

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

ОТЧЕТ
по учебной практике
по теме «Генетические алгоритмы»
Тема: Задача о кратчайшем пути.

Студенты гр. 0382

Преподаватель

Ильин Д.А.
Морева Е.С.
Смирнов И.А.
Жангиров Т. Р.

Санкт-Петербург
2022

Цель работы

Разработать и реализовать программу, решающую одну из оптимизационных задач (файл “Варианты”) с использованием генетических алгоритмов (ГА), а также визуализирующая работу алгоритма.

Задача

Вариант 10

Задача о раскраске графа (задача поиска раскраски графа с наименьшим кол-вом цветов).

Необходимо реализовать программу, которая для заданного графа выдаёт наименьшее кол-во цветов, в которые можно покрасить весь граф.

Входные данные:

- Список вершин
- Список ребер

Выполнение работы

Используемый язык программирования: Python 3.

Выполнение работы

1. Выбор метода решения задачи.

Будем начинать с раскраски в k цветов (k - количество вершин), наши хромосомы (представители) - вектора длиной k с числами от 1 до проверяемого числа цветов (собственно раскраска, на 3 месте стоит 4 - вершина с индексом 3 покрашена в четвёртый цвет) которые мы скрещиваем и мутируем какое-то количество поколений и если получаем сходимость 0 (граф раскрашен), то переходим к числу цветов меньшего, нежели предыдущее на 1. Если спустя некоторое кол-во поколений сходимость не получили, то делаем вывод, что последнее значение, для которого сходимость нашли, является минимальным.

Первично выбранные параметры ГА:

- вероятность кроссинговера - 90%
- вероятность мутации - 0.5%
- размер популяции - 30 особей
- способ отбора - совмещение элитарного отбора и отбора с усечением
- кодирование особей - список с номерами цветов для вершин с номером, равным индексу цвета в списке, цвета в списке имеют номера порядка, в котором встречаются (то есть каждая раскраска задана однозначно, обмен цветов номерами не учитывается)
- значение приспособленности - количество правильно окрашенных ребер, при приспособленности равной числу рёбер достигнута правильная раскраска
- модель ГА — островная

2. Формирование прототипа GUI.

На рисунке 1 представлен вид программы после запуска. Пользователь может выбрать одну из следующих команд:

- 1) Создать вершину — при выборе этой команды, наведя на рабочую область(под панелью инструментов, далее — рабочая область) можно создать вершину, она появится в месте нажатия.
- 2) Удалить вершину — удаляет вершину, которую выберет пользователь, посредством нажатия на неё.
- 3) Создать ребро — создаёт ребро между двумя выбранными вершинами.
- 4) Удалить ребро — работает аналогично удалению вершины.

Результат выполнения команд выше представлен на рисунке 2.

Так же рабочую область можно приблизить или отдалить, для этого служат две кнопки с лупами, с плюсом приближает, с минусом отдаляет.

Зелёный треугольник запускает ГА для заданного на данный момент графа, находящего в рабочей области.

Зелёный жучок является кнопкой запуска с более подробными переходами. Данная кнопка вызывает окно(Рисунок 3), в котором выбирается шаг, с которым будет выводиться раскраска(количество всех раскрасок обработанных программой может быть очень большим и дабы иметь возможность не выводить все промежуточные раскраски вводится шаг).

Промежуточный результат работы программы представлен на рисунке 4.

В самом низу выводится результат — минимальное кол-во цветов для раскраски графа.

После работы программы вершины окрашиваются в определённую раскраску подходящую требованиям задачи и содержащую выведенное кол-во цветов(Рисунок 5).



Рисунок 1.

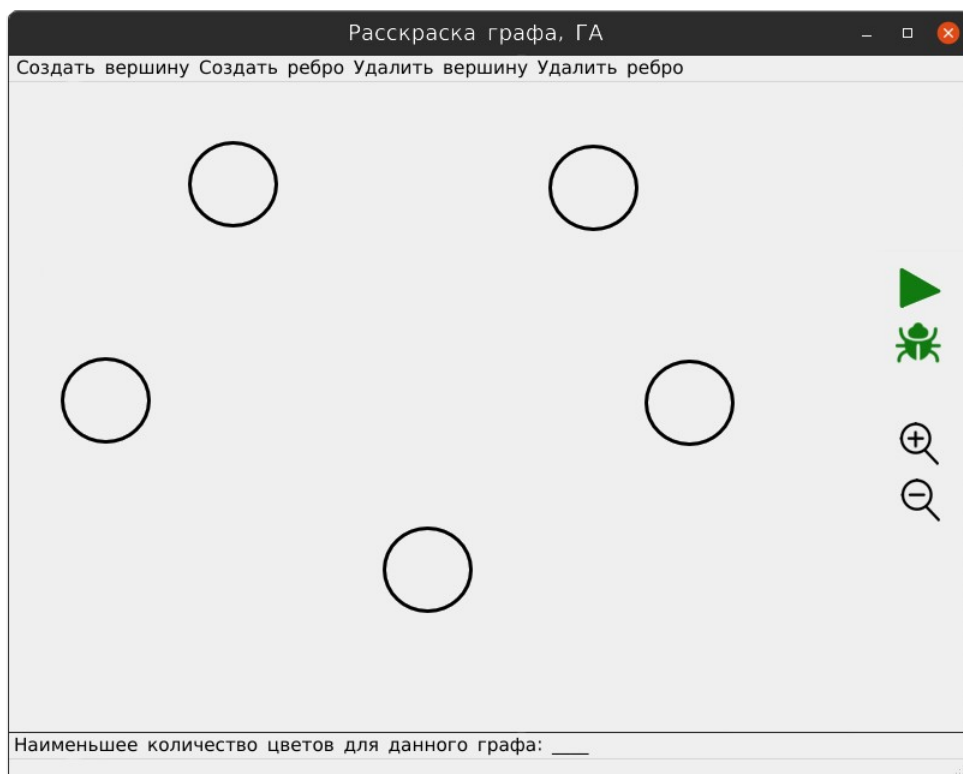


Рисунок 2.

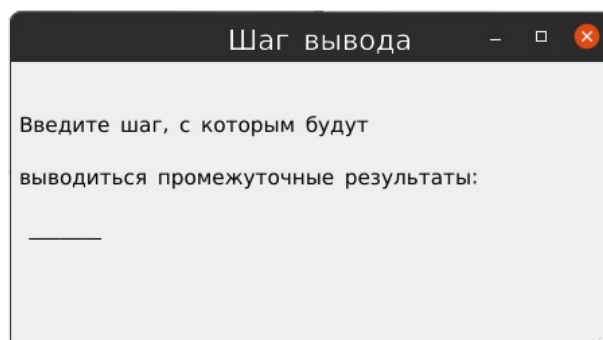


Рисунок 3.

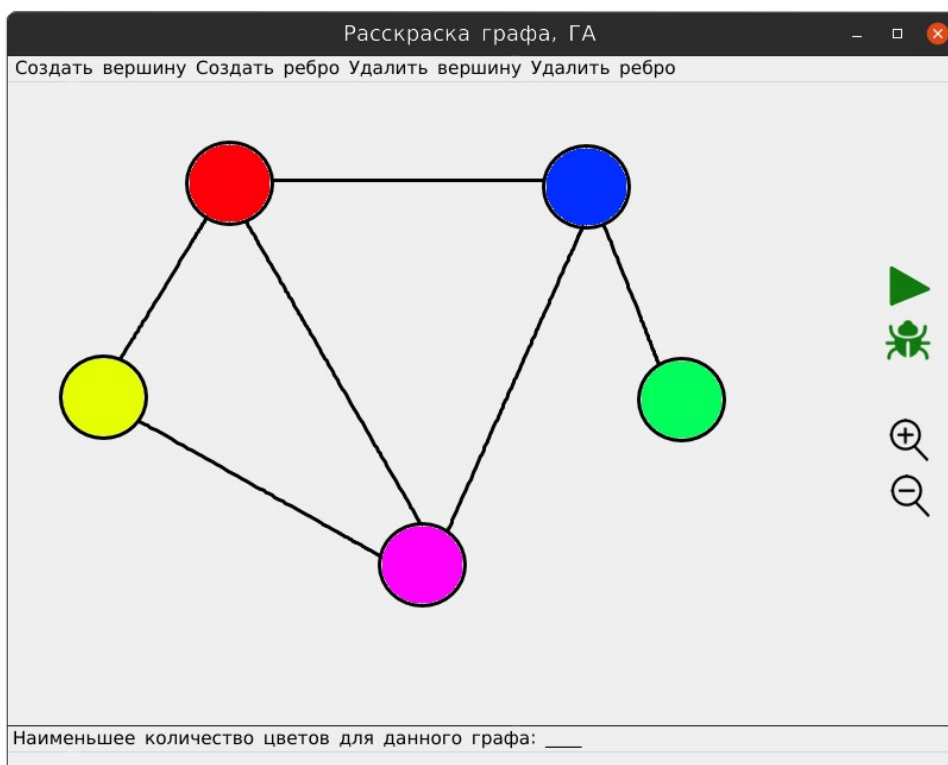


Рисунок 4.

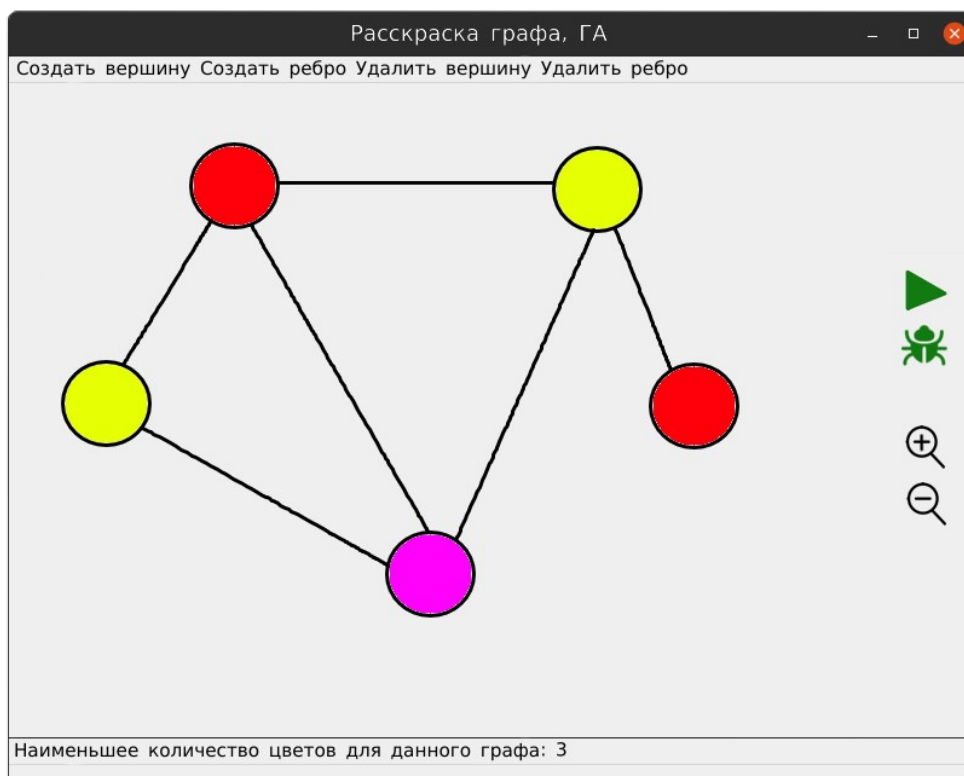


Рисунок 5.