

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МОЭВМ**

**ОТЧЕТ**  
**по учебной практике**  
**Тема: Генетические алгоритмы**

Студент гр. 3343

Жучков О.Д.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2025

## **Цель работы**

Изучение принципов генетических алгоритмов и их применения для решения оптимизационных задач, программная реализация генетического алгоритма для решения задачи и пользовательского интерфейса для него.

## **Задание**

Вариант 12. Задача прямоугольного раскроя.

Дана полубесконечная полоса шириной  $N$ , необходимо разместить  $M$  прямоугольников, размеры  $i$ -го прямоугольника задаются пользователем как ширина  $w_i$  и длина  $l_i$ . Разместить прямоугольники необходимо так, чтобы потребовался участок полосы минимальной длины..

Требования к решению:

Программа должна иметь GUI

Должна быть возможность задать данные из разных источников по выбору пользователя: из файла, ввод через GUI, случайная генерация

При реализации алгоритмов нельзя использовать библиотеки решающие задачу напрямую. Генетический алгоритм должен быть реализован вручную. Использовать библиотеки для графического интерфейса/загрузки данных – можно.

Пользователь должен иметь возможность задавать параметры генетического алгоритма через GUI.

В приложении должна быть реализована пошаговая демонстрация работы генетического алгоритма. Т.е. должна быть кнопка «следующий шаг» и кнопка «выполнить до конца» (чтобы перейти к конечным результатам). На каждом шаге алгоритма должна быть показано графическое представление наилучшего решения, стоимость этого решения, средняя стоимость поколения.

В GUI должно быть поле с графиком изменения приспособленности лучший и средней в зависимости от поколения. График должен строиться по ходу выполнения генетического алгоритма.

После завершения выполнения генетического алгоритма, программа не должна закрываться, а должна быть возможность выполнить его на этих же данных, но с другими параметрами алгоритма, либо загрузить новые данные.

## **Выполнение работы**

Программа для реализации алгоритма написана на языке Python. Для графического пользовательского интерфейса и вывода графиков используются библиотеки `tkinter` и `matplotlib`.

В файле `interface.py` описан класс, создающий окно `tkinter` и обеспечивающий пользовательское взаимодействие с программой. Пользователь может: ввести данные для решаемой задачи, поменять параметры генетического алгоритма, произвести пошаговое выполнение алгоритма и проследить за промежуточными результатами.

Основной файл `main.py` отвечает за создание объектов пользовательского интерфейса и исполнителя генетического алгоритма и осуществляет взаимодействие между ними.

В файле `model.py` описан класс, производящий решение задачи генетическим алгоритмом. Генетическое представление решения – перестановки чисел от 0 до  $N$ , которые сопоставляются с набором  $N$  прямоугольников, которые необходимо разместить на полосе. Для декодирования данного представления написан алгоритм, работающий следующим образом: берется  $i$ -ый прямоугольник из перестановки и размещается в ближайшее к началу доступное место (не занятое предыдущими  $i-1$  прямоугольниками) на полосе (позиции рассматриваются слева направо, снизу вверх). Таким образом высчитываются два значения: длина участка, требуемого для размещения, и число приспособленности, обратное длине (чем меньше длина, тем оптимальнее решение и выше приспособленность). Данный подход к кодированию решения исключает возможность некорректного решения или заведомо неоптимального.

К создаваемым случайным перестановкам в процессе выполнения генетического алгоритма применяются методы скрещивания (упорядоченное скрещивание) и мутации (мутация случайным обменом двух элементов в перестановке).

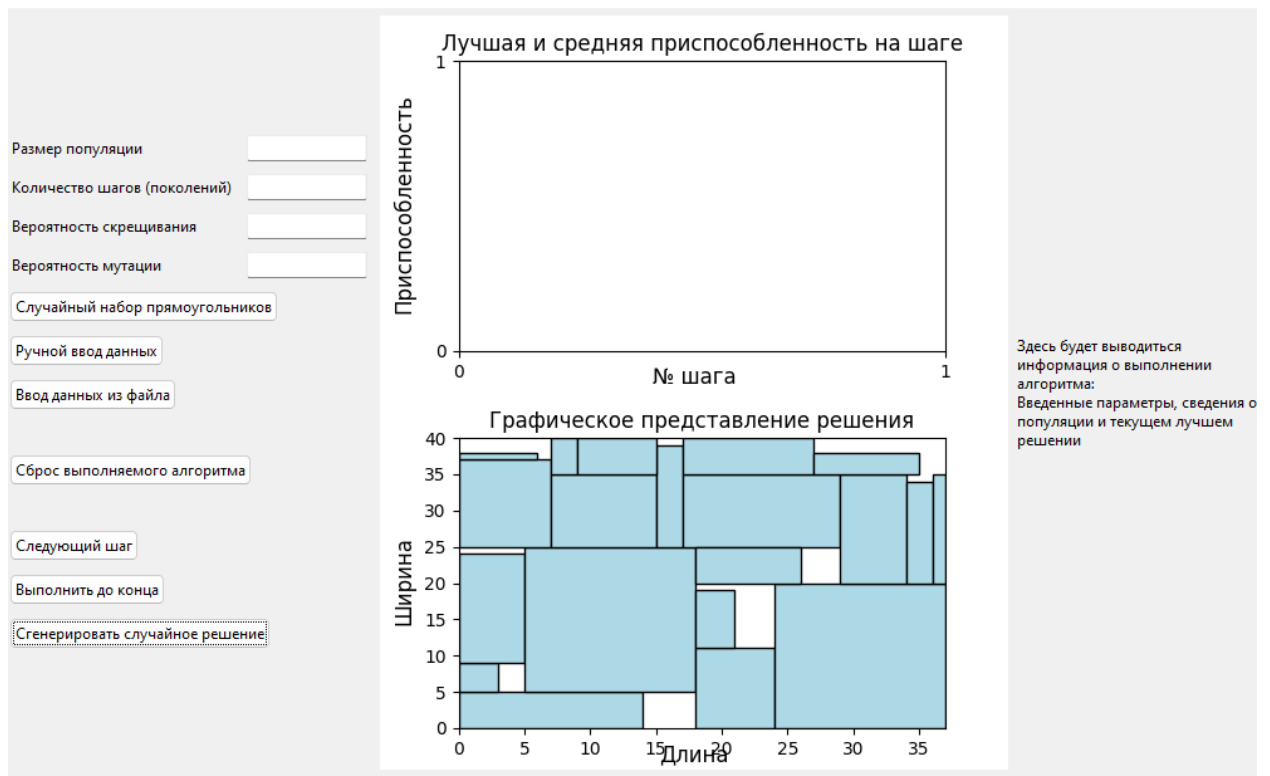


Рисунок 1 – Первый прототип GUI