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

ОТЧЕТ

по учебной практике

Тема: Генетические алгоритмы

Студентка гр. 1304 _____ Виноградова М.О.
Студент гр. 1304 _____ Стародубов М.В.
Студент гр. 1304 _____ Макки К.Ю.
Руководитель _____ Жангиров Т.Р.

Санкт-Петербург
2023

ЗАДАНИЕ

НА УЧЕБНУЮ ПРАКТИКУ

Студентка Виноградова М.О. группы 1304

Студент Стародубов М.В. группы 1304

Студент Макки К.Ю. группы 1304

Тема практики: Генетические алгоритмы

Задание на практику:

Решение задачи коммивояжера на C++ с графическим интерфейсом.

Алгоритм: коммивояжера.

Сроки прохождения практики: 30.06.2023 – 13.07.2023

Дата сдачи отчета: 13.07.2020

Дата защиты отчета: 13.07.2020

Студентка	_____	Виноградова М.О.
Студент	_____	Стародубов М.В.
Студент	_____	Макки К.Ю.
Руководитель	_____	Жангиров Т.Р.

АННОТАЦИЯ

С. 19. Ил. 3. Литература 3. Прил. 1

Объектом исследования данной практической работы является алгоритм коммивояжера. Этот алгоритм предназначен для нахождения наиболее оптимального маршрута, проходящего через указанные города хотя бы один раз и возвращающегося в исходный город.

Целью данного проекта является разработка программы с графическим интерфейсом на основе генетических алгоритмов. Этот интерфейс позволит пользователям взаимодействовать с программой и наблюдать пошаговую визуализацию процесса поиска оптимального решения.

В процессе подготовки к практической работе были использованы следующие ресурсы:

Панченко, Т. В. Генетические алгоритмы [Текст] : учебно-методическое пособие / под ред. Ю. Ю. Тарасевича. — Астрахань : Издательский дом «Астраханский университет», 2007. — 87 [3] с.

В работе был представлен процесс решения задачи, документация по использованию программы, а также приведены примеры кода на языке C++ и результаты тестирования.

SUMMARY

The object of research of this practical work is the traveling salesman algorithm(TSP). This algorithm is designed to find the most optimal route passing through the specified cities at least once and returning to the original city.

The purpose of this project is to develop a program with a graphical interface based on genetic algorithms. This interface will allow users to interact with the program and observe a step-by-step visualization of the process of finding the optimal solution.

In the process of preparing for the practical work, the following resources were used:

Панченко, Т. В. Генетические алгоритмы [Текст] : учебно-методическое пособие / под ред. Ю. Ю. Тарасевича. — Астрахань : Издательский дом «Астраханский университет», 2007. — 87 [3] с.

The paper presented the process of solving the problem, documentation on the use of the program, as well as C++ code examples and test results.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Цель практической работы заключается в разработке программы с графическим интерфейсом, основанной на генетических алгоритмах, для решения задачи коммивояжера.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ

1.1. Исходные Требования к программе

- Программа должна иметь GUI
- Должна быть возможность задать данные через GUI (CLI), чтение из файла, случайную генерацию.
- Алгоритмы реализуются самостоятельно.
- Настройкой параметров алгоритмов должна производиться пользователем.
- Пошаговая визуализация поиска решения (как меняется аппроксимирующая функция, текущий экстремум, текущее решение оптимизационной задачи). Также должны отображаться 2 следующих лучших решения.
- Должна быть возможность перейти к конечному решению пропустив все шаги.
- Должен присутствовать график изменения функции качества решения с каждым шагом, дополняющийся с каждым шагом.

Доп. баллы даются за:

- Реализацию возможности вернуться на несколько шагов назад.
- Реализацию и возможность выбора модификаций алгоритмов для поиска решения.

1.2. Уточнение требований после...

2. РАСПРЕДЕЛЕНИЕ РОЛЕЙ В БРИГАДЕ

Разработка алгоритма - Виноградова, Стародубов, Макки.

Реализация графического интерфейса, реализация алгоритма отбора в новую популяцию – Виноградова.

Реализация основных структур данных, реализация алгоритма отбора родителей – Стародубов.

Реализация алгоритма рекомбинации и мутации – Макки.

Распределение ролей представлено на сх. 1.

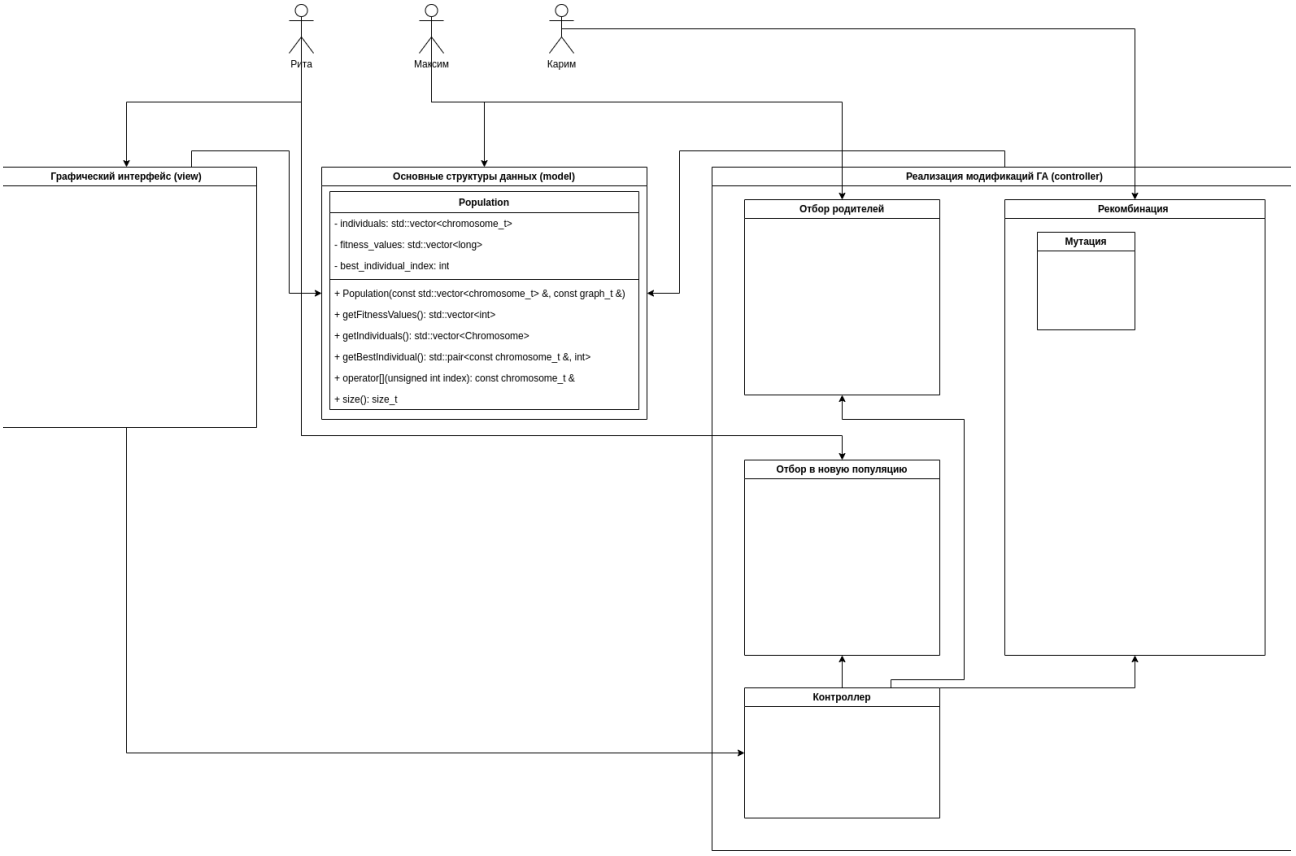

Схема 1 - Распределение ролей

3. ИТЕРАЦИЯ №1

3.1. План разработки

3.1.1 GUI

Для удобства взаимодействия пользователя с программой был разработан следующий графический интерфейс (рис.1-3)

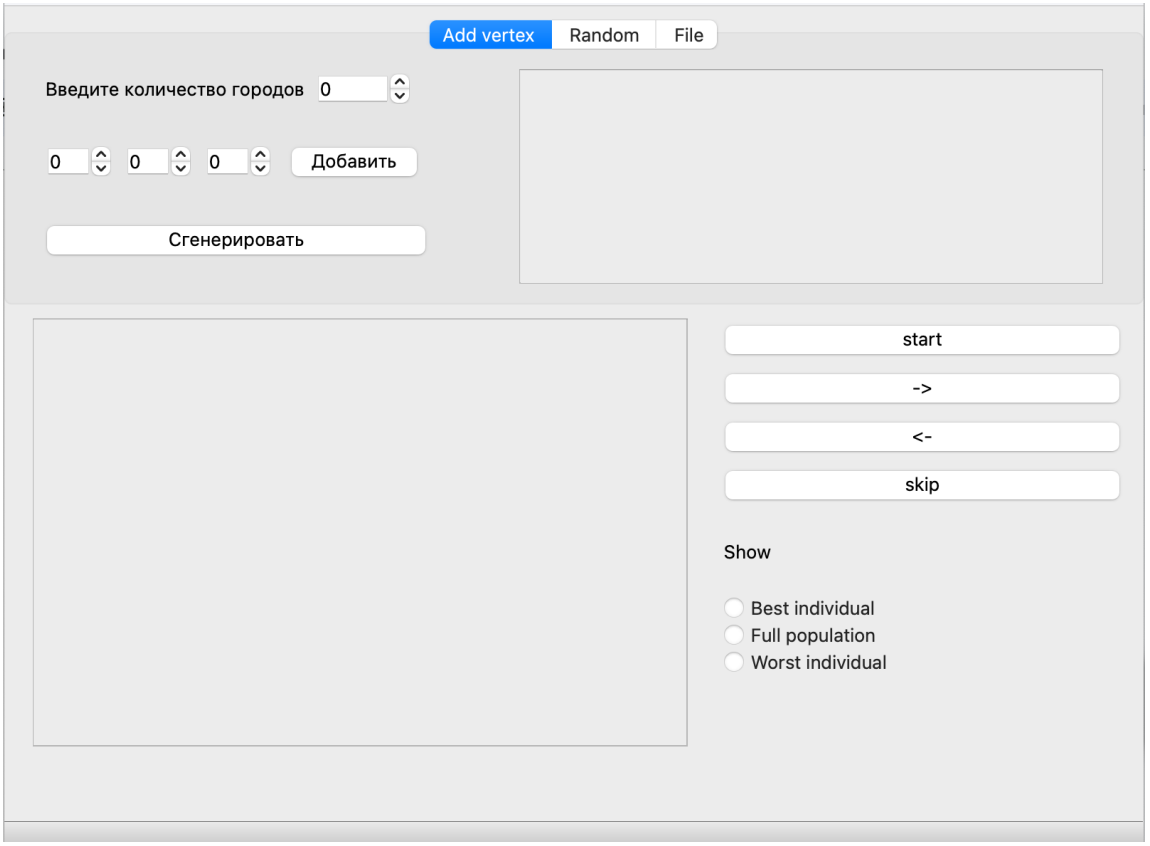

Рисунок 1 - Прототип GUI (вкладка Add vertex)

На вкладке “Add vertex” пользователь может ввести количество городов, а также может указать какие дороги проходят между городами, все добавленные связи будут отображаться в соседней области. После нажатия кнопки “Сгенерировать”, будет создан граф.

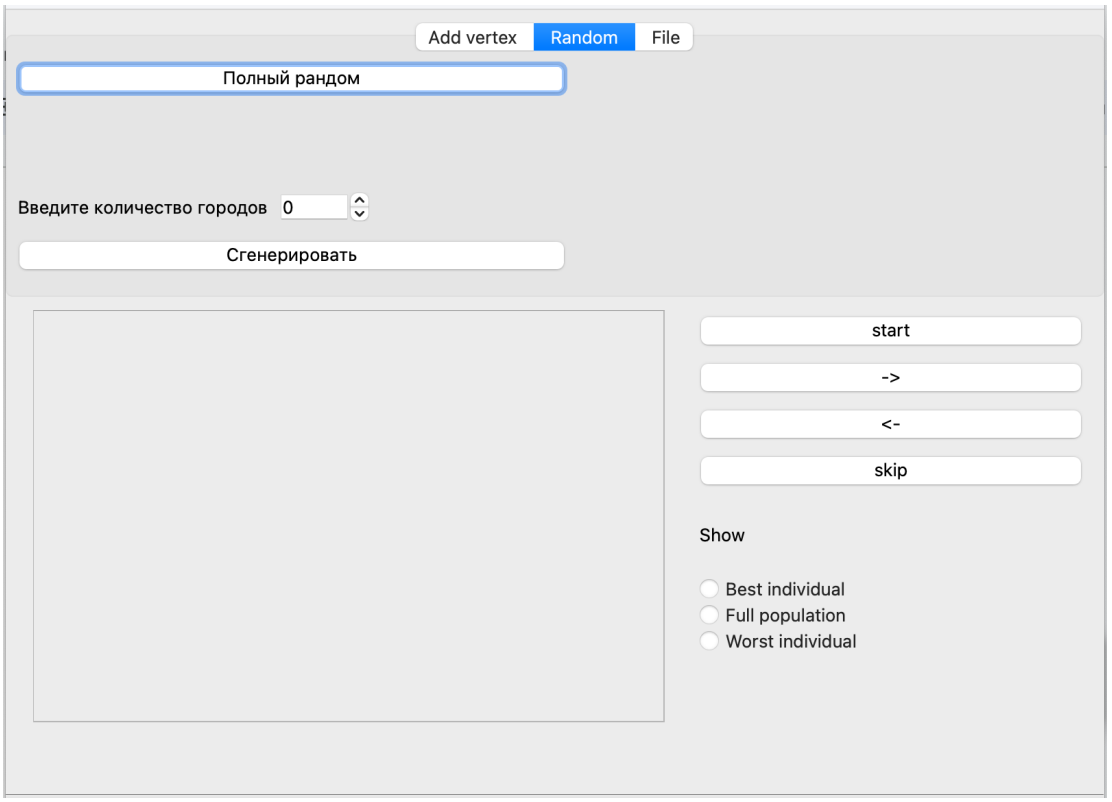

Рисунок 2 - Прототип GUI (вкладка Random)

На вкладке “Random” пользователь может ввести количество городов. После нажатия кнопки “Сгенерировать” будут автоматически сгенерированы “дороги” между городами.

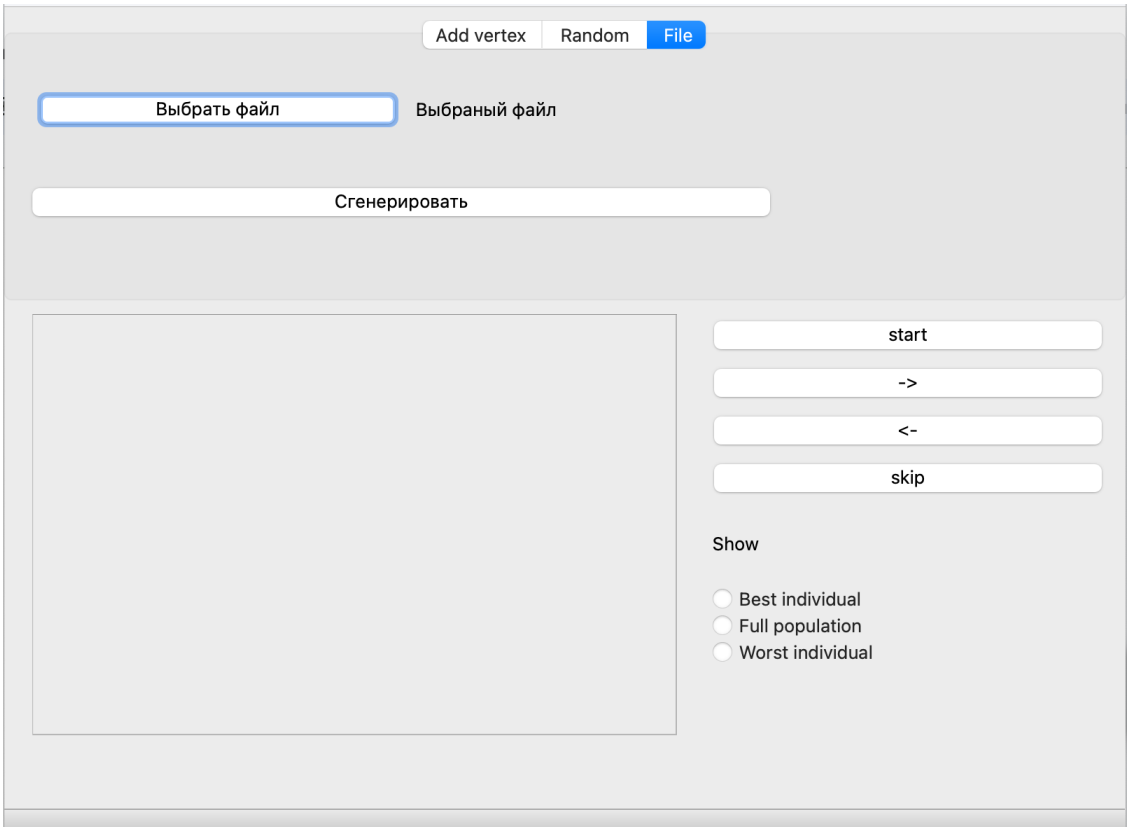

Рисунок 3 - Прототип GUI (вкладка File)

На данной вкладке пользователь может выбрать файл, по содержимому которого будет построен граф. (Требования к файлу: n строк, n столбцов, разделитель пробел).

Также есть область, где будет отображаться, граф/популяция. Справа находятся кнопки управления: start – по входным данными определяет начальную популяцию, -> - шаг вперед, <- – шаг назад, skip – финальное решение. Также можно выбрать данные для отображения: лучшая особь популяции, худшая особь популяции, все особи популяции.

3.1.2 Модификации генетического алгоритма, метрики и настраиваемые параметры

Следующие модификации были выбраны при реализации алгоритма:

1. Инициализация начальной популяции:

Для начала решения задачи коммивояжера случайно выбираются $m \cdot n$ особей, являющихся гамильтоновыми циклами исходного графа, где m – некоторая константа, n – число вершин графа. Каждый цикл представляет собой случайную перестановку вершин графа.

2. Выбор родителей может осуществляться несколькими методами:

Один из которых *панмиксия* – простой оператор отбора, при котором каждой особи в популяции присваивается случайное целое число от 1 до n. Он универсален для различных задач, но менее эффективен с увеличением численности популяции. Вторым методом является *инбридинг*, при котором первый родитель выбирается случайно, а вторым становится член популяции, наиболее близкий к первому. Близость может определяться расстоянием Хемминга (для бинарных строк) или евклидовым расстоянием (для вещественных векторов). *Аутбридинг* же формирует брачные пары из максимально далеких особей.

Селекция в генетическом алгоритме выбирает родителей на основе их приспособленности. Это обеспечивает быструю сходимость, но может привести к преждевременной сходимости к одному решению. Используются разные механизмы отбора, такие как турнирный отбор и метод рулетки. Пороговая величина для селекции может быть вычислена разными способами.

Турнирный отбор выбирает t случайных особей из популяции размером N. Лучшая из них записывается в промежуточный массив, повторяя эту операцию N раз. Особи из промежуточного массива используются для скрещивания. Преимущество турнирного отбора - отсутствие дополнительных вычислений.

В методе рулетки особи отбираются с помощью N «запусков» рулетки, где N – размер популяции. Колесо рулетки содержит по одному сектору для

каждого члена популяции. Размер i-го сектора пропорционален вероятности попадания в новую популяцию $P(i)$, вычисляемой по формуле:

$$P(i) = \frac{f(i)}{\sum_{i=1}^{N} f(i)}$$

В результате анализа данных модификаций мы сделали выбор в пользу метода рулетки, так как в случае панмиксии мы не учитываем наше лучшее решение, а использование инбридинга в нашем случае будет малоэффективным, так как повторяющиеся циклы породят еще один похожий цикл. Реализация аутбридинга требует больших вычислительных затрат.

3. Выбор Рекомбинации:

Из двух методов рекомбинации был выбран метод Кроссинговер, при котором два родителя выбираются для создания потомства. Применяется кроссинговер по одной точке, где случайным образом выбирается точка разреза. Генетический материал от родителей объединяется, и получается потомок. Всего создается $m \cdot n$ потомков.

4. Мутация:

Так как в результате применения кроссинговера полученный цикл может иметь повторяющиеся вершины, необходимо провести корректировку, для этого повторяющиеся вершины в цикле будут заменяться на вершины, отсутствующие в нем. После этого происходит случайная перестановка некоторого количества вершин.

5. Выбор новой популяции:

Новая популяция формируется на основе элитарного признака. Лучшие $m \cdot n$ особей, включая родителей и потомков, добавляются в новую популяцию.

3.1.3 Сторонние библиотеки

Для создания графического интерфейса использовался кроссплатформенный фреймворк Qt. Он предоставляет набор инструментов и библиотек для разработки приложений с красивыми и функциональными интерфейсами, которые могут работать на разных операционных системах. Qt имеет широкий выбор виджетов и элементов управления, а также мощные возможности для обработки событий и работы с сетью и базами данных. Это делает его популярным выбором для разработчиков приложений с графическим интерфейсом.

3.1.4 Структуры данных

Для хранения графа используется матрица смежности, данный способ хранения графа позволяет получить доступ к весу ребра ориентированного графа за константное время.

Для решения задачи с помощью генетического алгоритма необходимо определить основные структуры данных, определяющие особь и популяцию.

Принято решение хранить особь как одну хромосому, представляющую из себя упорядоченный массив целых чисел, каждое число в хромосоме определяется как вершина графа, упорядоченное множество вершин образует цикл.

Популяция определяется как группа особей, для удобства дальнейшей работы с популяцией, в класс добавлены методы, позволяющие узнать значения функции приспособленности для каждой особи, получить информацию об особи с наилучшим значением функции приспособленности.

Для удобного хранения различных поколений популяций определена структура списка популяций, она будет использоваться для отображения предыдущих популяций в графическом интерфейсе.

3.2. Результаты по итерации № 1

В результате первой итерации было произведено распределение участников бригады по разным ролям, в соответствие с которыми будет реализовываться дальнейшая программа. Был разработан прототип графического интерфейса в среде Qt Designer. Также были описаны модификации и метрики генетического алгоритма для поставленной задачи (коммивояжера), структуры данных для хранения. Представлен предполагаемых стек технологий.

4. ТЕСТИРОВАНИЕ

4.1. Первый подраздел третьего раздела

4.2. Второй подраздел третьего раздела

ЗАКЛЮЧЕНИЕ

Кратко подвести итоги, проанализировать соответствие поставленной цели и полученного результата.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Панченко, Т. В. Генетические алгоритмы [Текст] : учебно-методическое пособие / под ред. Ю. Ю. Тарасевича. — Астрахань : Издательский дом «Астраханский университет», 2007. — 87 [3] с.
2. https://www.geeksforgeeks.org/traveling-salesman-problem-using-genetic-algorithm/
3. https://jaketae.github.io/study/genetic-algorithm/
4. https://doc.qt.io/

ПРИЛОЖЕНИЕ А

НАЗВАНИЕ ПРИЛОЖЕНИЯ

полный код программы должен быть в приложении, печатать его не надо