**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**Факультет прикладной математики – процессов управления**

**отчет**

**по лабораторной работе №4**

**по дисциплине «Алгоритмы и структуры данных»**

**на тему «Исследование генетического алгоритма. Изучение различных кодировок генотипа»**

**3 вариант**

|  |  |  |
| --- | --- | --- |
| Студент гр. 22.Б16 |  | Гареев К.В. |
| Преподаватель |  | Дик А.Г. |

**Санкт-Петербург**

**2023 г.**

Оглавление

[1.Цель работы 3](#_Toc154452941)

[2. Задачи 3](#_Toc154452942)

[3. Теоретическая часть 3](#_Toc154452943)

[4. Принцип работы алгоритма 4](#_Toc154452944)

[5.Описание программы 5](#_Toc154452945)

[5.1 Описание классов 5](#_Toc154452946)

[5.2 Описание функций 5](#_Toc154452947)

[6. Рекомендации пользователя 7](#_Toc154452948)

[7. Рекомендации программисту 7](#_Toc154452949)

[8.Контрольный пример 8](#_Toc154452950)

[9.Анализ в зависимости от входных условий 8](#_Toc154452951)

[10. Вывод 9](#_Toc154452952)

# 1.Цель работы

Цель работы заключается в создании программы, использующую генетический алгоритм для поиска минимального значения определенной математической функции

# 2. Задачи

1.Провести анализ основных теоретических концепций в области генетических алгоритмов и методов их оптимизации.

2.Изучить два принципиальных подхода к кодированию хромосом в генетических алгоритмах.

3.Написать программу, цель которой - поиск минимального значения функции.

4.Провести тестирование программы на выбранной тестовой функции с целью оценки ее эффективности.

5.Проанализировать работу алгоритма в зависимости от методов оптимизаций, способов кодирования хромосом, параметров: размера популяции, вероятности мутаций и др.

# 3. Теоретическая часть

Генетический алгоритм представляет собой метод оптимизации, вдохновленный естественным отбором и принципами генетики. Его основные механизмы - мутация, скрещивание и селекция - направлены на поиск оптимальных решений задач оптимизации.

Основные функции в генетическом алгоритме включают:

Инициализация популяции: Создание начального набора хромосом с случайными генетическими значениями.

Расчет пригодности (fitness): Оценка каждой хромосомы путем вычисления значения функции приспособленности.

Селекция: Выбор лучших хромосом для следующего поколения на основе их пригодности.

Скрещивание: Образование новых хромосом путем комбинирования генетической информации двух родителей.

Мутация: Случайное изменение генетических значений с целью внесения разнообразия в популяцию.

В генетическом алгоритме применяется стратегия эволюционного скрещивания, которая включает в себя взвешенное сочетание генетической информации родителей с коэффициентом α. Этот параметр позволяет более гибко управлять влиянием каждого родителя на потомство.

Существуют два основных типа кодирования хромосом:

Бинарное кодирование: Гены хромосомы представлены в виде бинарных строк, что полезно для решения задач с дискретным пространством поиска.

Вещественное кодирование: Гены хромосомы представлены в виде вещественных чисел, позволяя работать с непрерывным пространством поиска.

# 4. Принцип работы алгоритма

Ввод основных характеристик через интерфейс программы: Пользователь задает параметры: выбранная функция, вероятность мутации, количество хромосом, минимальное и максимальное значения гена, количество поколений для выполнения алгоритма, и тип кодирования для хромосом.

Запуск генетического алгоритма с установленными параметрами: Программа запускает генетический алгоритм, используя заданные пользователем параметры для инициализации, проведения селекции, скрещивания, мутации и определенного количества поколений.

Отображение лучшего решения и точки: после завершения алгоритма программа выводит лучшее найденное решение и его координаты. Также программа выводит таблицу, содержащую информацию о популяции.

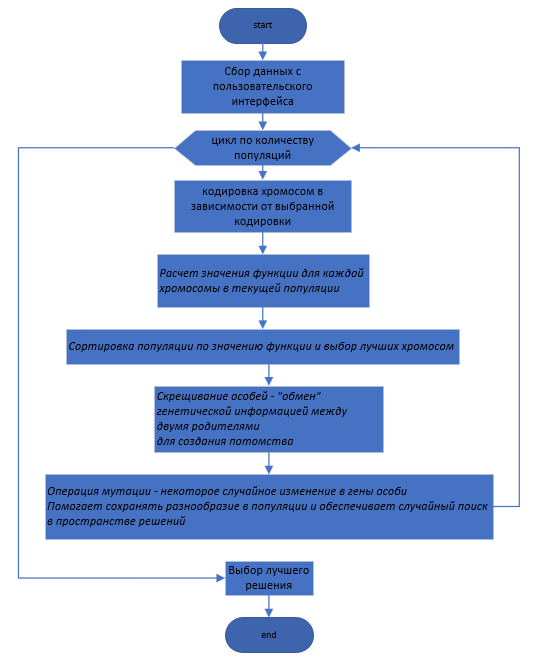


Рисунок 4.1. Блок схема программы

# 5.Описание программы

# 5.1 Описание классов

Таблица 5.1.1. Описание классов

|  |  |
| --- | --- |
| Класс | Описание класса |
| Genetic | Класс, реализующий генетический алгоритм. |
| GeneticAlgorithmInterface | Класс, реализующий интерфейс генетического алгоритма на Tkinter. |

# 5.2 Описание функций

Таблица 5.2.1. Описание функций

|  |  |
| --- | --- |
| Функция | Описание функций |
| calculate\_function(function\_choice, x1, x2) | Данная функция представляет собой целевую функцию, которую генетический алгоритм пытается минимизировать. Принимает на вход параметры x1 и x2 и, в зависимости от выбора функции (function\_choice), вычисляет значение целевой функции. |
| run(self) | Метод класса Genetic, отвечающий за запуск генетического алгоритма. Итерируется через заданное количество поколений, выполняя этапы: оценка приспособленности, отбор, скрещивание и мутация. |
| run\_algorithm(self) | Метод класса GeneticAlgorithmInterface, отвечающий за запуск генетического алгоритма при нажатии кнопки "Calculate". |
| fill\_table(self, results) | Метод класса GeneticAlgorithmInterface, заполняющий таблицу результатами популяции. |
| binary(self) | Метод класса Genetic, реализующий бинарное кодирование хромосом. |
| real(self) | Метод класса Genetic, реализующий вещественное кодирование хромосом. |
| initialize\_population(self) | Метод класса Genetic, отвечающий за начальное создание популяции в зависимости от выбранного типа кодирования. |
| fitness(self) | Метод класса Genetic, вычисляющий значения функции для каждой хромосомы в текущей популяции. |
| selection(self) | Метод класса Genetic, отбирающий лучшие хромосомы в текущей популяции. |
| crossover(self) | Метод класса Genetic, выполняющий операцию скрещивания между хромосомами. Для каждой пары хромосом выполняет скрещивание методом эволюционной стратегии. |
| mutation(self) | Метод класса Genetic, реализующий операцию мутации для поддержания разнообразия в популяции. Использует эволюционную стратегию для изменения генов хромосом с некоторой вероятностью. |
| best\_solution(self) | Метод класса Genetic, возвращающий лучшее решение в текущей популяции. |
| get\_population(self) | Метод класса Genetic, возвращающий информацию о популяции. |

# 6. Рекомендации пользователя

Поле *«Function»* позволяет выбрать математическую функцию из выпадающего списка. Поля *«Mytation probabolity (%)»*, *«Number of Chromocomes»*, *«Min Gene value»*, *«Max Gene value»*, *«Number of Generations»* — позволяют настроить параметры генетического алгоритма. В поле *«Encoding Type»* доступен выбор кодировок хромосом для алгоритма.

Кнопка *«Calculate»* отвечает за запуск работы программы. Далее лучшее решение и значение выводится в поле *«best result»*

Для завершения работы нажмите на крестик в левом верхнем углу.

# 7. Рекомендации программисту

Для запуска программы необходим Python версии не ниже 3.10.6, а также 64-битная операционная система Windows/Linux/macOS. Предварительно необходимо установить библиотеку numpy версии не ниже 1.24.2. Для работы с кодом необходим PyCharm версии не ниже 2022.2.1.

Минимальное необходимое место на диске: 0.5 МБ. Минимальное необходимое количество оперативной памяти: 100 МБ.

# 8.Контрольный пример

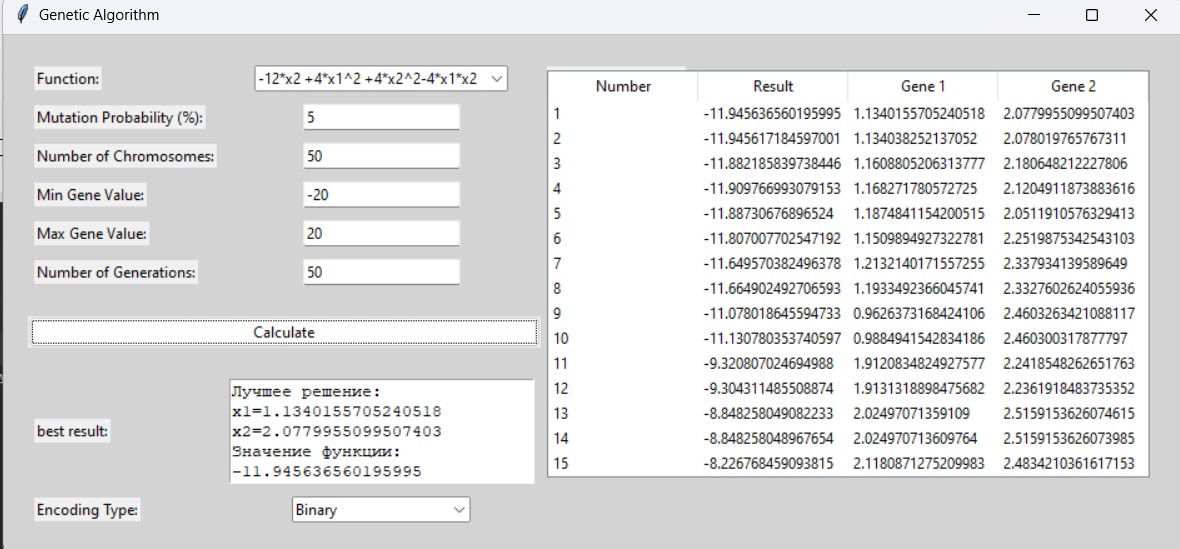


Рисунок 8.1. Пример результата работы алгоритма

# 9.Анализ в зависимости от входных условий

Тесты будут проводиться для вероятности мутации 5%. Диапазон изменения Генов от -20 до 20

Таблица 9.1. Результаты Бинарной кодировки

|  |  |  |  |
| --- | --- | --- | --- |
| Количество итераций | Размер популяции | Лучшее решение | Лучшее значение |
| 10 | 50 | (0,96;2,00) | -11,993 |
| 30 | 50 | (0,99;2,00) | -11,999 |
| 50 | 50 | (1,00;2,00) | -11,999 |

Таблица 9. Результаты вещественной кодировки

|  |  |  |  |
| --- | --- | --- | --- |
| Количество итераций | Размер популяции | Лучшее решение | Лучшее значение |
| 10 | 50 | (0,86;1,99) | -11,991 |
| 30 | 50 | (0,99;2,00) | -11,999 |
| 50 | 50 | (1,07;2,05) | -11,983 |

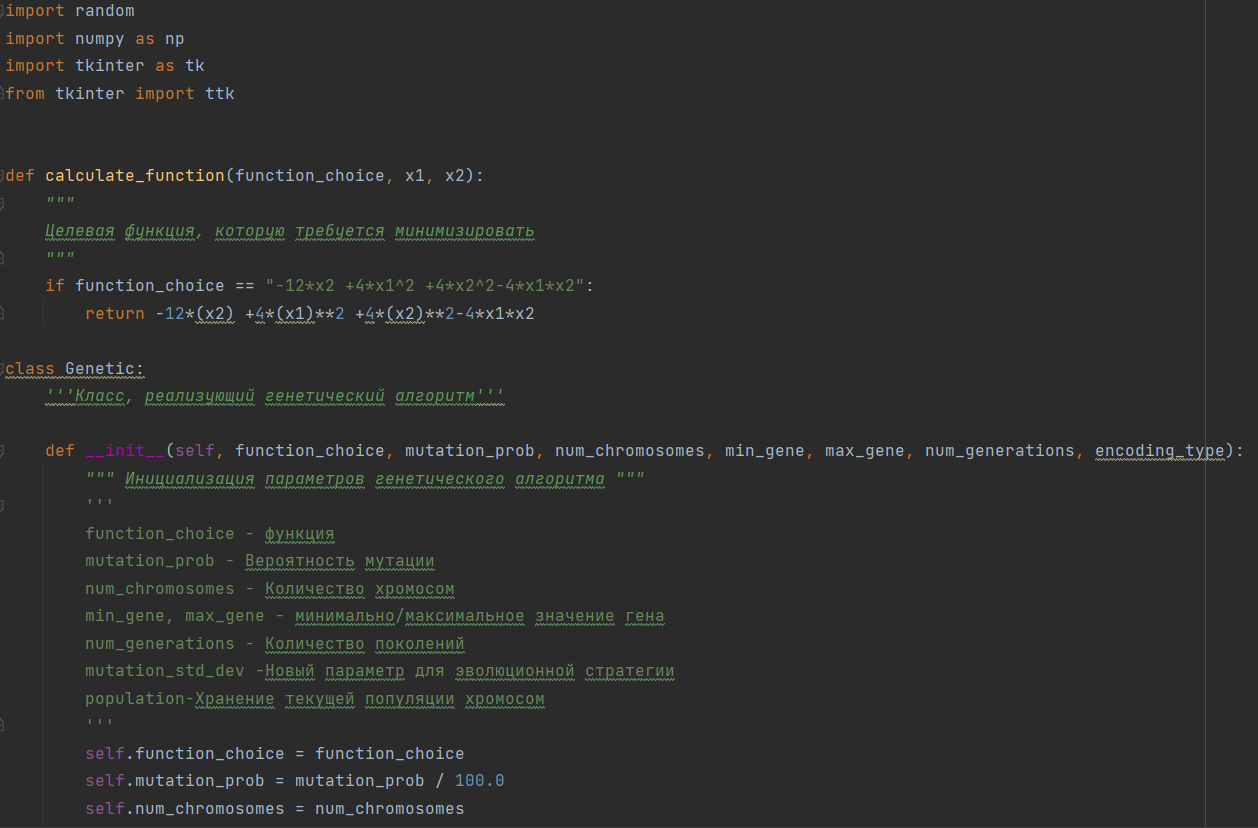
Анализируя полученные результаты можно сказать, что Бинарный вариант показал себя быстрее в плане сходимости и тем более оказался более устойчивым. Результаты тестов показывают, что даже с небольшим количеством итераций (10), бинарный геном способен предоставлять высококачественные решения.

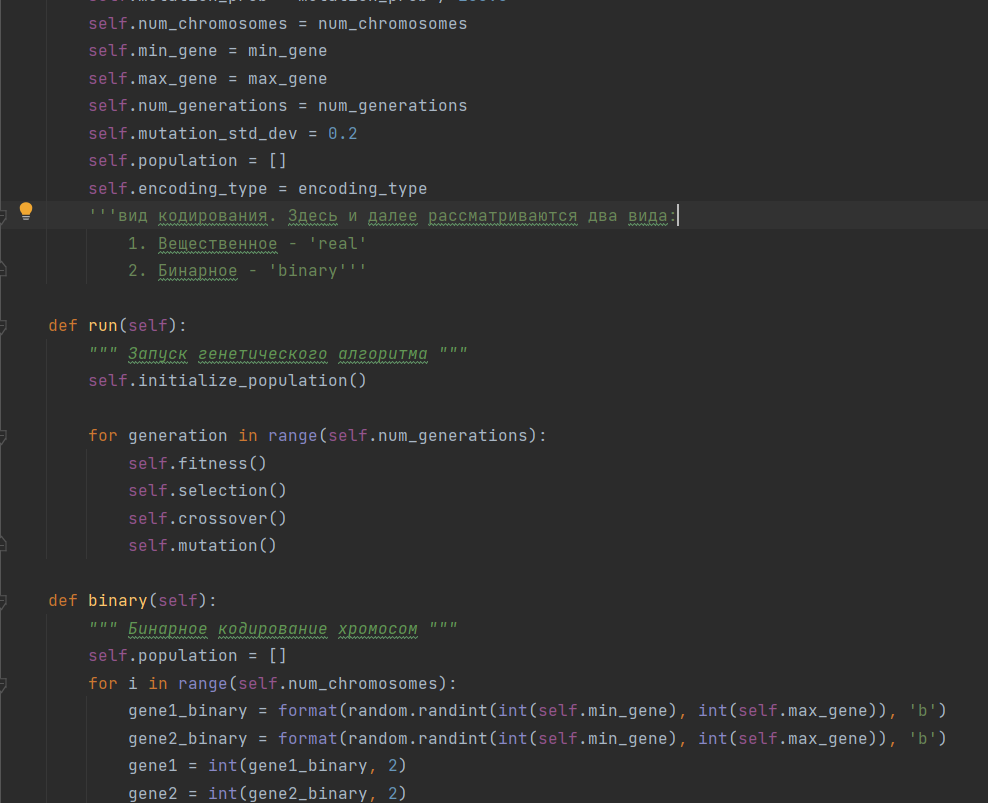
# 10. Вывод

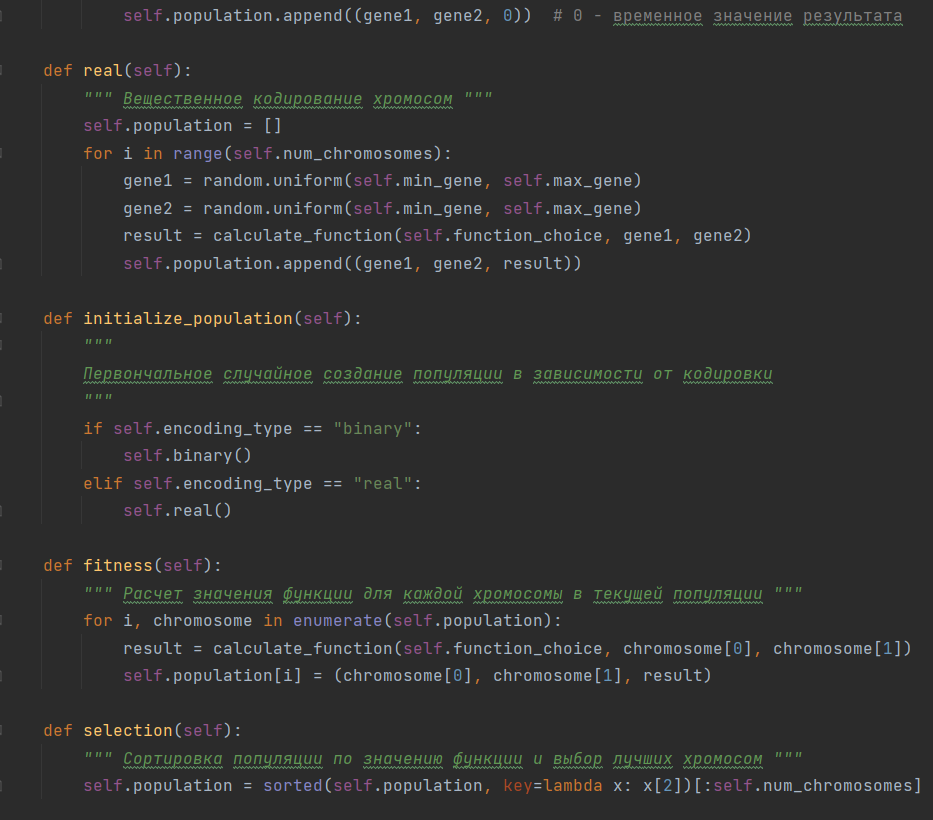
Разработанная программа, основанная на генетическом алгоритме для оптимизации математических функций, продемонстрировала значительную эффективность в решении задач оптимизации. Результаты экспериментов на тестовой функции подтверждают успешное функционирование генетического алгоритма в достижении оптимальных значений.

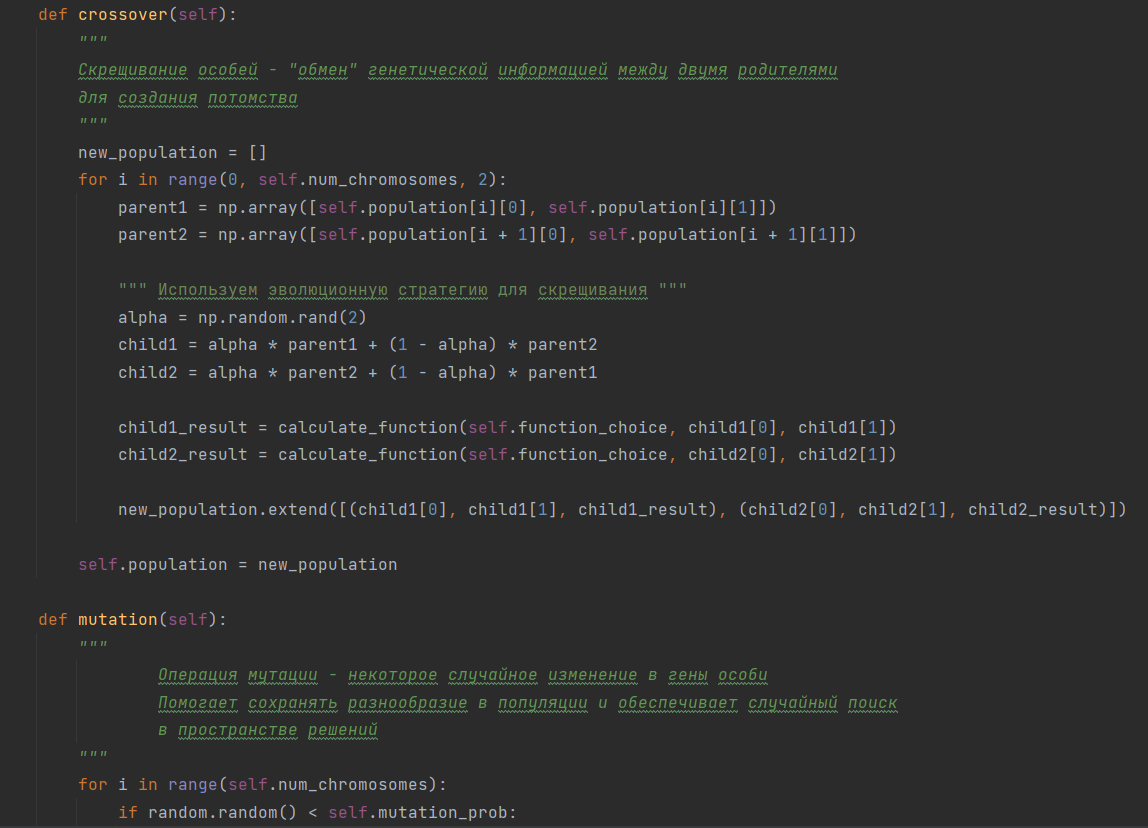
Основное внимание было уделено двум различным способам кодирования хромосом — бинарному и вещественному. Это дополнение позволяет пользователям выбирать подходящий тип кодировки в зависимости от характера задачи оптимизации, обеспечивая большую гибкость и универсальность использования генетического алгоритма.

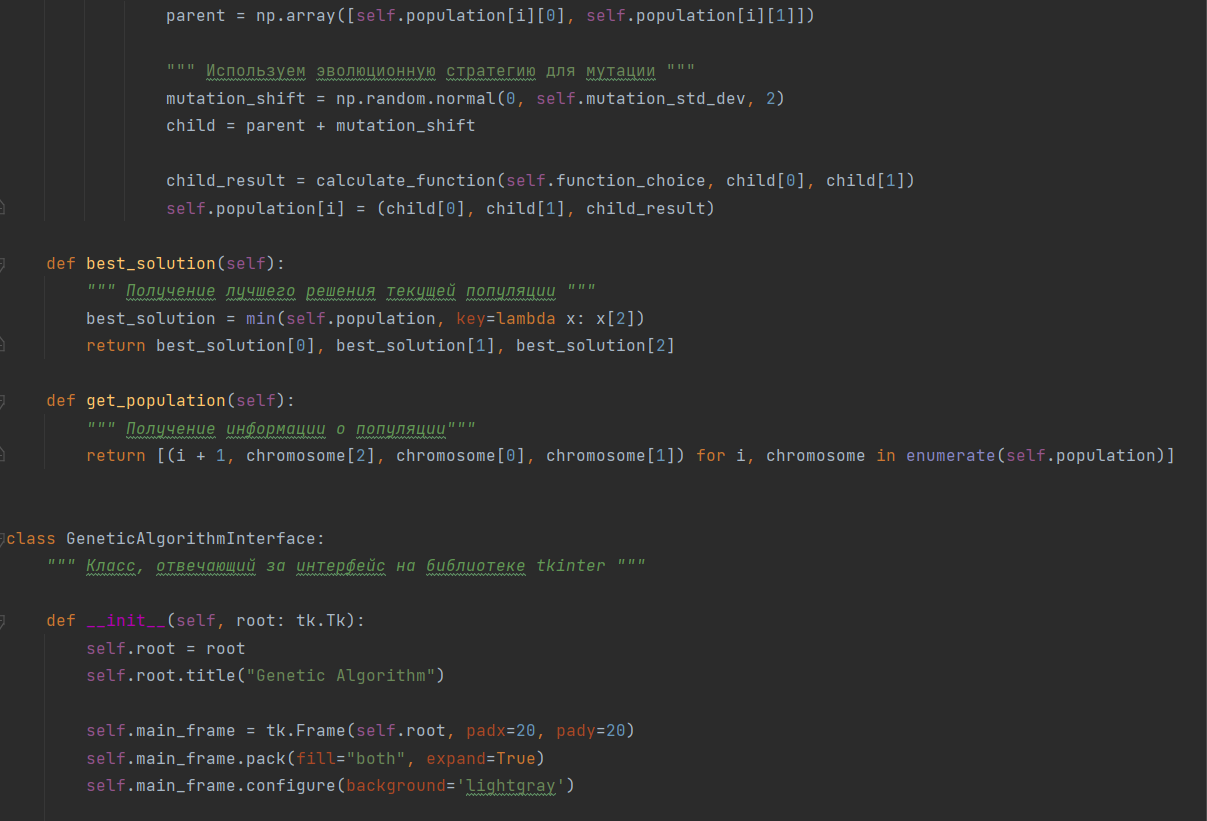
# 11.Листинг программы

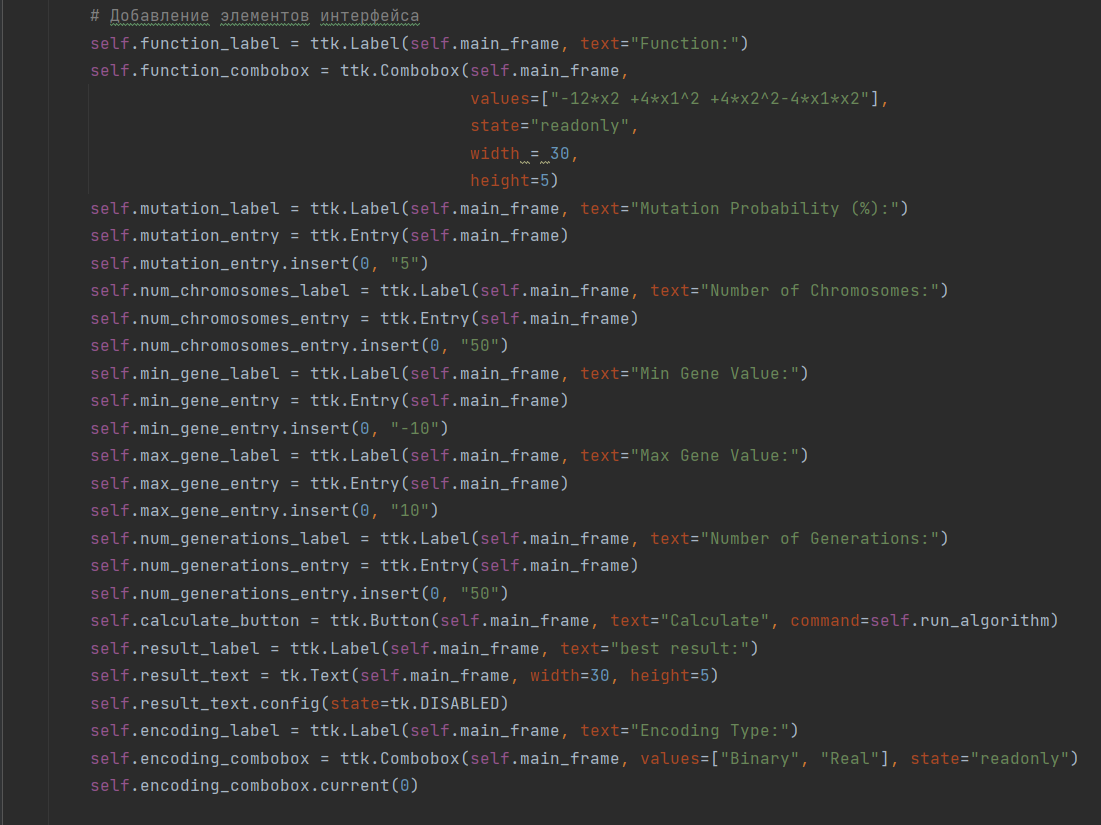


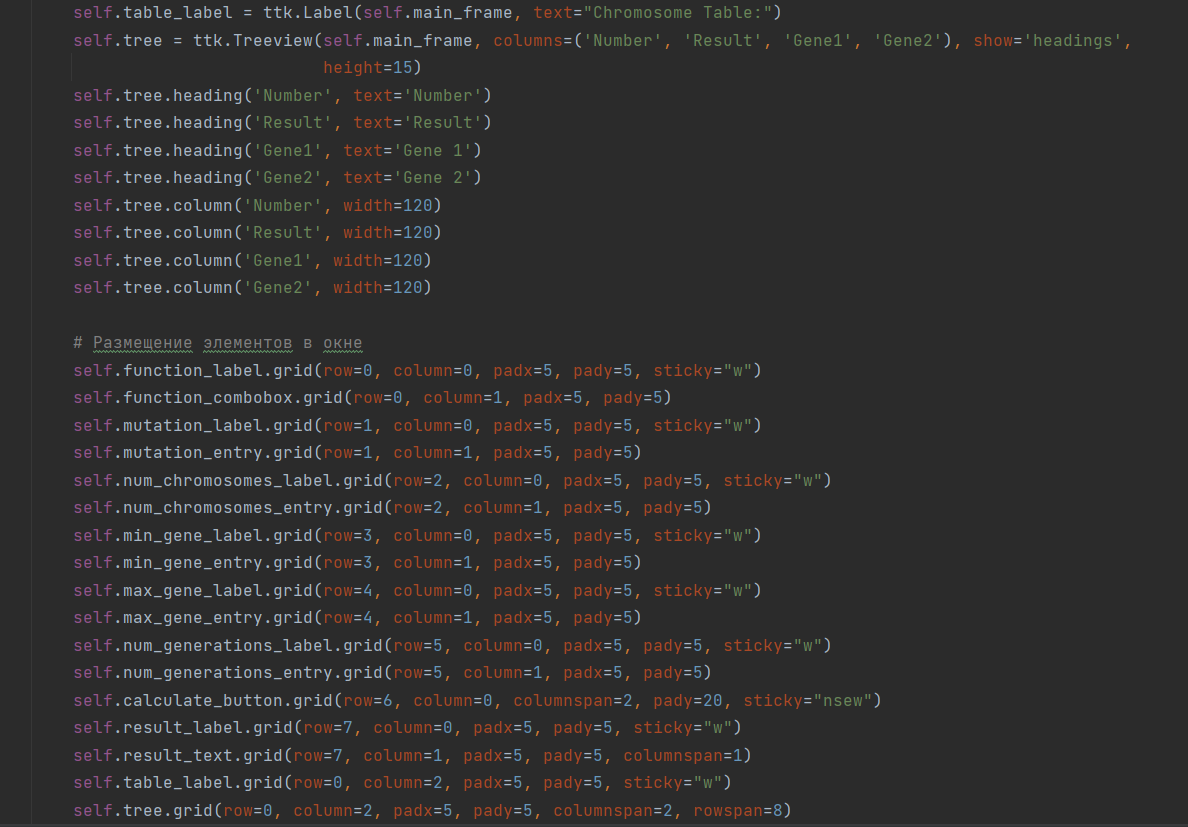


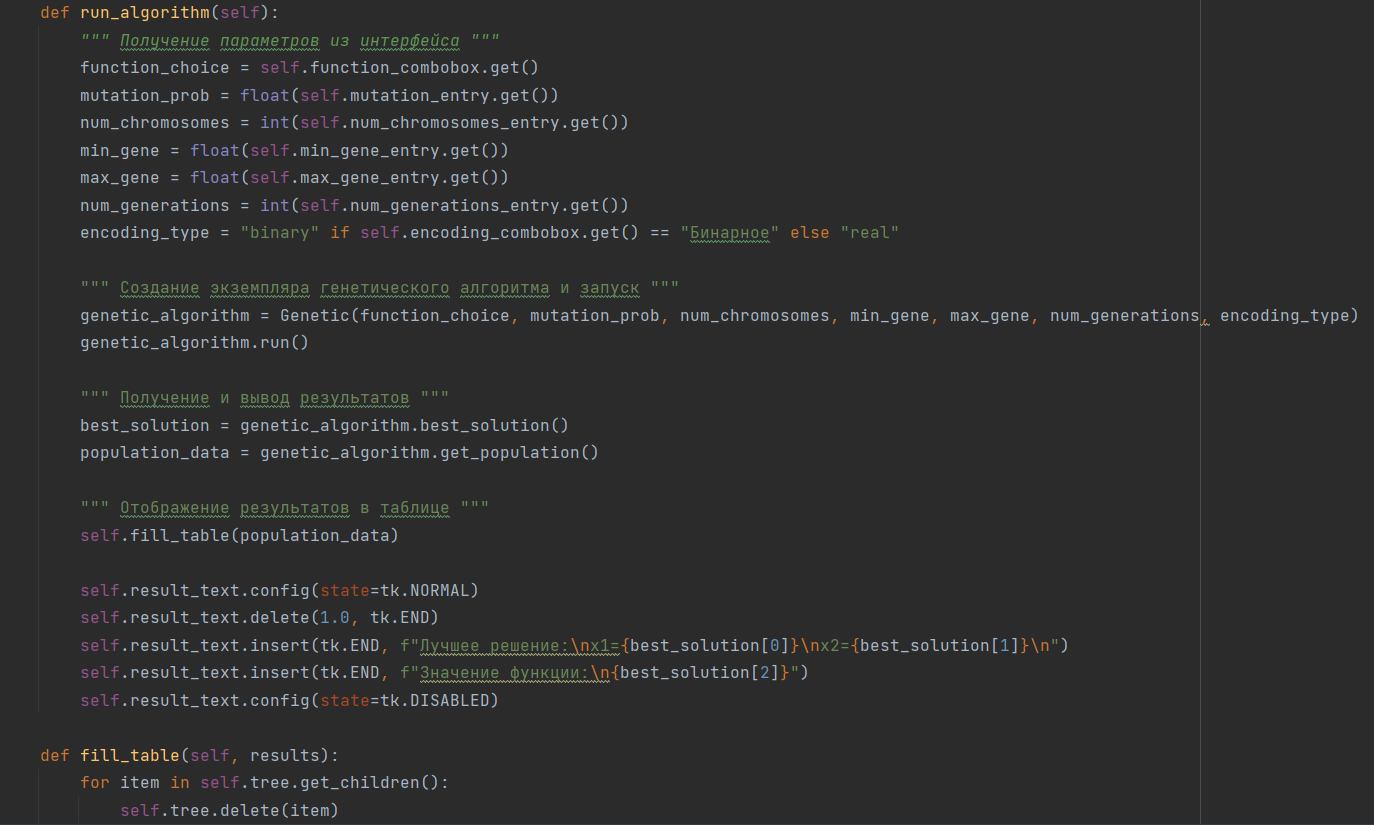


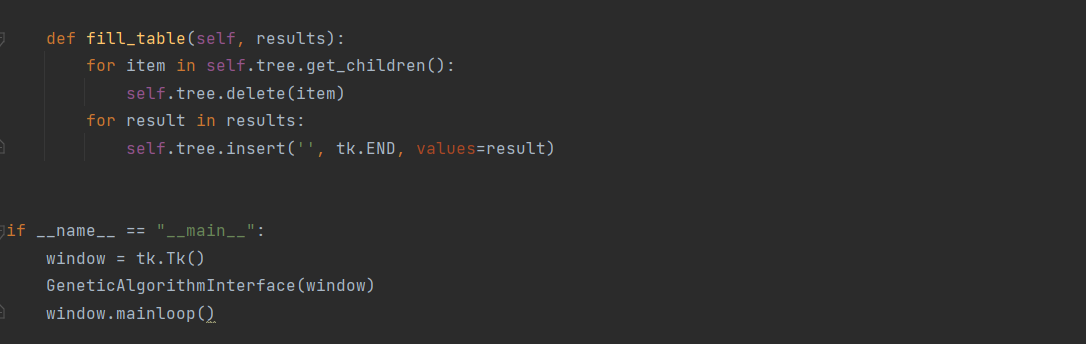












11.1. Листинг программы