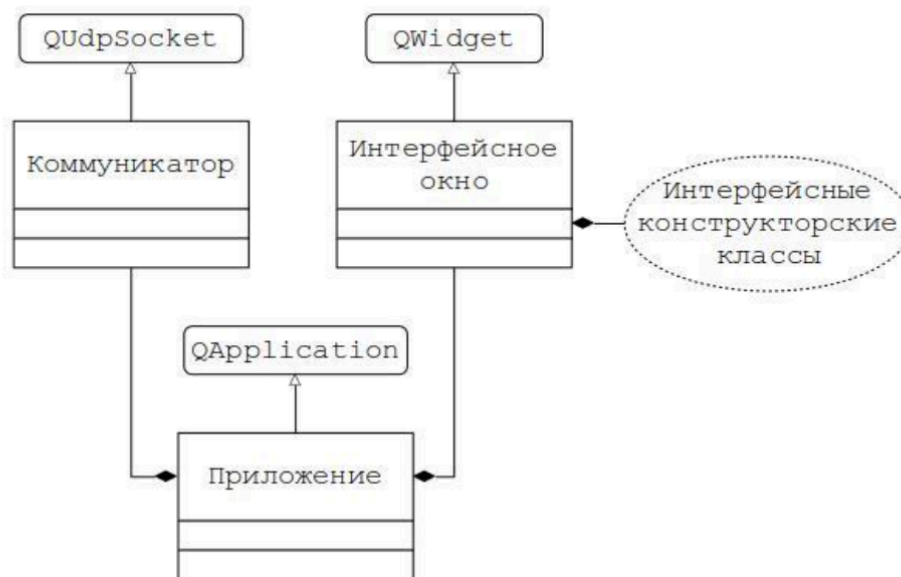
Преподаватель _____ Егоров С. С.

г. Санкт-Петербург

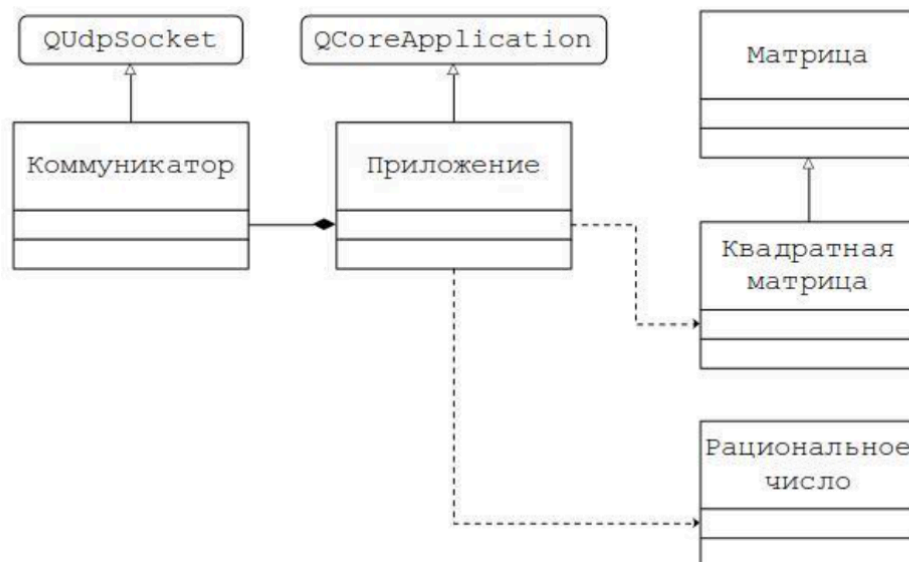
2023 г.

Задание: Создать распределенное приложение, включающее клиентскую и серверную части, взаимодействующие посредством сетевого обмена сообщениями.

Клиентская часть



Серверная часть



Клиентские и серверные части представляют собой приложения, реализованные в работе №4. Отличие заключается в том,

что класс «Квадратная матрица» делается наследников класса «Матрица» с произвольным числом строк и столбцов (отношение обобщения). Это влечет за собой перенос некоторых атрибутов и методов класса «квадратная матрица» в родительский класс «Матрица», что и предстоит сделать в этой работе.

Реализовать и отладить программу, удовлетворяющую сформулированным требованиям и заявленным целям. Разработать контрольные примеры и протестировать на них программу. Оформить отчет, сделать выводы по работе.

Спецификации классов:

Class Communicator:

Атрибуты:

- bool ready;
- CommParams params;

Методы:

- Communicator(CommParams&,QObject *parent = nullptr);
- bool isReady();
- void recieved(QByteArray);
- void send(QByteArray);
- void recieve();

Серверная часть:

Class Rational:

Атрибуты:

- int numerator, denominator - числитель и знаменатель соответственно;
- static QChar SEPARATOR;

Методы:

- Rational();
- Rational(int, int);

- Rational operator+ (Rational) - метод эквивалентный плюсу на поле рациональных чисел;
- Rational operator- (Rational) - метод эквивалентный минусу;
- Rational operator- () - метод эквивалентный унарному минусу;
- Rational operator* (Rational) - метод эквивалентный умножению;
- Rational operator/ (Rational) - метод эквивалентный делению;
- bool operator==(Rational) - метод эквивалентный равенству;
- int getNum() - получить числитель;
- int getDenom() - получить знаменатель;
- friend double Abs(Rational) - модуль числа;
- friend ostream& operator<<(ostream&, Rational) - вывод;
- friend istream& operator>>(istream&, Rational&) - ввод;
- friend QString& operator<<(QString&, Rational) - вывод
- static void setSeparator(QChar);
- friend QByteArray& operator>>(QByteArray&,Rational&);
- operator QString ();

Class Matrix:

Атрибуты:

number** Matrix;

Методы:

- matrix();
- matrix(unsigned short, unsigned short);
- ~matrix();
- void transposition(int);
- number MatrDeterminant(int);
- int rankOfMatrix3x3();
- int findMatrixRank(int);
- void show(QString&, int);
- void input(number**, int);

Class Application:

Атрибуты:

- Communicator *comm;

Методы:

- Application(int, char**);
- void recieve(QByteArray);

Class square_matrix

Методы:

square_matrix(unsigned short);
number MatrDeterminant(int);

Клиентская часть:

Class Application:

Communicator *comm;

Interface *interface, *window;

public:

Application(int, char**);

public slots:

void fromCommunicator(QByteArray);

void toCommunicator(QString);

class Interface:

Атрибуты:

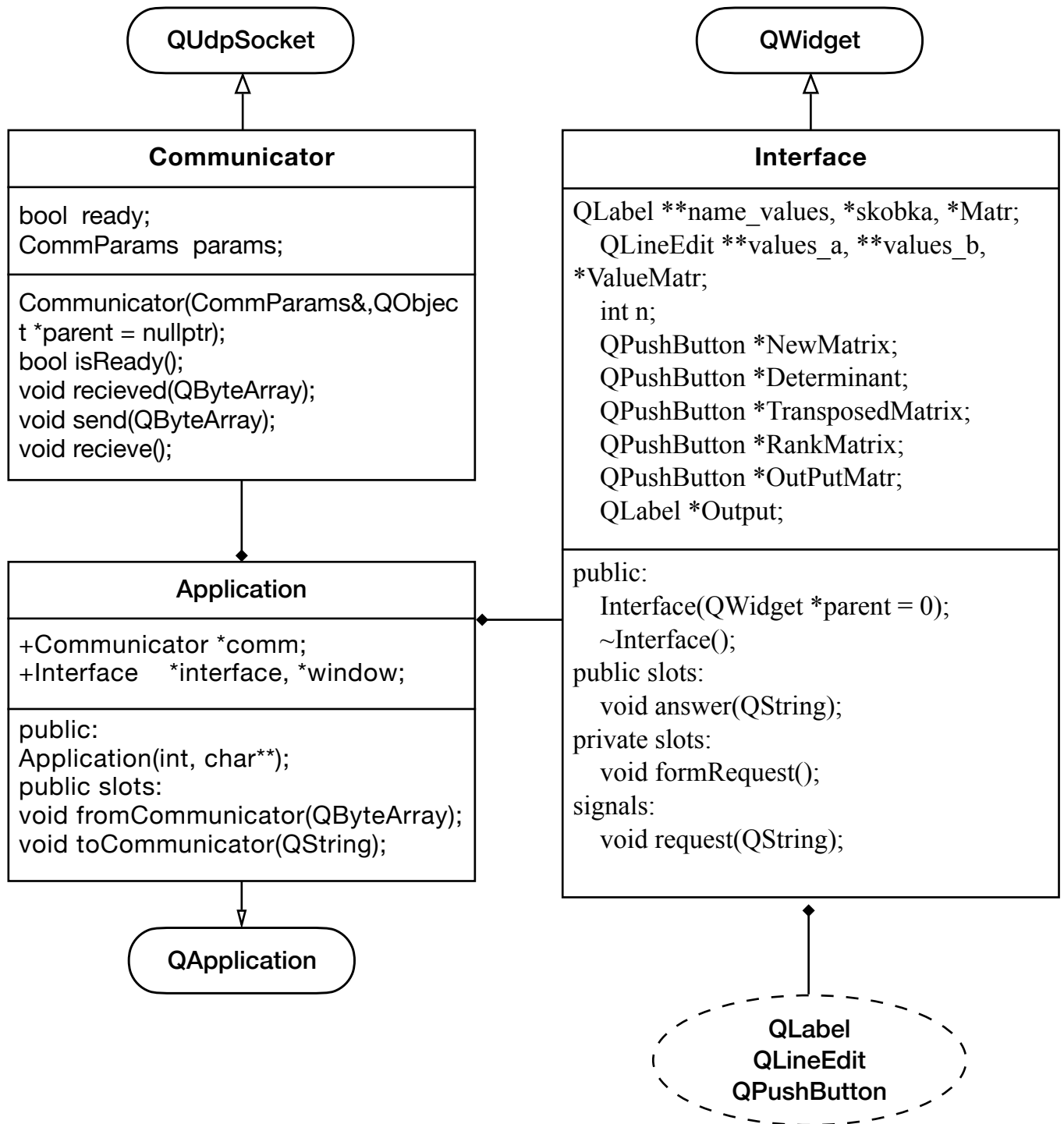
- QLabel **name_values, *skobka, *Matr;
- QLineEdit **values_a, **values_b, *ValueMatr;
- int n;
- QPushButton *NewMatrix;
- QPushButton *Determinant;
- QPushButton *TransposedMatrix;
- QPushButton *RankMatrix;

- QPushButton *OutPutMatr;
- QLabel *Output;

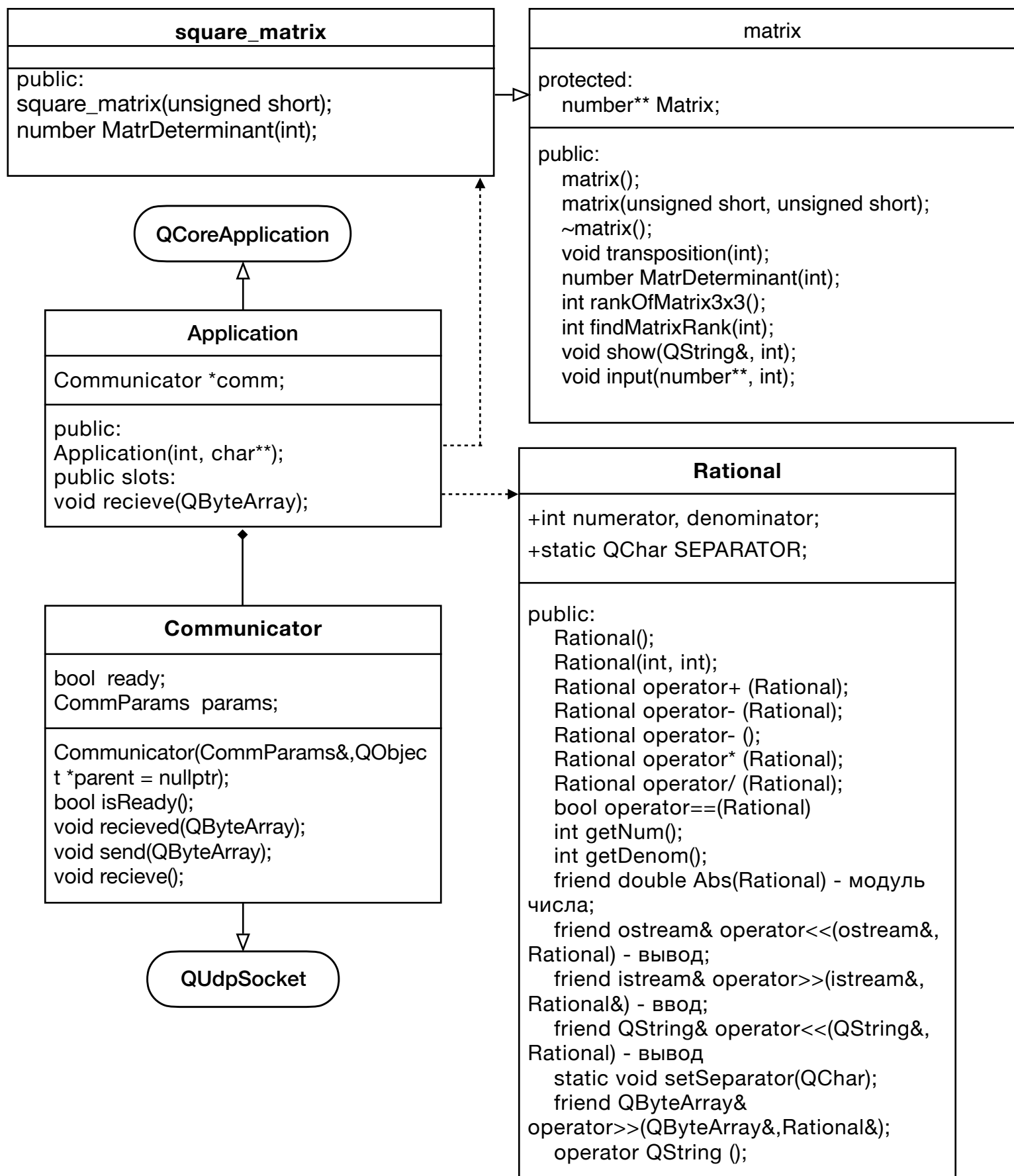
Методы:

- Interface(QWidget *parent = 0);
- ~Interface();
- void answer(QString);
- void formRequest();
- void request(QString);

Клиентская часть



Серверная часть



Контрольные примеры:

Матрица:

$$\mathbf{A} = \begin{pmatrix} 5 & 8 \\ -\frac{5}{6} & -\frac{8}{4} \\ 3 & 6 \\ -\frac{3}{7} & -\frac{6}{9} \end{pmatrix}$$

Определитель:

$$\det \mathbf{A} = \begin{vmatrix} 5 & 8 \\ -\frac{5}{6} & -\frac{8}{4} \\ 3 & 6 \\ -\frac{3}{7} & -\frac{6}{9} \end{vmatrix} = \left(-\frac{5}{6}\right) \cdot \left(-\frac{6}{9}\right) - \left(-\frac{3}{7}\right) \cdot \left(-\frac{8}{4}\right) = \frac{5}{9} - \frac{6}{7} = -\frac{19}{63}$$

Ранг:

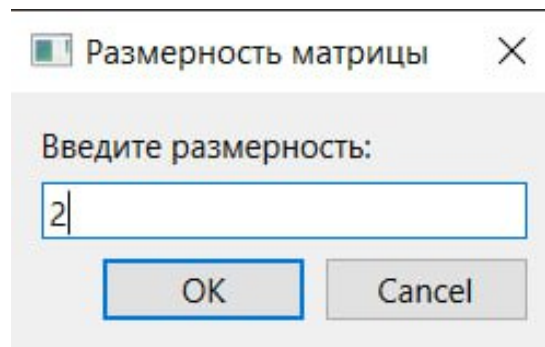
Так как ненулевых строк после использования метода Гаусса 2, то $\text{Rank}(\mathbf{A}) = 2$.

Транспозиция:

Чтобы найти транспонированную матрицу поменяем рядки и столбики матрицы \mathbf{A} местами:

$$\mathbf{A}^T = \begin{pmatrix} 5 & 3 \\ -\frac{5}{6} & -\frac{3}{7} \\ 8 & 6 \\ -\frac{8}{4} & -\frac{6}{9} \end{pmatrix}$$

Пример работы программы:

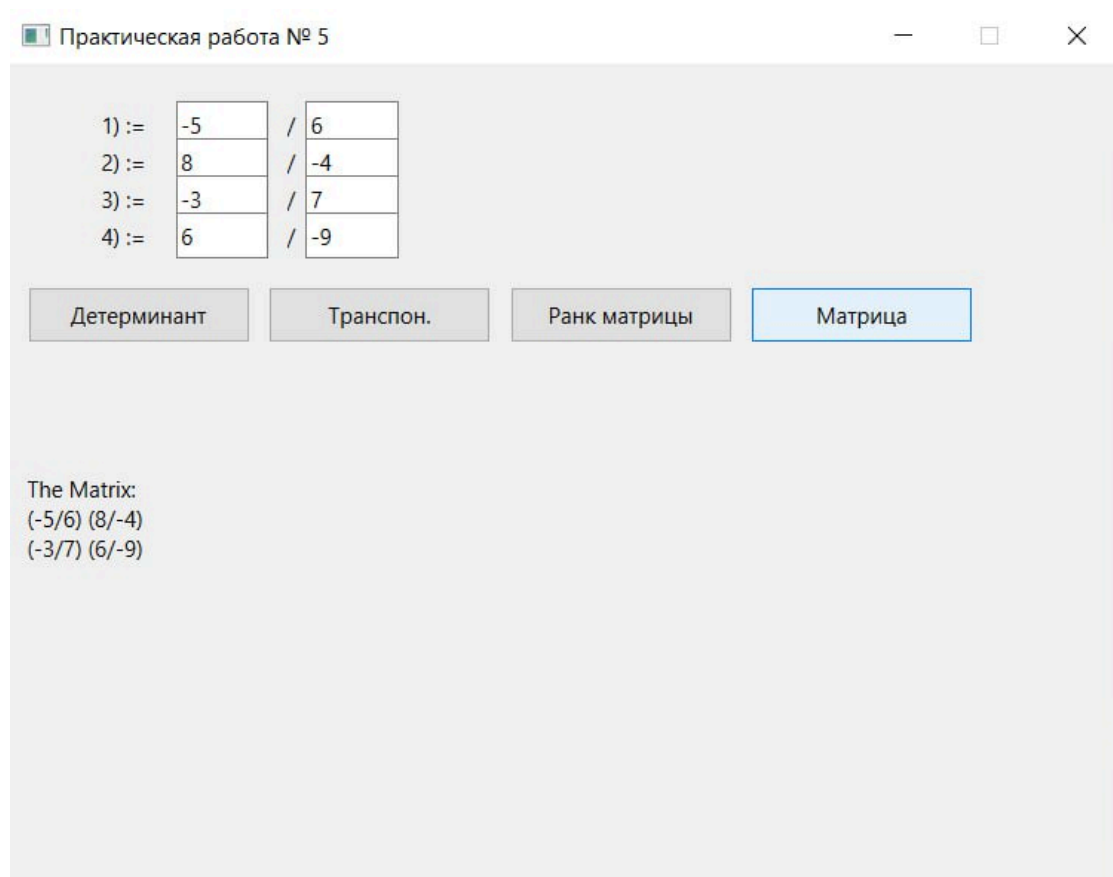


Размерность матрицы

Введите размерность:

2

OK Cancel



Практическая работа № 5

1) :=	-5	/	6
2) :=	8	/	-4
3) :=	-3	/	7
4) :=	6	/	-9

Детерминант Транспон. Ранг матрицы Матрица

The Matrix:
(-5/6) (8/-4)
(-3/7) (6/-9)

1) :=	-5	/	6
2) :=	8	/	-4
3) :=	-3	/	7
4) :=	6	/	-9

Детерминант

Транспон.

Ранг матрицы

Матрица

The rank = 2

1) :=	-5	/	6
2) :=	8	/	-4
3) :=	-3	/	7
4) :=	6	/	-9

Детерминант

Транспон.

Ранг матрицы

Матрица

The Transposed Matrix:

(-5/6) (-3/7)

(8/-4) (6/-9)

Практическая работа № 5

1) :=	-5	/	6
2) :=	8	/	-4
3) :=	-3	/	7
4) :=	6	/	-9

The Determinant = (-19/63)

Вывод:

В ходе пятой практической работы мы научились создавать распределенное приложение, включающее клиентскую и серверную части, взаимодействующие посредством сетевого обмена сообщениями. Так как мы научились использовать наследование на практике.