МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Систем обработки информации и управления»

ОТЧЕТ

**Лабораторная работа №\_\_4\_\_**

по дисциплине«Проектирование интеллектуальных систем»

Тема: «Решение оптимизационных задач с помощью генетических алгоритмов»

ИСПОЛНИТЕЛЬ: Белоусов Е.А.

ФИО

группа ИУ5-71 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"\_\_"\_\_\_\_\_\_\_\_\_2020 г.

ПРЕПОДАВАТЕЛЬ: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ФИО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись

"\_\_"\_\_\_\_\_\_\_\_\_2020 г.

Москва - 2020

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Цель работы**
   1. Целью лабораторной работы является углубление и закрепление теоретических знаний, полученных на лекциях, приобретение практических навыков самостоятельной работы при решении оптимизационных задач больших размерностей с помощью генетических алгоритмов.
   2. В процессе выполнения лабораторной работы по теме «Решение оптимизационных задач с помощью генетических алгоритмов» на примере задачи поиска кратчайшего пути для информационного пакета (сообщения) в компьютерной сети студенты решают следующие задачи (задания):
      * описывают предметную область;
      * определяют исходные данные задачи;
      * формулируют задачу и исходные данные в терминах генетических алгоритмов;
      * определяют последовательность работы генетического алгоритма;
      * разрабатывают компьютерную программу;
      * исследуют работу генетического алгоритма и полученное решение.
2. **Задание**
   1. Сформулировать задачу и описать исходные данные в терминах генетических алгоритмов.
   2. Разработать программу, которая осуществляет поиск кратчайшего пути для информационного пакета (сообщения) в компьютерной сети с помощью генетического алгоритма.
   3. При проведения серии экспериментов (не меньше 10) по исследованию работы генетического алгоритма программа должна позволять пользователю задавать топологию сети (пропускные способности каналов связи), содержащей не менее 10 компьютеров (серверов), а также указывать компьютер-отправитель и компьютер-получатель. Должны отображаться все решения (хромосомы) одного поколения до и после применения каждого оператора (скрещивания, селекции, редукции и мутации). Переход к следующему поколению должен осуществляться: в автоматическом режиме в соответствии с заданным критерием; в ручном режиме.
3. **Описание предметной области и выбранной задачи.**

Предметной областью работы являются компьютерные сети. Между некоторыми компьютерами в сети существуют соединения, которые, в зависимости от их вида, способны передавать информацию за различное время. Алгоритм программы ищет путь пакета от компьютера отправителя до компьютера приемника, который занимает меньше всего времени.

1. **Формулировка задачи, описание исходных данных в терминах генетических алгоритмов и блок-схема генетического алгоритма.**

В качестве хромосомы будем использовать список номеров промежуточных компьютеров пути пакета. Компьютер отправитель и компьютер приемник в хромосому не входят.

Длина между компьютерами сети может быть задана вручную или сгенерирована случайным образом.

Задача алгоритма – минимизировать длину пути пакета в сети.

Скрещивание происходит между двумя родителями, выбираемыми случайным образом. При скрещивании потомок получает ген одного из родителей с равной вероятностью

После операции скрещивания происходит мутация – случайное изменение гена.

После чего из родителей и потомков отбирается некоторое количество лучших хромосом и переводится в следующую популяцию.





1. **Описание программы, ее ключевые особенности и новшества.**

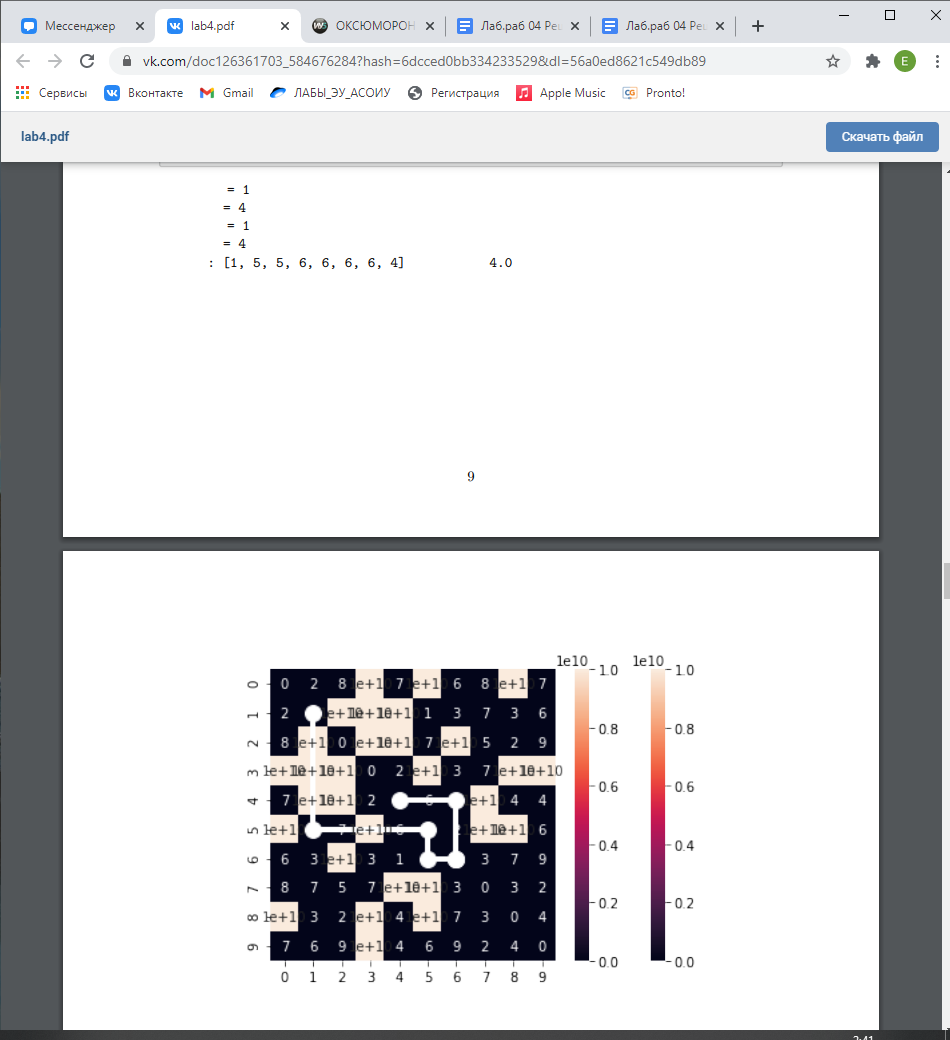
В программе представлена возможность остановить выполнение алгоритма и получить лучший результат на данный момент, это оказывается полезным при работе с большим количеством данных.

В целом, авторы работы, пытались придать алгоритму свойства генетического алгоритма селекционера.

1. **Протоколы проведенных экспериментов.**

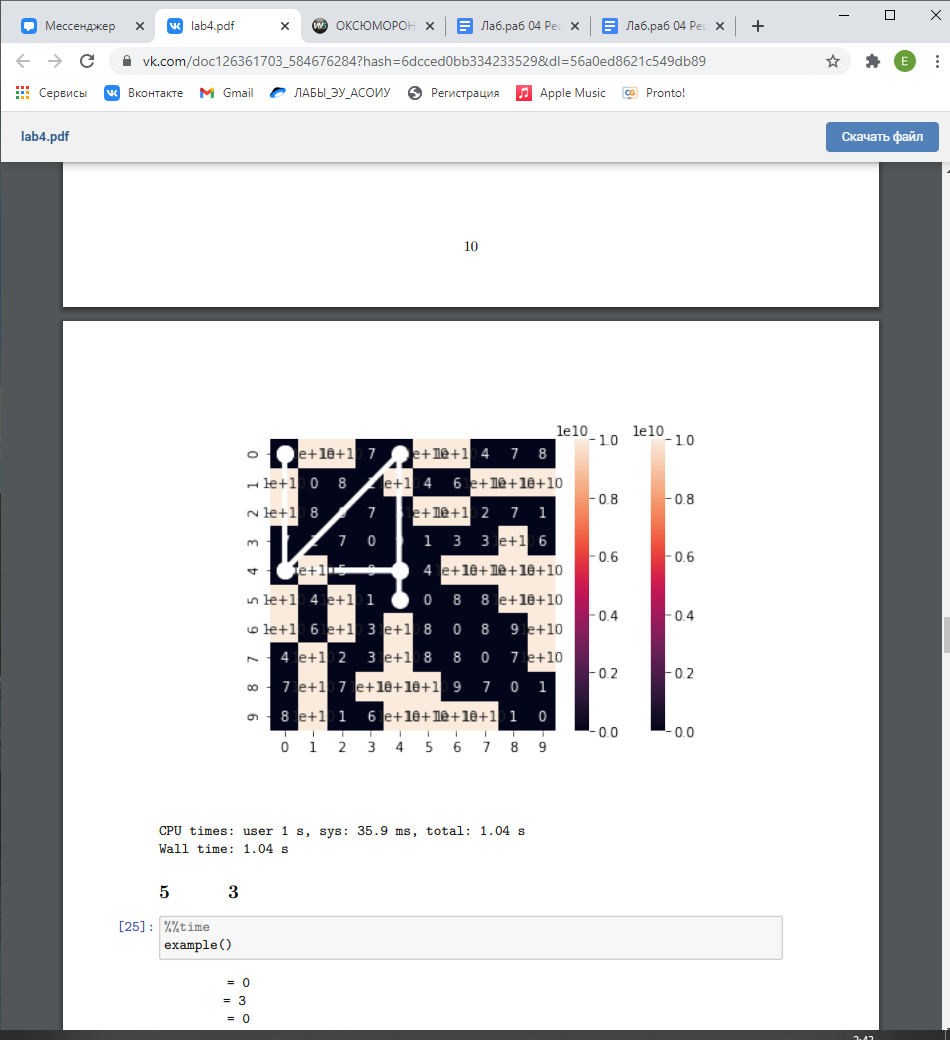
Эксперимент 1

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 1
* Конечный компьютер = 4
* Результат = [1, 5, 5, 6, 6, 6, 6, 4] с длинной 4



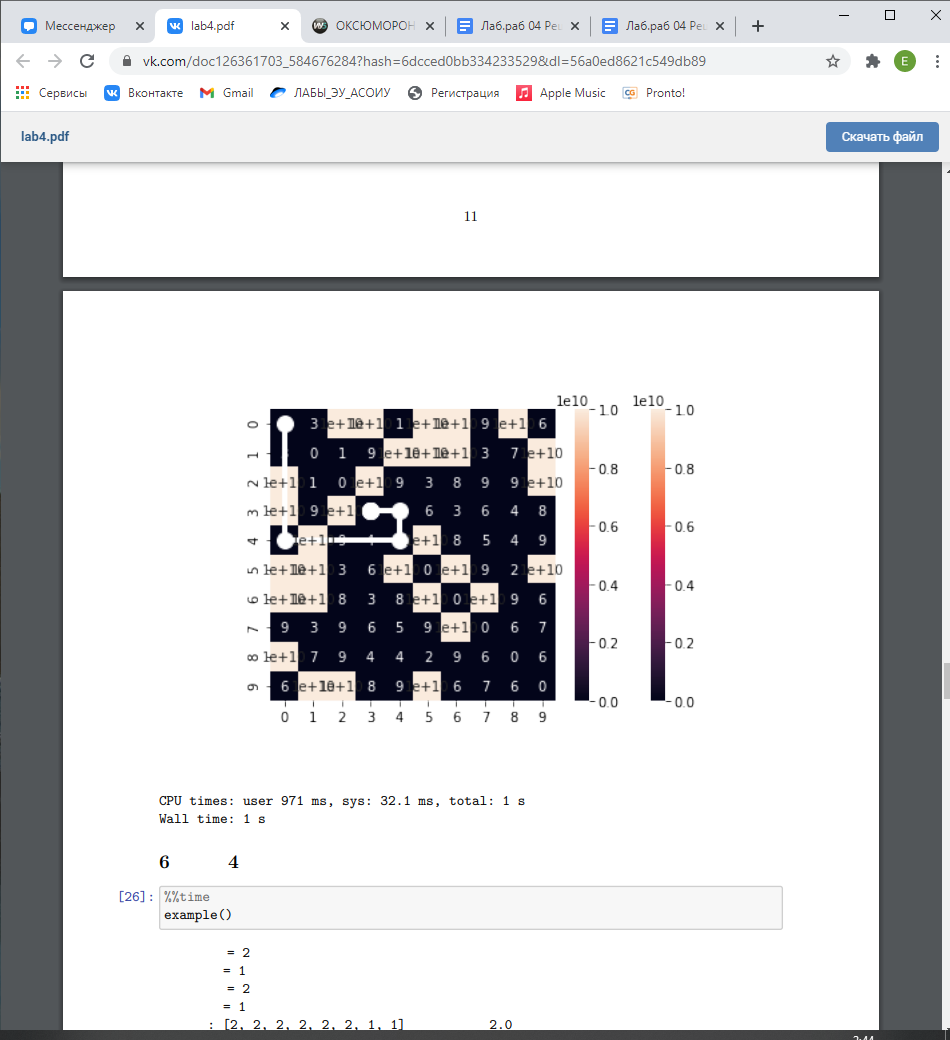
Эксперимент 2

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 0
* Конечный компьютер = 5
* Результат = [0, 0, 0, 4, 4, 0, 4, 4] с длинной 7



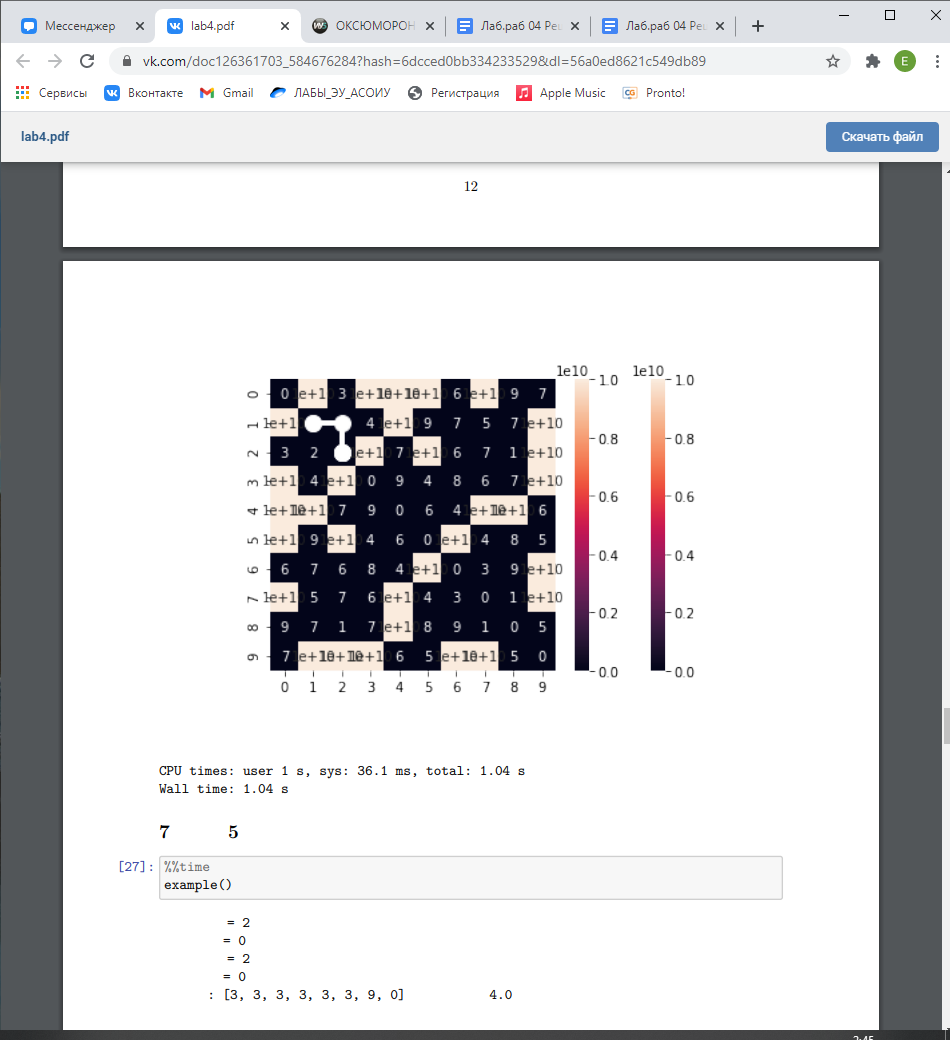
Эксперимент 3

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 0
* Конечный компьютер = 3
* Результат = [0, 4, 4, 3, 3, 3, 3, 3] с длинной 5



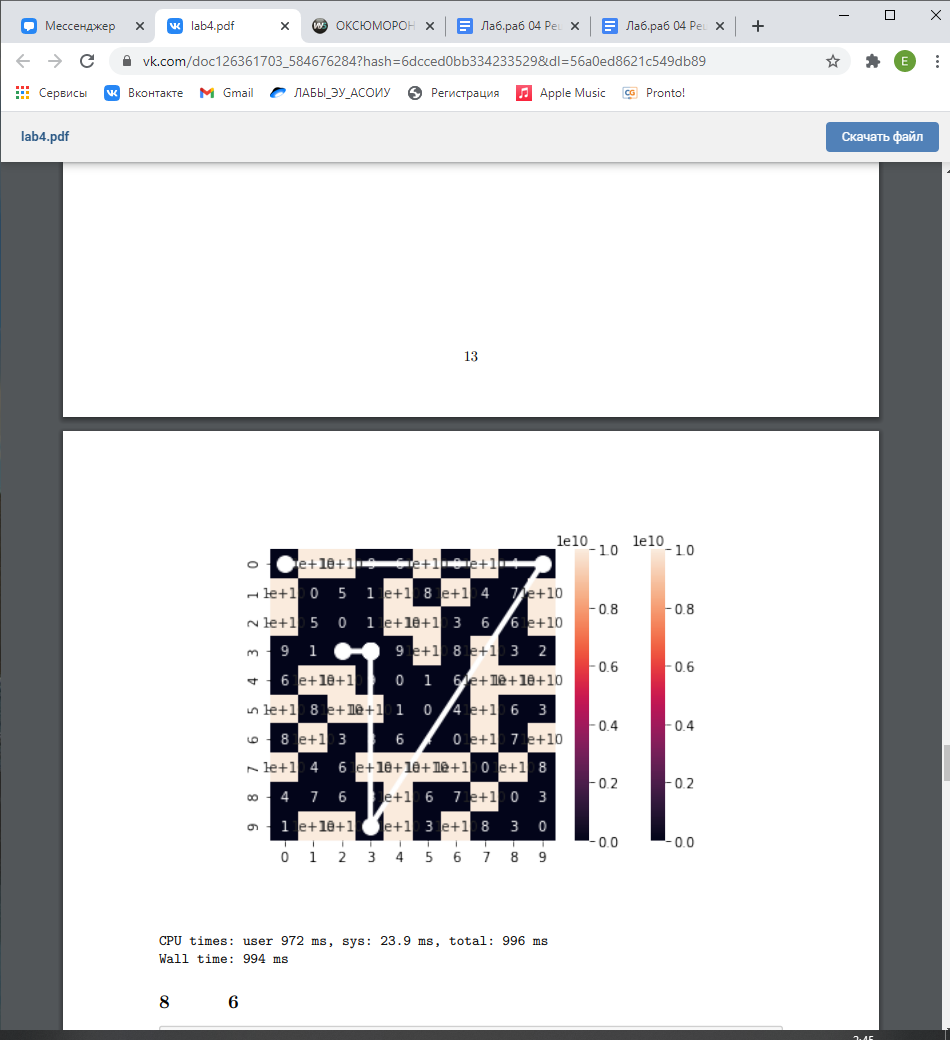
Эксперимент 4

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 2
* Конечный компьютер = 1
* Результат = [2, 2, 2, 2, 2, 2, 1, 1] с длинной 2



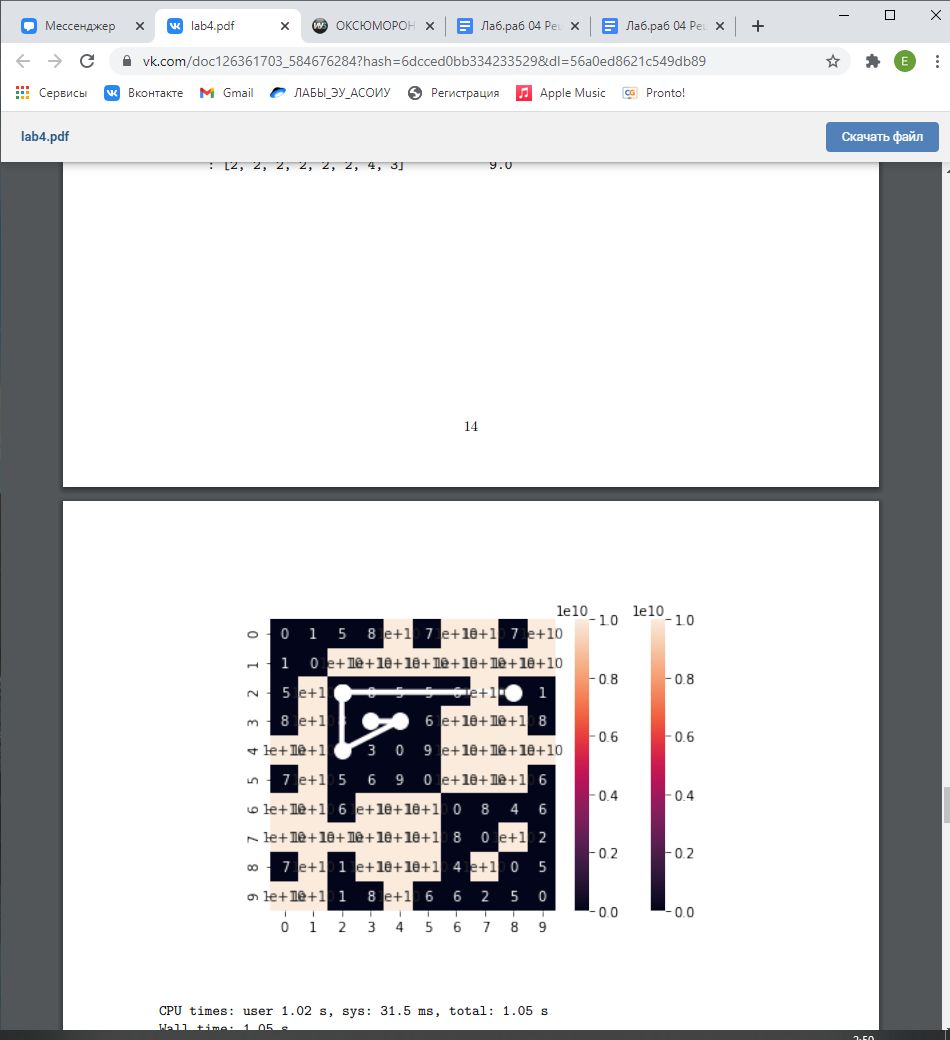
Эксперимент 5

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 2
* Конечный компьютер = 0
* Результат = [3, 3, 3, 3, 3, 3, 9, 0] с длинной 4



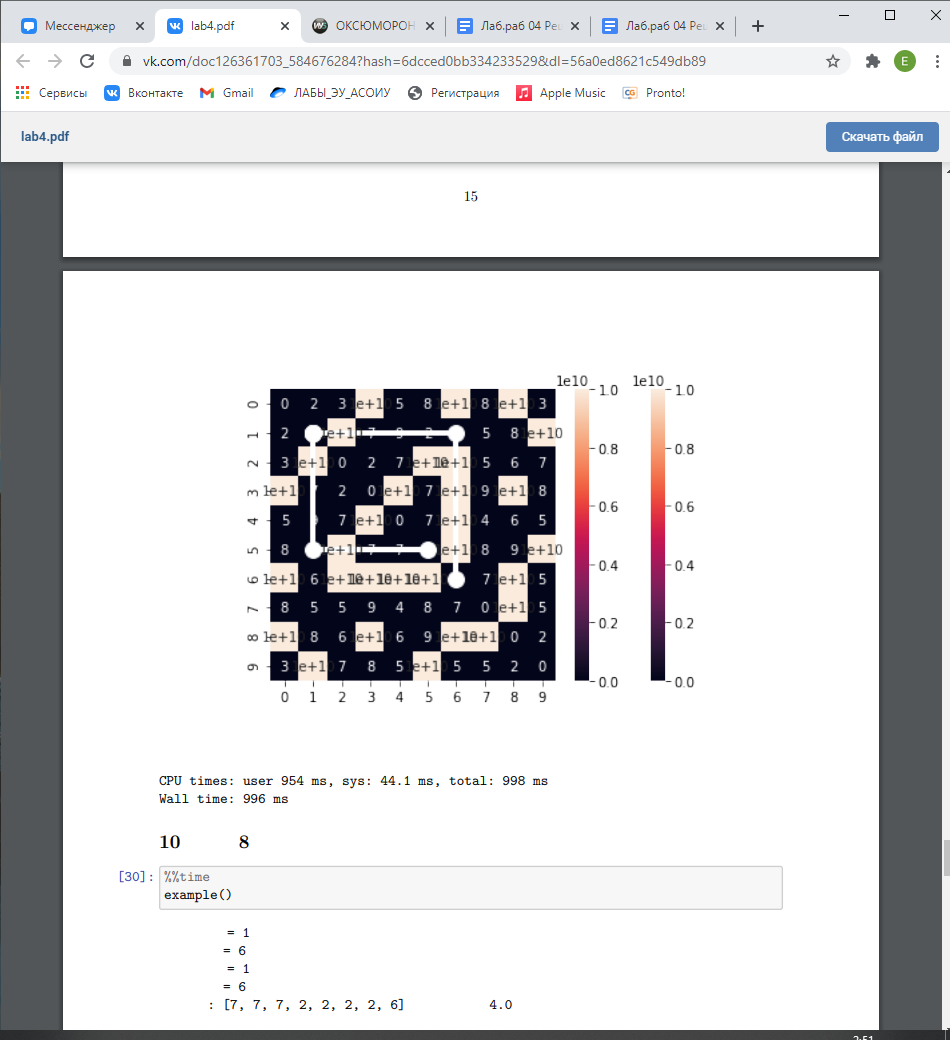
Эксперимент 6

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 8
* Конечный компьютер = 3
* Результат = [2, 2, 2, 2, 2, 2, 4, 3] с длинной 9



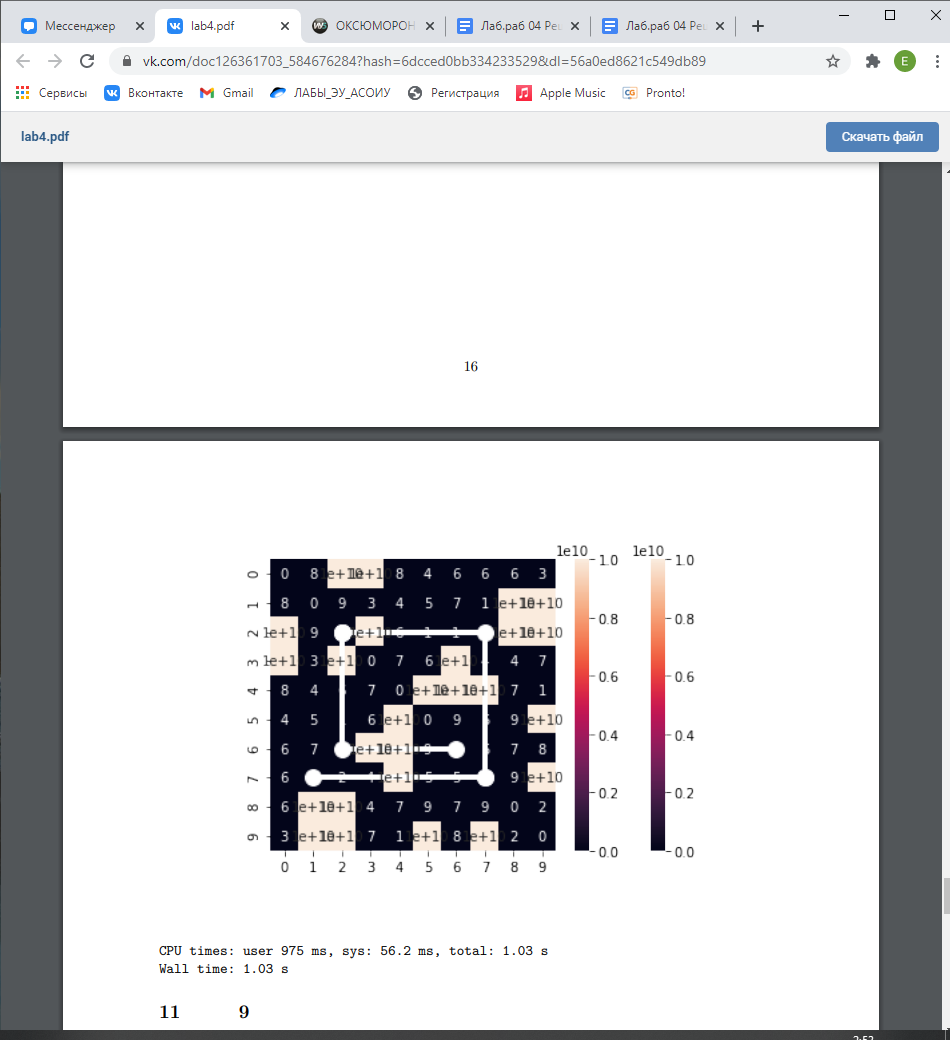
Эксперимент 7

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 6
* Конечный компьютер = 5
* Результат = [6, 1, 1, 1, 1, 1, 1, 5] с длинной 8



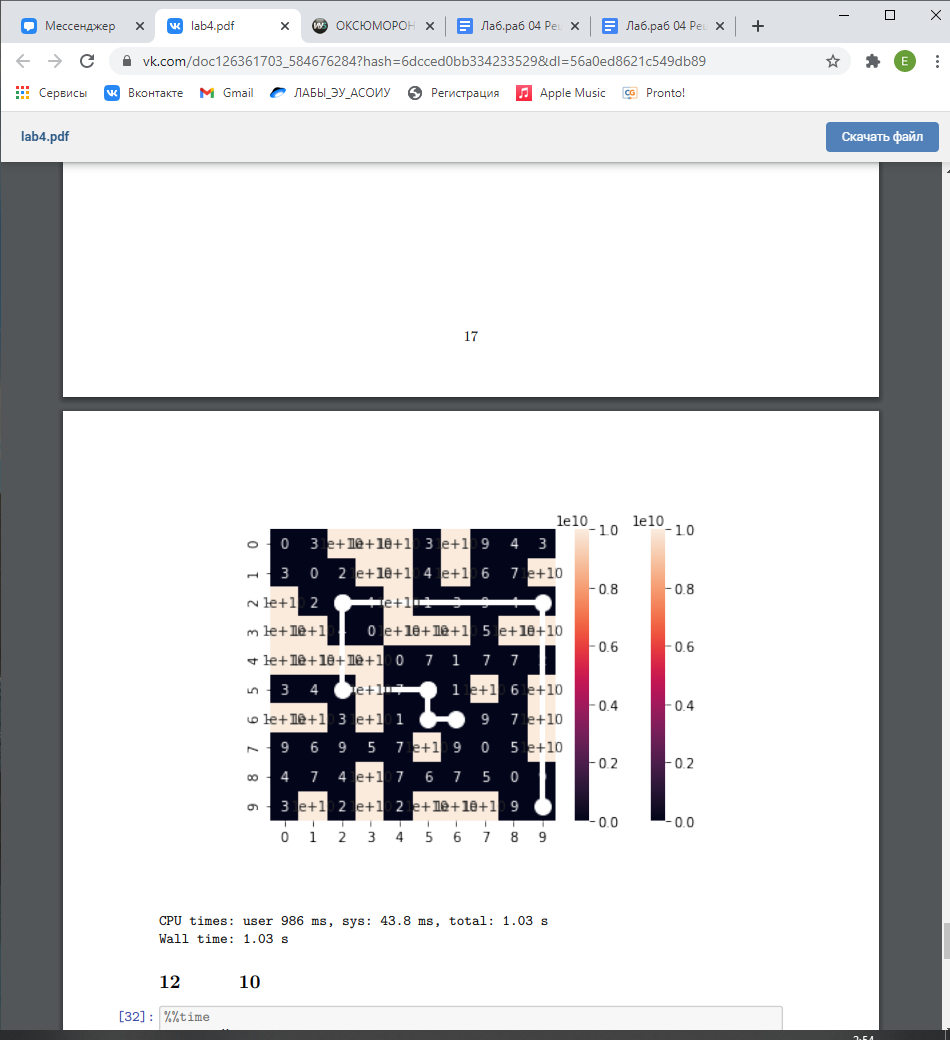
Эксперимент 8

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 1
* Конечный компьютер = 6
* Результат = [7, 7, 7, 2, 2, 2, 2, 6] с длинной 4



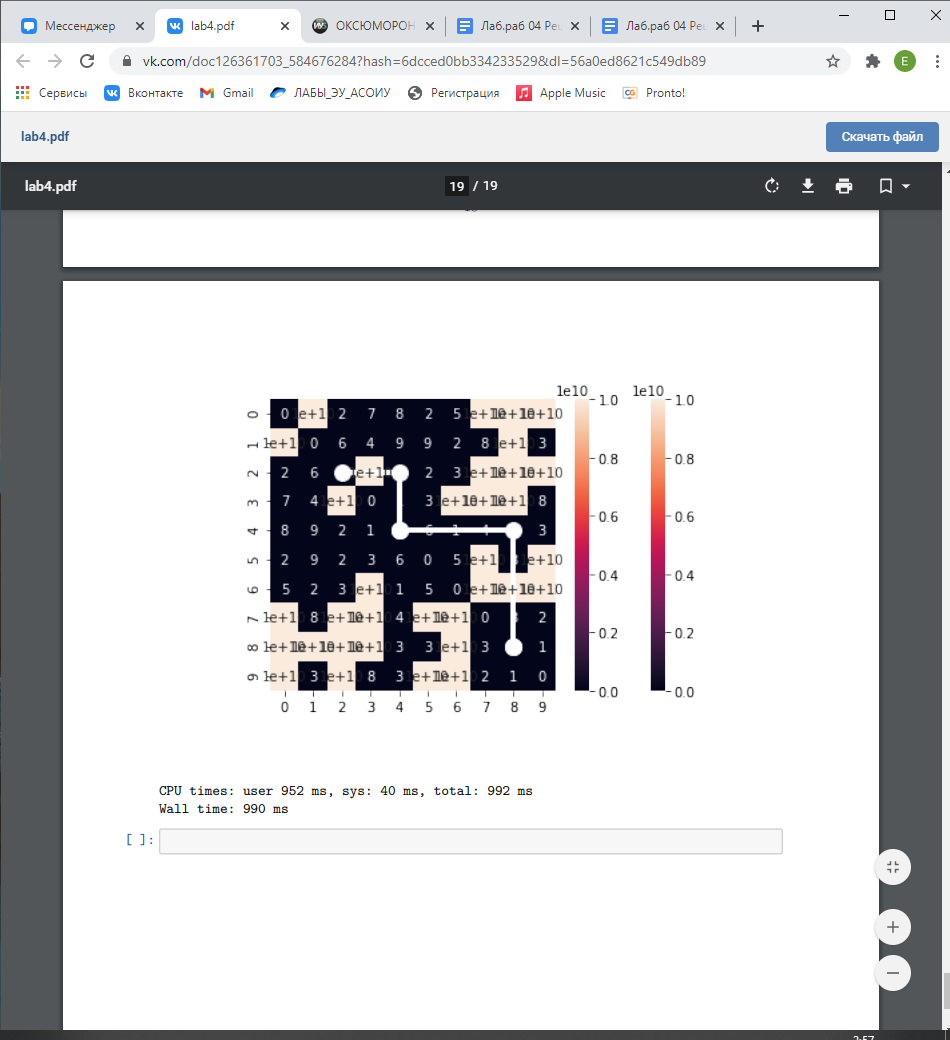
Эксперимент 9

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 9
* Конечный компьютер = 6
* Результат = [9, 2, 2, 2, 5, 5, 5, 6] с длинной 4



Эксперимент 10

* Количество компьютеров в сети = 10
* Соединение между двумя компьютерами отсутствует с вероятностью 40%
* Размер начальной популяции = 10000
* Эры = 10000
* Выход в случае, если минимальное значение расстояния не изменяется в течение 100 эр
* Вероятность мутации = 1%
* Начальный компьютер = 8
* Конечный компьютер = 2
* Результат = [8, 8, 8, 4, 4, 4, 4, 2] с длинной 5



1. **Выводы**

В результате проведенных экспериментов видно, что генетические алгоритмы не всегда находят оптимальное решение, однако, как правило, предлагают решения, которые не сильно уступают оптимальному. Это может происходить из-за недостаточно большой популяции, однако, при увеличении популяции сильно вырастает вычислительная сложность и, как следствие, время выполнения алгоритма.

1. **Используемая литература.**

Методические указания к курсу МППР.