

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Саратовский государственный технический университет имени Гагарина Ю.А.»

Кафедра « Системотехника и управление в технических системах »

Click to edit the notes format

## **КУРСОВАЯ РАБОТА НА ТЕМУ:**

**«Создание распределенной системы решения задачи обработки информации»**

по дисциплине

**«Системное программное обеспечение»**

Вариант 2.8.6.2.3.4

Выполнил: студент группы БУПТС-21

факультет ИнЭТиП

форма обучения очная

ФИО:

Новиков Алексей Романович

Проверил: профессор кафедры СТУ

Степанов М.Ф.

# Введение

Click to edit the notes format

Click to edit the notes format

Click to edit the notes format

# Результаты работы программы

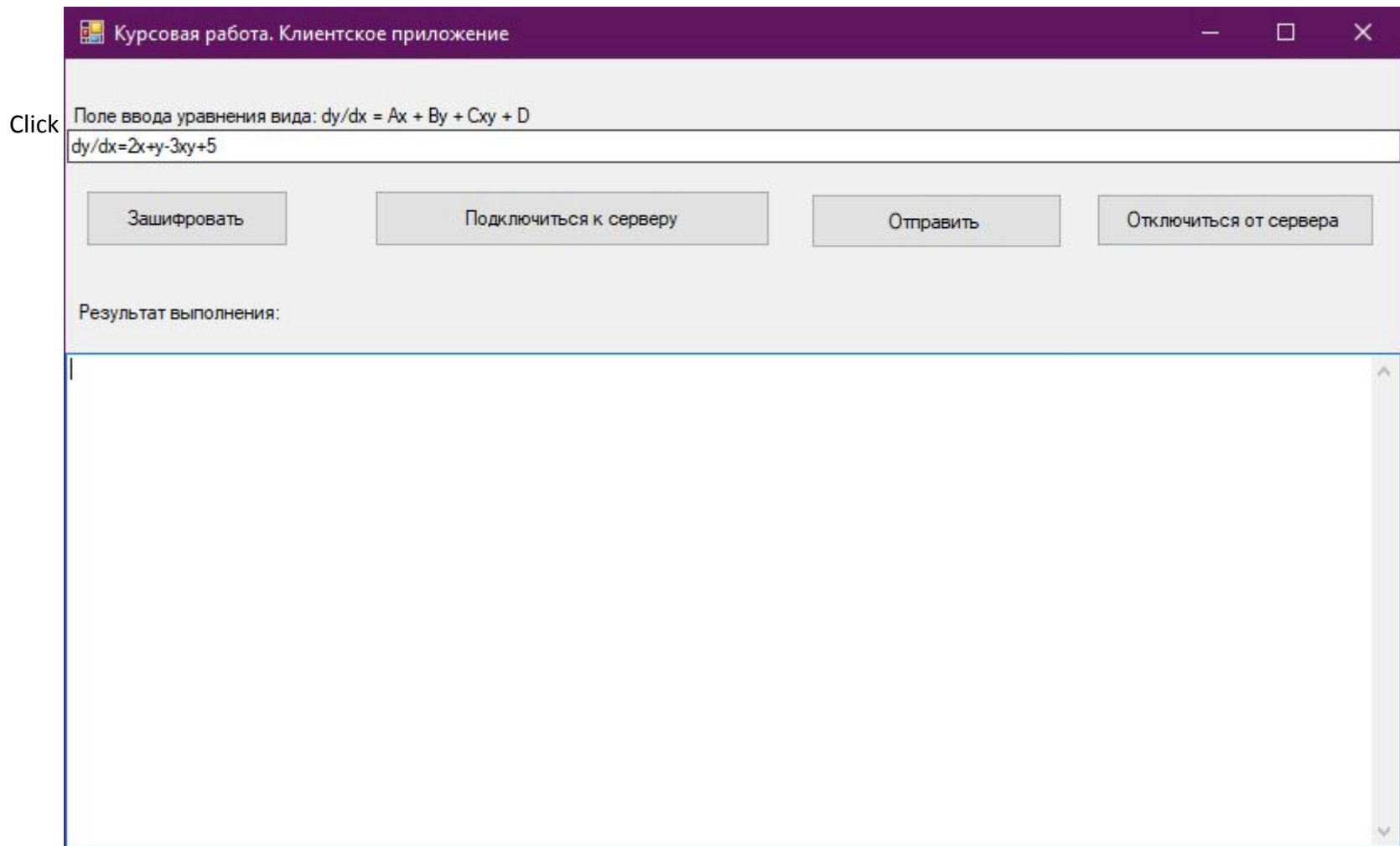


Рис.1. Ввод данных

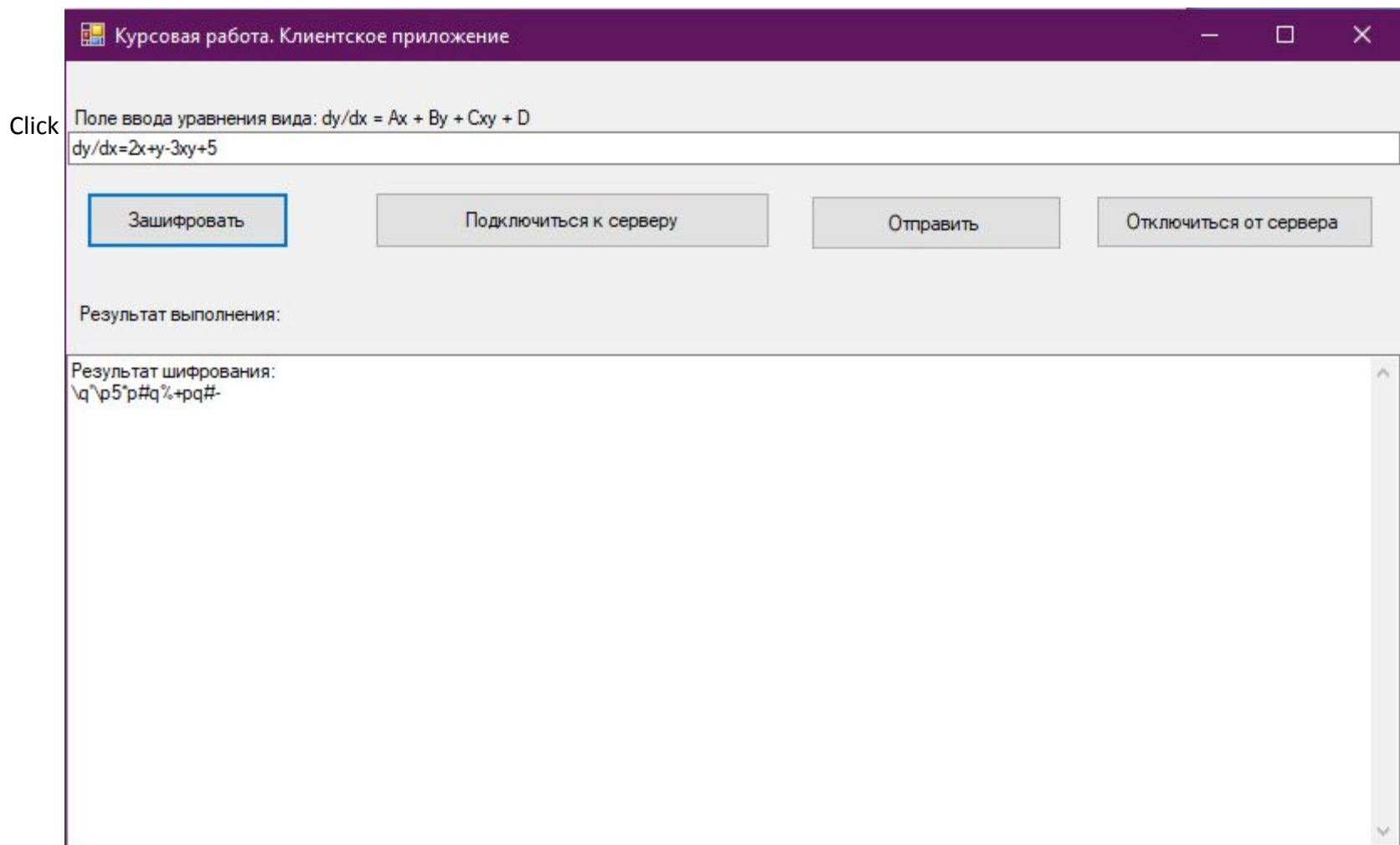


Рис.2. Шифрование данных методом Цезаря

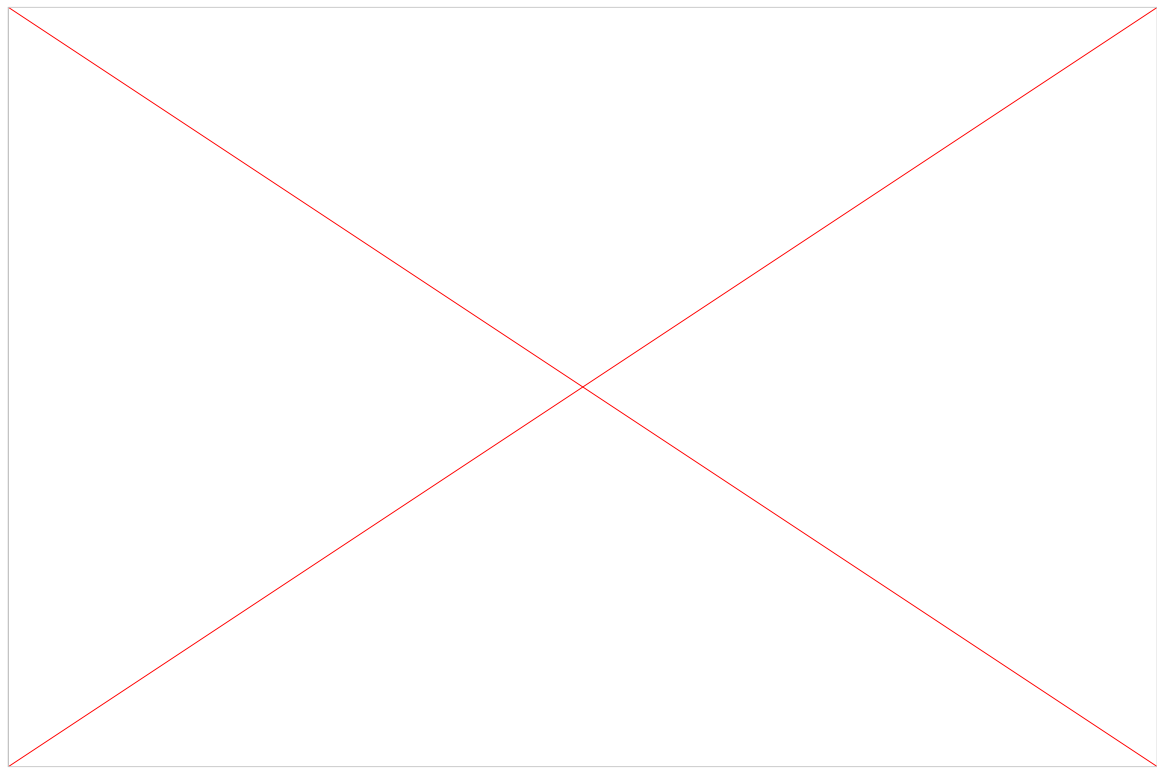


Рис.3. Подключение к серверу и отправка на него данных через сокет

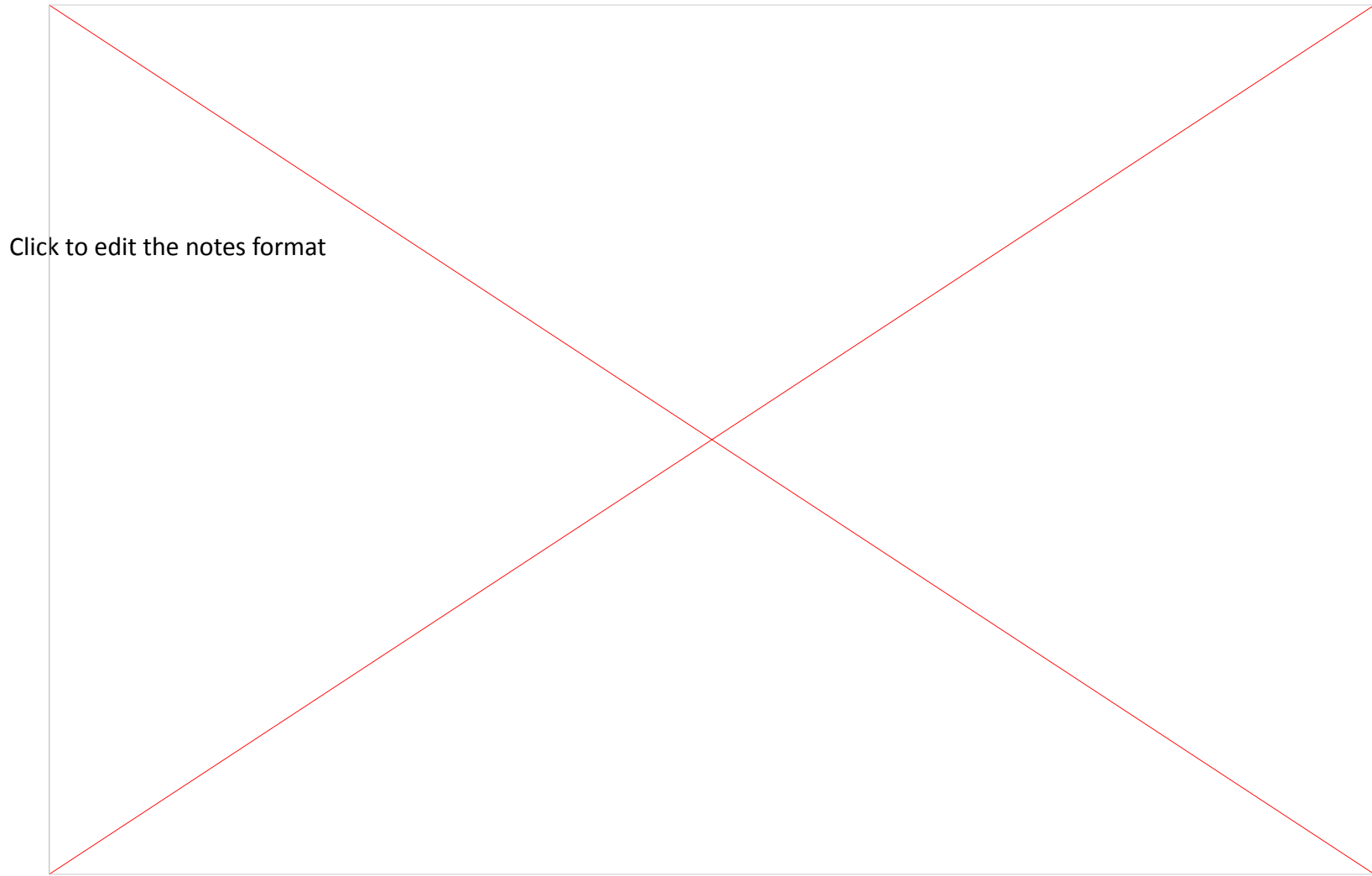


Рис.4. Отключение от сервера



# Вывод

В данной курсовой работе была проведена работа с распределенной системой решения задач, осуществляющую решение заданной задачи обработки информации: 1) с видом криптографической защиты, 2) с видом входного языка, 3) со способом организации таблиц идентификаторов, 4) с методом проведения синтаксического анализа, 5) с видом внутреннего представления программы, 6) с видом функциональной задачи. Программа была реализована на языке C++.

Так же в разработке подлежали:

1. Структуры данных (класс языка C++) для представления исходных данных и результатов решения функциональной задачи.
2. Функциональный программный модуль – функция языка C++, осуществляющая вычислительные действия по решению заданной функциональной задачи.
3. Библиотека динамической загрузки, содержащая разработанные классы языка C++.
4. Библиотека динамической загрузки, содержащая функциональный программный модуль.
5. Автономное тестовое приложение, служащее для проверки работоспособности функционального программного модуля, вызываемого из библиотеки динамической загрузки, подключаемой динамически в специально созданном потоке команд (нити) для разгрузки основного потока команд (нити), обеспечивающего интерфейс с пользователем.
6. Средства криптографической защиты информации.
7. Средства идентификации и аутентификации пользователя приложения.
8. Проблемно-ориентированный язык (заданной структуры) для представления внешней (пользовательской) формы постановки функциональной задачи.
9. Транслятор внешнего представления постановки функциональной задачи (на проблемно-ориентированном языке заданной структуры) во внутреннее представление (заданной структуры) в составе: лексический анализатор, синтаксический анализатор, генератор выходного представления.
10. Серверное приложение, осуществляющее с помощью построенного транслятора анализ текстового представления постановки задачи пользователя, полученного от клиентского приложения, вызов функционального программного модуля, передача результатов работы функционального программного модуля клиентскому приложению.
11. Клиентское приложение, обеспечивающее интерфейс с пользователем, формирование текстового представления постановки пользовательской задачи обработки информации, отправку запроса на обработку информации серверу, получение сообщения сервера с результатами обработки информации, визуализация полученных результатов для пользователя.