

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

ОТЧЕТ

по учебной практике

Тема: реализация генетического алгоритма в задаче на нахождение
локального и глобального максимума функции

Автор		Ильин П.О.
-------	--	------------

Санкт-Петербург

2024

1. Введение

Наша команда приступила к разработке проекта, который включает реализацию интерфейса программы и определение используемых типов данных и методов. Цель проекта – создание приложения для решения задач оптимизации с использованием генетического алгоритма, а также разработка графического интерфейса пользователя (GUI) для визуализации работы алгоритма.

2. Распределение ролей в команде

Жданов Антон: Разработка генетического алгоритма и тестирование

Ильин Павел: Дизайн и реализация GUI

Борисов Иван В: Разработка генетического алгоритма, утилиты и интеграция кода

3. План проекта

GUI

GUI будет разрабатываться на основе фреймворка Qt C++. Основные требования к интерфейсу:

Отображение графика функции на заданном интервале $[l; r]$

Возможность перемещения между состояниями графика (сохранение и восстановление каждого решения)

Возможность пропуска промежуточных шагов с возможностью их сохранения (но без отображения)

Задание интервала и параметров функции:

Через выбор файла на компьютере

Через интерфейс GUI

Генетический алгоритм

Представление данных

Геном: Представляется в виде `std::uint64_t`, который мапится на интервал $[l; r]$, где 0 соответствует l , а 1111111111 соответствует r .

Поколение: Множество геномов `std::set<std::uint64_t>`.

Оценка особи

Функция: $f(x)$, максимизация значения полинома в точке x .

Отбор особей

Использование метода "вращения барабана".

Перекрещивание и мутация

Перекрещивание: Случайное перекрещивание участков геномов равной длины.

Мутация: Переворот (flip) определенных битов.

Вероятности:

Вероятность мутации `p_m`

Вероятность скрещивания `p_merge`

API

Реализация функции `next_generation(old_generation)` для создания нового поколения.

4. Связка модулей

Чтение данных

Чтение данных о полиноме из файла (формат данных будет согласован).

Сохранение графика.

Общая схема

Интерфейс программы (GUI) взаимодействует с генетическим алгоритмом (GA) через определенное API.

5. Структура классов

Общие классы

`Polynomial`: Представление полинома.

Классы, связанные с генетическими алгоритмами

`Genom`: Представление генома, оценка, сравнение и т.д.

`Generation`: Представление поколения, создание нового поколения, перекрещивание, мутация.

`Selection`: Наследуется от `ISelection` (различные алгоритмы селекции).

Классы GUI

Будут определены в следующем этапе разработки.
Прототип дизайна:

Start

Поиск локальных и глобального максимумов на отрезке полинома

Полином

Как задать:

$5+12x-3x^2+0.445x^3-0.2198x^4$

Отрезок

от

-30.5

до

50

Параметры алгоритма

Число генов на старте:

36

Число поколений:

36

Шанс скрещивания:

0,9

Шанс мутации:

0,9

Решить задачу

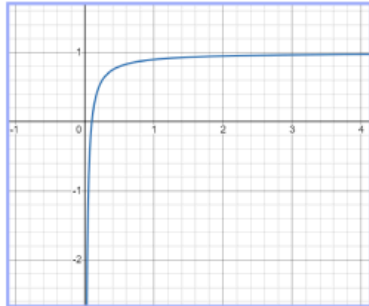
Process

Номер поколения: 6

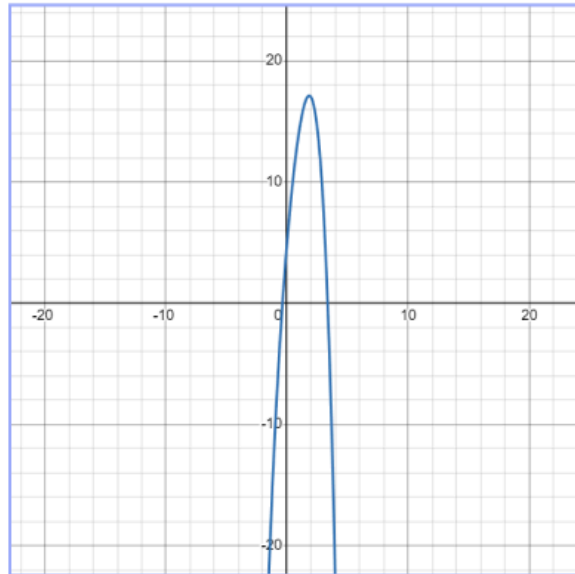
Топ 3 решений на данный момент:

1. $f(2,5)=17,5$
2. $f(2,4)=17,4$
3. $f(2,6)=17,4$

График функции качества решения



Пропустить

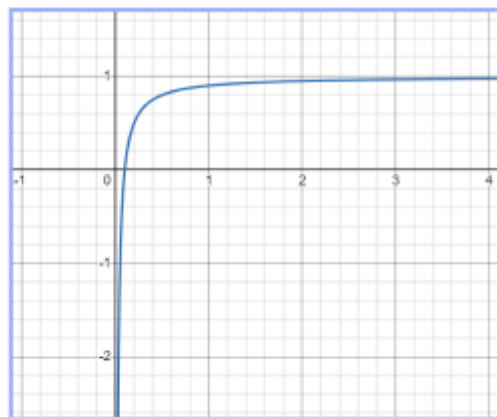


Result

График функции качества решения

Полученное решение

1. 2,49
2. 2,51
3. 34
4. 35
5. 36
6. 37



6. Тестирование

Для тестирования будет использоваться библиотека Google Test. Тесты будут охватывать основные функции генетического алгоритма и корректность взаимодействия между модулями.

7. Заключение

На данный момент была проведена работа по определению требований к проекту, распределению ролей в команде, планированию разработки GUI и генетического алгоритма, а также по проектированию структуры классов и методов. В следующем этапе мы приступим к реализации этих компонентов и их интеграции.