Методы оптимизации.

Отчет по лабораторной работе №2

Работа выполнена группой:

Дзюба Мария M3235  
Карасева Екатерина M3235  
Рындина Валерия M3235

Университет ИТМО, 2021

Цель работы: Изучить и реализовать градиентные методы, провести анализ их работы и сравнение.

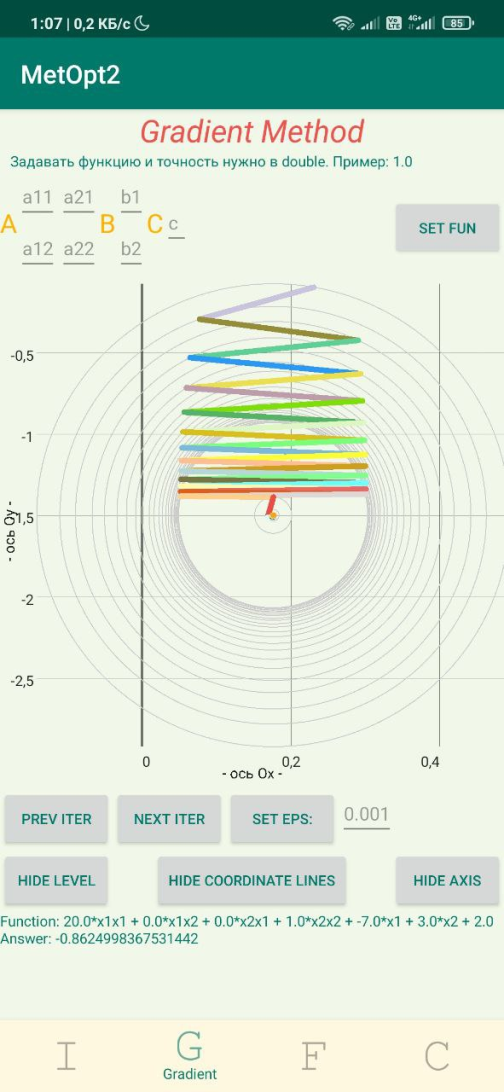
* 1. Постановка задачи:  
     Реализовать алгоритмы:

метод градиентного спуска;

метод наискорейшего спуска;

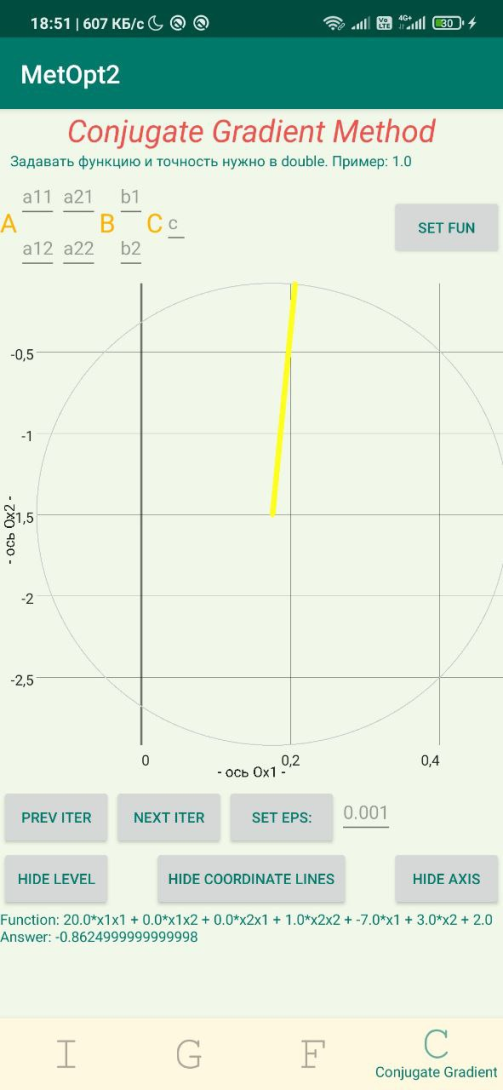
метод сопряженных градиентов.

Оцените, как меняется скорость сходимости, если для поиска величины шага использовать различные методы одномерного поиска.

* 1. Решение задачи:
     + Вычислительная схема всех методов:  
       f(x) дифференцируема в En, xk+1 = xk + αkpk, k ∈ N, где pk определяется с учетом информации о частных производных, величина αk > 0 такова, что: f(xk+1) < f(xk).  
       Остановка итерационного процесса: ║∇f(xk)║< ε
     + Метод градиентного спуска:
       - Вычислительная схема данного метода:  
         Предполагаем, что pk = -∇f(xk), тогда если ∇f(xk) ≠ 0, то (∇f(xk), pk) < 0, и, следовательно, pk – направление убывания f(x), таким образом, найдутся такие ak > 0, что выполнится условие: f(xk+1) < f(xk)
       - Задача минимизации:   
         f(x1, x2) = 20\*(x1)2 + (x2)2 – 7\*x1 + 3\*x2 + 2  
         a = 1.0  
         ε = 0.001
       - Численный результат решения:  
         минимум функции: -0,862481  
         вектор минимума: [0,175976, -1,500425]
       - Итерации поиска решения в виде таблицы  
         приведены в **Приложении 1.**
       - Иллюстрация работы метода:  
         
     + Метод наискорейшего спуска:
       - Вычислительная схема данного метода:  
         pk = -∇f(xk), ak – находится из решения задачи одномерной минимизации:  
         Фk(a) -> min, Фk(a) = f(xk – a\*∇f(xk)), a > 0
       - Задача минимизации:   
         f(x1, x2) = 20\*(x1)2 + (x2)2 – 7\*x1 + 3\*x2 + 2  
         ε = 0.001
       - Численный результат решения:  
         минимум функции: -0,862500  
         вектор минимума: [0,174947, -1,499543]
       - Итерации поиска решения в виде таблицы  
         приведены в **Приложении 2**
       - Иллюстрация работы метода:



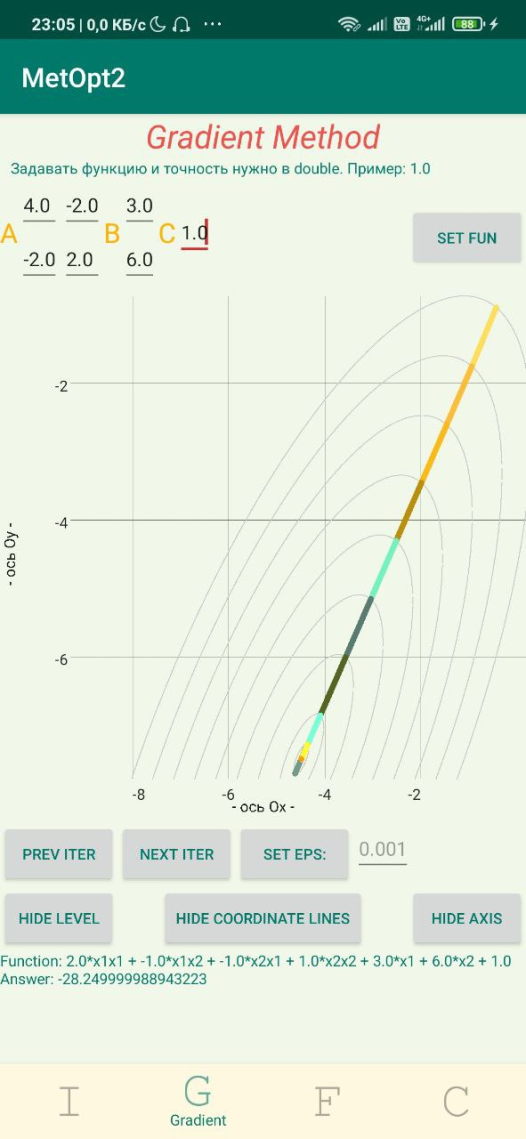
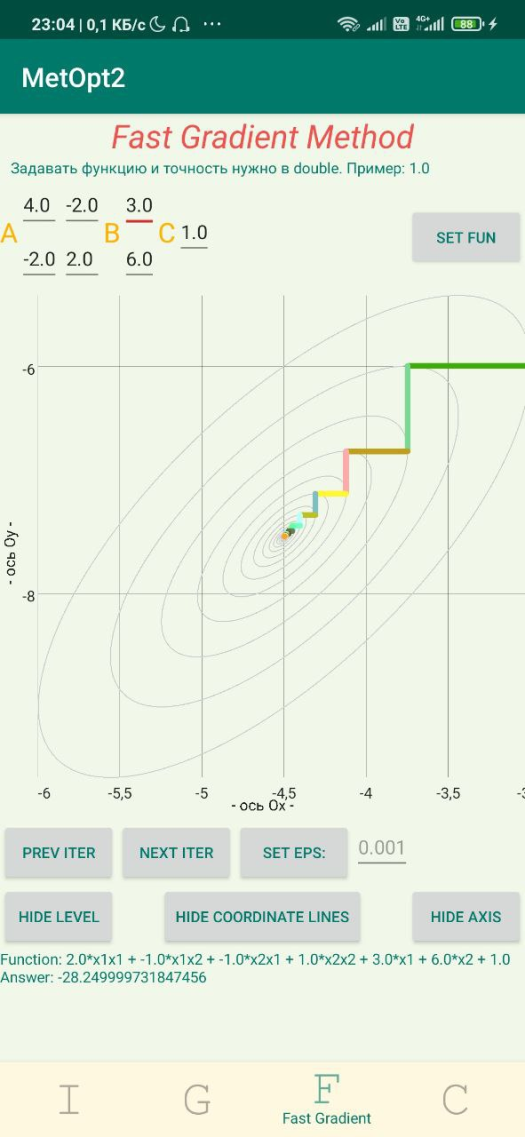
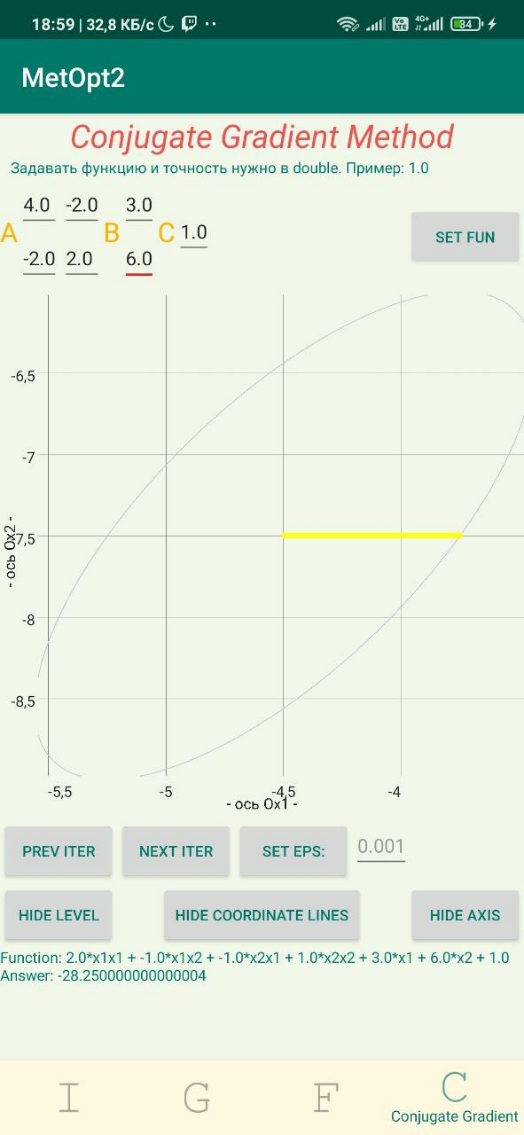
* + - Метод сопряженных градиентов.
      * Вычислительная схема данного метода:  
        p0 = = -∇f(x0), x0 ∈ En  
        для квадратичной функции:  
        ak = ;  
        pk+1 = -∇f(xk+1) + bkpk;  
        bk =
      * Задача минимизации:   
        f(x1, x2) = 20\*(x1)2 + (x2)2 – 7\*x1 + 3\*x2 + 2  
        ε = 0.001
      * Численный результат решения:  
        минимум функции: -0,862500  
        вектор минимума: [0,175000, -1,500000]
      * Итерации поиска решения в виде таблицы  
        приведены в **Приложении 3**
      * Иллюстрация работы метода:



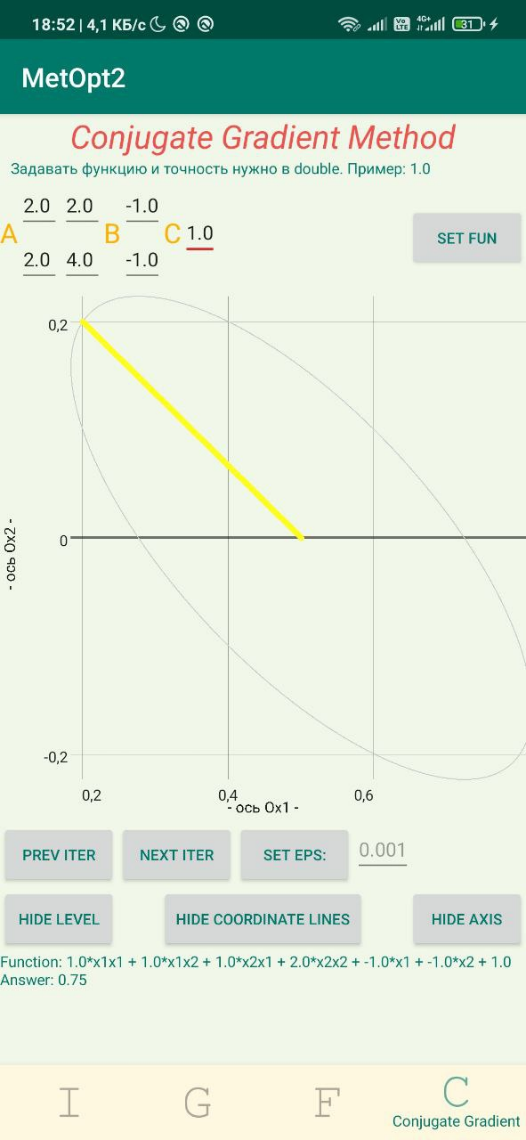
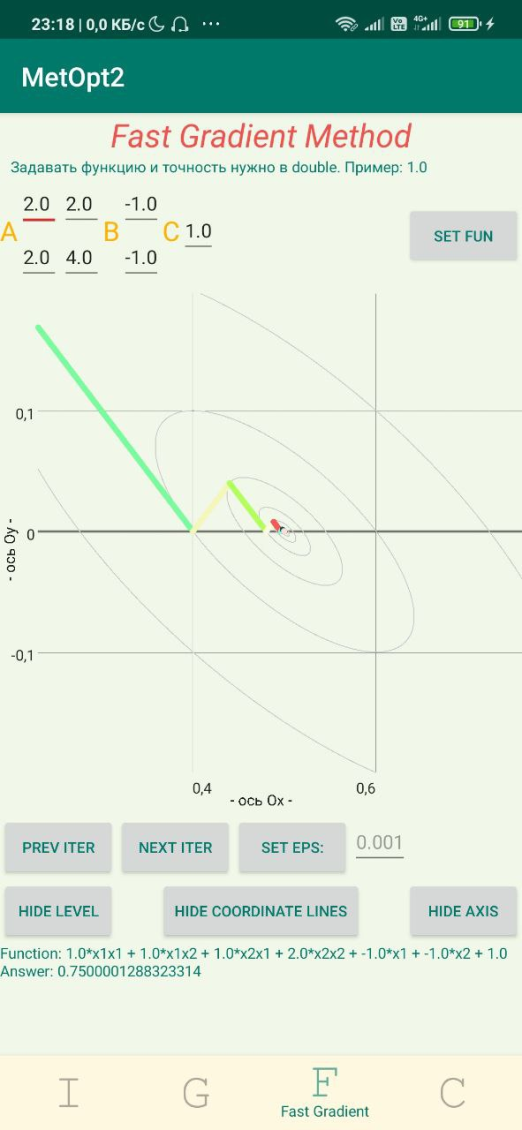
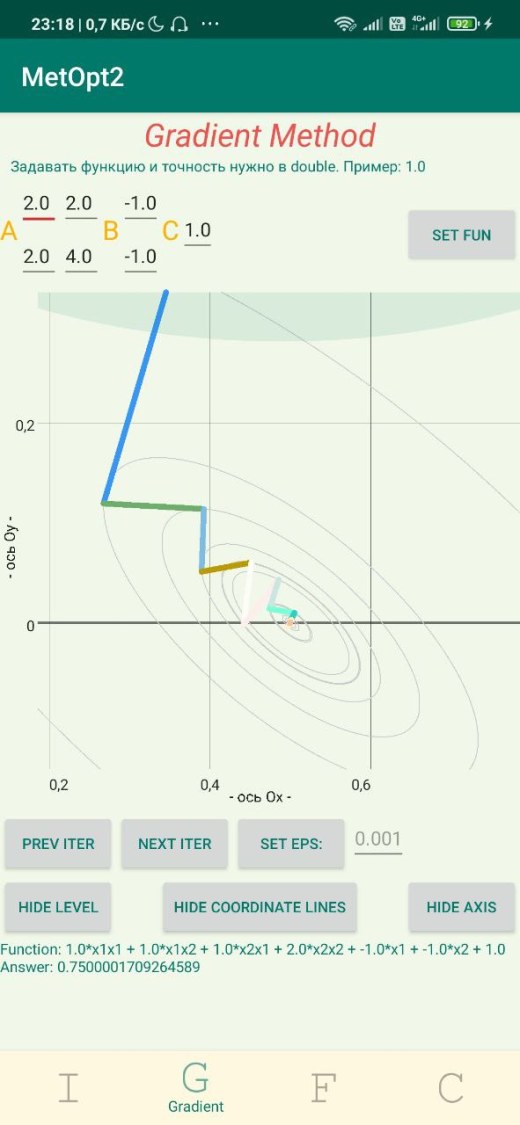
* + - Сравнение времени поиска минимума методом наискорейшего спуска в зависимости от используемого метода одномерной минимизации:  
      f(x1, x2) = 20\*(x1)2 + (x2)2 – 7\*x1 + 3\*x2 + 2

|  |  |  |
| --- | --- | --- |
| Метод | Количество итераций | Время (наносекунды) |
| Дихотомия | 47 | 203.709498 |
| Фибоначчи | 47 | 199.781086 |
| Золотое сечение | 47 | 203.130228 |
| Параболы | 47 | 208.442307 |
| Брент | 47 | 205.629247 |

Вывод: Рассмотрев полученные данные, можно еще раз убедиться в правильности выводов первой лабораторной работы. Чем быстрее сходился метод одномерной оптимизации - тем быстрее сходится метод градиентного спуска, основанный на этой одномерной оптимизации. Но на количество итераций метода это не влияет.

* 1. Постановка задачи:  
     Проанализируйте траектории методов для нескольких квадратичных функций: придумайте две-три квадратичные двумерные функции, на которых работа каждого из методов будет отличаться. Нарисуйте графики с линиями уровня функций и траекториями методов.
  2. Решение задачи:  
     f(x1, x2) = 2\*(x1)2 – 2\*x1\*x2 + (x2)2 + 3\*x1 + 6\*x2 + 1  
     a = 1.0  
     ε = 0.001  
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
       
     Можно заметить, что даже на одной функции все методы работают по разному и имеют кардинально разные траектории.

f(x1, x2) = (x1)2 + 2\*x1\*x2 + 2\*(x2)2 - x1 - x2 + 1  
a = 1.0  
ε = 0.001

  
  
Стандартный градиентный спуск имеет зигзагообразный вид. Очень хорошо видно, что последовательность точек сходится к минимуму линейно.   
Наискорейший спуск выбирает почти оптимальный путь и, что находит минимум он намного быстрее, чем простой градиентный спуск.Метод сопряженных градиентов работает на двумерных функциях очень быстро и точно. (на всех подобранных нами двумерных функциях он работал за 2 итерации)

* 1. Постановка задачи:  
     Исследуйте, как зависит число итераций, необходимое методам для сходимости, от следующих двух параметров:
     + - числа обусловленности 𝑘 ≥ 1 оптимизируемой функции;
       - размерности пространства 𝑛 оптимизируемых переменных.

Для этого для заданных параметров 𝑛 и 𝑘 сгенерируйте случайным образом квадратичную задачу размера 𝑛 с числом обусловленности 𝑘 и запустите на ней методы с некоторой заданной точностью. Замерьте число итераций 𝑇(𝑛, 𝑘), которое потребовалось сделать методу до сходимости.

* 1. Решение задачи

Таблица и график зависимости количества итераций от n и k для метода градиентного спуска:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **n\k** | **5** | **15** | **25** | **35** | **45** | **55** |
| **10** | 3 | 13 | 16 | 35 | 49 | 40 |
| **102** | 7 | 11 | 21 | 35 | 47 | 51 |
| **103** | 18 | 21 | 28 | 41 | 41 | 50 |
| **104** | 54 | 56 | 60 | 61 | 86 | 79 |

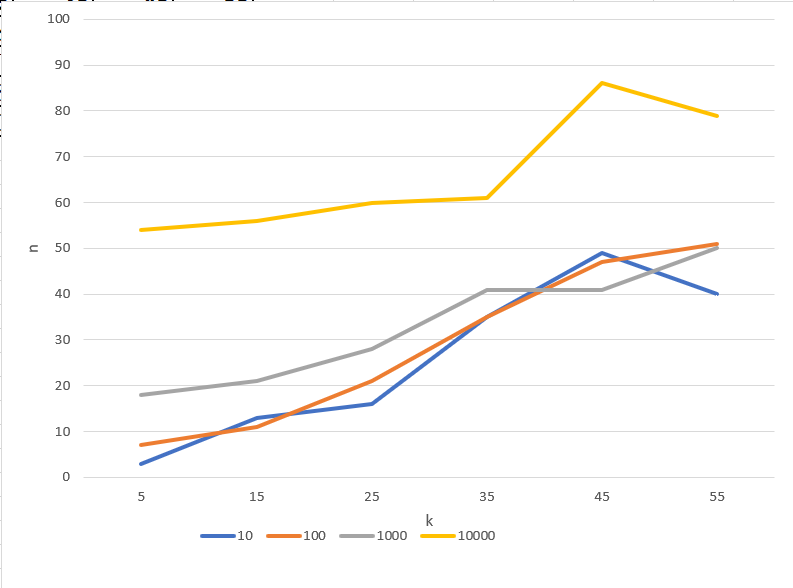


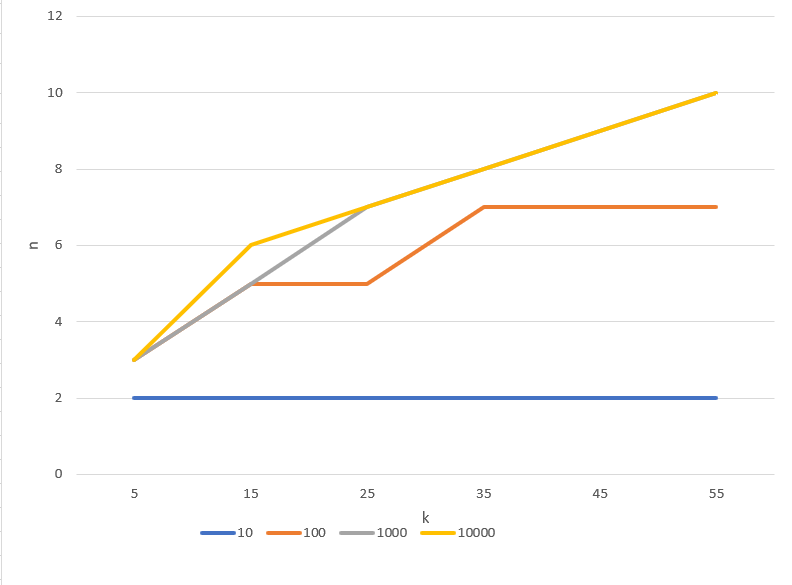
Таблица зависимости количества итераций от n и k для метода наискорейшего спуска:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **n\k** | **5** | **15** | **25** | **35** | **45** | **55** |
| **10** | 3 | 12 | 22 | 31 | 34 | 42 |
| **102** | 4 | 12 | 18 | 25 | 35 | 40 |
| **103** | 4 | 13 | 23 | 30 | 39 | 48 |
| **104** | 5 | 15 | 21 | 29 | 36 | 46 |



Таблица зависимости количества итераций от n и k для метода сопряженных градиентов:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **n\k** | **5** | **15** | **25** | **35** | **45** | **55** |
| **10** | 2 | 2 | 2 | 2 | 2 | 2 |
| **102** | 3 | 5 | 5 | 7 | 7 | 7 |
| **103** | 3 | 5 | 7 | 8 | 9 | 10 |
| **104** | 3 | 6 | 7 | 8 | 9 | 10 |



Вывод: Количества итераций от n зависит пропорционально, и в большинстве случаев пропорционально k, но иногда нет, вероятно, на это влияет стартовая точка поиска.

* 1. Постановка задачи:

1. Для разработанного программного кода в отчете привести код основных модулей, диаграмму классов, сделать текстовое описание.
2. Графический интерфейс должен быть продемонстрирован несколькими показательными иллюстрациями, описаны основные инструменты для работы с интерфейсом.

Инструментарий в графическом интерфейсе:

* возможность отображения/скрытия линий уровня функции,
* масштабирования изображения,
* подписи к координатным линиям (скрыть/показать);
* координатные оси (скрыть/показать);
* кнопки перехода (вперед/назад) по итерациям;
* метода решения (среди 3х заданных),
* задание начального приближения, точности.
  1. Решение задачи

1. Код основных модулей и текстовое описание представлены по ссылке <https://github.com/valrun/MetOpt2> . Диаграмма классов приведена в **Приложение 4**.
2. Код графического интерфейса, а также файл для установки представлены по ссылке <https://github.com/valrun/MetOpt2> . Иллюстрации и описание инструментов для работы с интерфейсом приведены в **Приложение 5**.

Приложение 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Вектор минимума | Значение минимума | № | Вектор минимума | Значение минимума |
| 0 | [0,000000, 0,000000] | 2,000000 | 39 | [0,175964, -1,502835] | -0,862473 |
| 1 | [0,229786, -0,098480] | 1,161789 | 40 | [0,174032, -1,502551] | -0,862475 |
| 2 | [0,075807, -0,295432] | 0,785270 | 41 | [0,175968, -1,502296] | -0,862476 |
| 3 | [0,289500, -0,425183] | 0,554936 | 42 | [0,174029, -1,502066] | -0,862477 |
| 4 | [0,063188, -0,531403] | 0,325720 | 43 | [0,175971, -1,501860] | -0,862478 |
| 5 | [0,292593, -0,630767] | 0,169629 | 44 | [0,174027, -1,501673] | -0,862478 |
| 6 | [0,058097, -0,717435] | 0,023236 | 45 | [0,175973, -1,501506] | -0,862479 |
| 7 | [0,295170, -0,796785] | -0,079173 | 46 | [0,174025, -1,501355] | -0,862479 |
| 8 | [0,055229, -0,866990] | -0,174898 | 47 | [0,175974, -1,501220] | -0,862480 |
| 9 | [0,296932, -0,930862] | -0,241231 | 48 | [0,174025, -1,501098] | -0,862480 |
| 10 | [0,053475, -0,987681] | -0,304661 | 49 | [0,175975, -1,500988] | -0,862480 |
| 11 | [0,298099, -1,039244] | -0,347135 | 50 | [0,174024, -1,500889] | -0,862480 |
| 12 | [0,052366, -1,085233] | -0,389685 | 51 | [0,175975, -1,500800] | -0,862480 |
| 13 | [0,298866, -1,126918] | -0,416455 | 52 | [0,174024, -1,500720] | -0,862480 |
| 14 | [0,051654, -1,164148] | -0,445416 | 53 | [0,175975, -1,500648] | -0,862481 |
| 15 | [0,299368, -1,197872] | -0,461869 | 54 | [0,174023, -1,500583] | -0,862481 |
| 16 | [0,051193, -1,228017] | -0,481959 | 55 | [0,175976, -1,500525] | -0,862481 |
| 17 | [0,299698, -1,255313] | -0,491637 | 56 | [0,174023, -1,500473] | -0,862481 |
| 18 | [0,050893, -1,279724] | -0,505925 | 57 | [0,175976, -1,500425] | -0,862481 |
| 19 | [0,299914, -1,301823] | -0,511156 |  |  |  |
| 20 | [0,050697, -1,321592] | -0,521645 |  |  |  |
| 21 | [0,300055, -1,339487] | -0,523958 |  |  |  |
| 22 | [0,050569, -1,355498] | -0,531956 |  |  |  |
| 23 | [0,300148, -1,369990] | -0,532355 |  |  |  |
| 24 | [0,050485, -1,382958] | -0,538721 |  |  |  |
| 25 | [0,175347, -1,388827] | -0,850138 |  |  |  |
| 26 | [0,167558, -1,513584] | -0,861208 |  |  |  |
| 27 | [0,175338, -1,512874] | -0,862332 |  |  |  |
| 28 | [0,173523, -1,509415] | -0,862368 |  |  |  |
| 29 | [0,175384, -1,508822] | -0,862419 |  |  |  |
| 30 | [0,174102, -1,507348] | -0,862430 |  |  |  |
| 31 | [0,175910, -1,506608] | -0,862440 |  |  |  |
| 32 | [0,174074, -1,505942] | -0,862448 |  |  |  |
| 33 | [0,175934, -1,505345] | -0,862454 |  |  |  |
| 34 | [0,174056, -1,504808] | -0,862459 |  |  |  |
| 35 | [0,175949, -1,504326] | -0,862463 |  |  |  |
| 36 | [0,174044, -1,503891] | -0,862467 |  |  |  |
| 37 | [0,175958, -1,503502] | -0,862469 |  |  |  |
| 38 | [0,174037, -1,503151] | -0,862472 |  |  |  |

Приложение 2

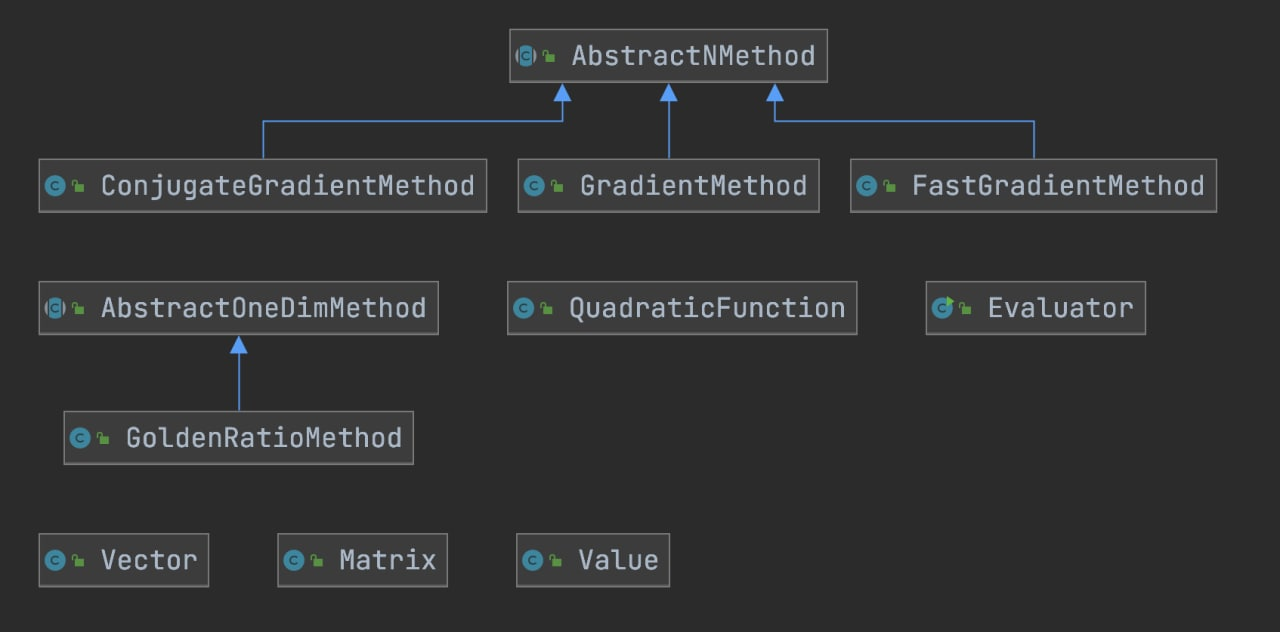
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Вектор минимума | Значение минимума | № | Вектор минимума | Значение минимума |
| 0 | [0,000000, 0,000000] | 2,000000 | 39 | [0,175038, -1,498240] | -0,862497 |
| 1 | [0,205291, -0,087982] | 1,149646 | 40 | [0,174847, -1,498685] | -0,862498 |
| 2 | [0,052069, -0,445102] | 0,552552 | 41 | [0,175027, -1,498762] | -0,862498 |
| 3 | [0,196308, -0,506989] | 0,132651 | 42 | [0,174892, -1,499075] | -0,862499 |
| 4 | [0,088561, -0,758055] | -0,162584 | 43 | [0,175019, -1,499130] | -0,862499 |
| 5 | [0,189983, -0,801582] | -0,370223 | 44 | [0,174924, -1,499350] | -0,862499 |
| 6 | [0,114196, -0,978222] | -0,516306 | 45 | [0,175013, -1,499388] | -0,862500 |
| 7 | [0,185539, -1,008833] | -0,619034 | 46 | [0,174947, -1,499543] | -0,862500 |
| 8 | [0,132243, -1,133022] | -0,691265 |  |  |  |
| 9 | [0,182411, -1,154551] | -0,742067 |  |  |  |
| 10 | [0,144926, -1,241914] | -0,777803 |  |  |  |
| 11 | [0,180213, -1,257055] | -0,802934 |  |  |  |
| 12 | [0,153851, -1,318487] | -0,820607 |  |  |  |
| 13 | [0,178666, -1,329136] | -0,833037 |  |  |  |
| 14 | [0,160125, -1,372344] | -0,841779 |  |  |  |
| 15 | [0,177578, -1,379834] | -0,847927 |  |  |  |
| 16 | [0,164539, -1,410219] | -0,852251 |  |  |  |
| 17 | [0,176813, -1,415486] | -0,855292 |  |  |  |
| 18 | [0,167643, -1,436859] | -0,857431 |  |  |  |
| 19 | [0,176275, -1,440563] | -0,858935 |  |  |  |
| 20 | [0,169826, -1,455592] | -0,859993 |  |  |  |
| 21 | [0,175897, -1,458198] | -0,860736 |  |  |  |
| 22 | [0,171361, -1,468769] | -0,861260 |  |  |  |
| 23 | [0,175631, -1,470601] | -0,861628 |  |  |  |
| 24 | [0,172441, -1,478035] | -0,861887 |  |  |  |
| 25 | [0,175444, -1,479323] | -0,862069 |  |  |  |
| 26 | [0,173200, -1,484553] | -0,862197 |  |  |  |
| 27 | [0,175312, -1,485459] | -0,862287 |  |  |  |
| 28 | [0,173734, -1,489135] | -0,862350 |  |  |  |
| 29 | [0,175219, -1,489773] | -0,862394 |  |  |  |
| 30 | [0,174110, -1,492359] | -0,862426 |  |  |  |
| 31 | [0,175154, -1,492808] | -0,862448 |  |  |  |
| 32 | [0,174374, -1,494626] | -0,862463 |  |  |  |
| 33 | [0,175109, -1,494941] | -0,862474 |  |  |  |
| 34 | [0,174560, -1,496221] | -0,862482 |  |  |  |
| 35 | [0,175076, -1,496443] | -0,862487 |  |  |  |
| 36 | [0,174690, -1,497342] | -0,862491 |  |  |  |
| 37 | [0,175054, -1,497498] | -0,862494 |  |  |  |
| 38 | [0,174782, -1,498131] | -0,862496 |  |  |  |

Приложение 3

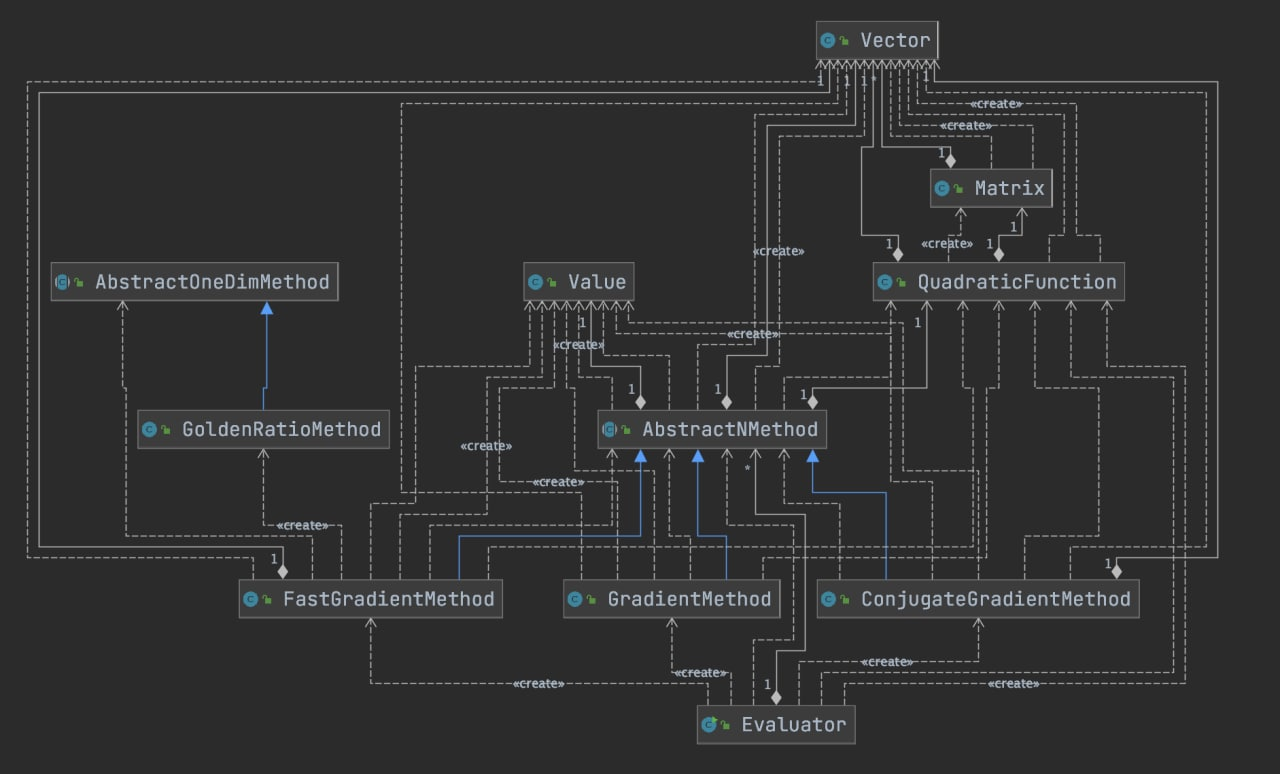
|  |  |  |
| --- | --- | --- |
| № | Вектор минимума | Значение минимума |
| 0 | [0,000000, 0,000000] | 2.000000 |
| 1 | [0,205258, -0,087968] | 1,149646 |
| 2 | [0,175000, -1,500000] | -0,862500 |

Приложение 4

Краткая диаграмма классов:



Развернутая диаграмма классов:



Приложение 5

Иллюстрации графического интерфейса:



Описание инструментов для работы с интерфейсом:

Данное приложение имеет четыре категории меню. Первая информационная, в неё указана краткая информация про приложение и идеи методов. Оставшиеся три для соответствующих методов. У них одинаковый интерфейс:

* Название метода.
* Поля и кнопка для задания функции. Чтобы задать для отрисовки новую функцию, нужно заполнить все поля (a11, a12, a21, a22, b1, b2, c) корректными значениями (то есть числами формата double, например: -3.0) и нажать на кнопку "*SET FUN*"
* График. К сожалению, график не всегда изначально "находится" в том месте, где есть функция, но стоит его немного подвинуть и он сразу перейдет к функции. График очень чувствительный к прикосновениям, он может растягивать оси Ox2, если двигать пальцами вертикально в противоположные стороны или равномерно, если по диагонали (то есть масштабирование), аналогично вдоль Ox1, если горизонтально. Чтобы подвинуть график, нужно провести одним пальцем в противоположном направлении. Серым цветом показаны линии уровня, если нажать на них, то всплывает уведомление, показывающее их значение. А разноцветными линиями показа траектория метода.
* Кнопки "*PREV ITER*", "*NEXT ITER*" соответственно скрывает последнюю или показывает следующую линию траектория метода (кнопки перехода по итерациям)
* Кнопка "*SET EPS*" и поле ввода рядом используются для задания точности. По умолчанию точность 0.001. Для задания точности нужно ввести число формата double.
* Кнопка "*HIDE LEVEL*"/"*SHOW LEVEL*" соответственно скрывает и показывает линии уровня функции
* Кнопка "*HIDE COORDINATE LINES*"/"*SHOW COORDINATE LINES*" соответственно скрывает и показывает подписи к координатным линиям
* Кнопка "*HIDE AXIS*"/"*SHOW AXIS*" соответственно скрывает и показывает координатные оси
* Текущая функция и её минимум.