

# Методы оптимизации.

## Отчет по лабораторной работе №2

Работа выполнена группой:

Дзюба Мария М3235

Карасева Екатерина М3235

Рындина Валерия М3235

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

Задача 1.

а. 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.

б. Решение задачи:

- Вычислительная схема всех методов:

$f(x)$  дифференцируема в  $E_n$ ,

$x^{k+1} = x^k + \alpha_k p^k$ ,  $k \in \mathbb{N}$ , где  $p^k$  определяется с учетом информации о частных производных, величина  $\alpha_k > 0$  такова, что:  $f(x^{k+1}) < f(x^k)$ .

Остановка итерационного процесса:  $\|\nabla f(x^k)\| < \varepsilon$

- Метод градиентного спуска:

- Вычислительная схема данного метода:

Предполагаем, что  $p^k = -\nabla f(x^k)$ , тогда если  $\nabla f(x^k) \neq 0$ , то  $(\nabla f(x^k), p^k) < 0$ , и, следовательно,  $p^k$  – направление убывания  $f(x)$ , таким образом, найдутся такие  $\alpha_k > 0$ , что выполнится условие:  $f(x^{k+1}) < f(x^k)$

- Задача минимизации:

$$f(x_1, x_2) = 20 \cdot (x_1)^2 + (x_2)^2 - 7 \cdot x_1 + 3 \cdot x_2 + 2$$

$$\alpha = 1.0$$

$$\varepsilon = 0.001$$

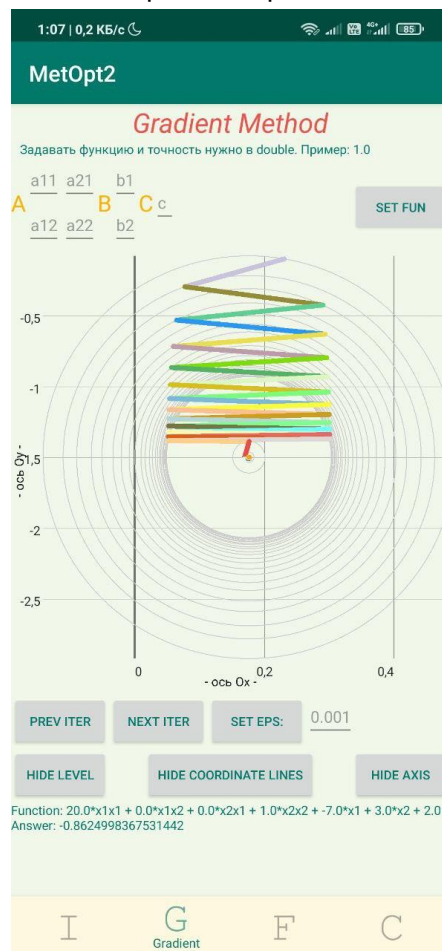
- Численный результат решения:

минимум функции: -0,862481

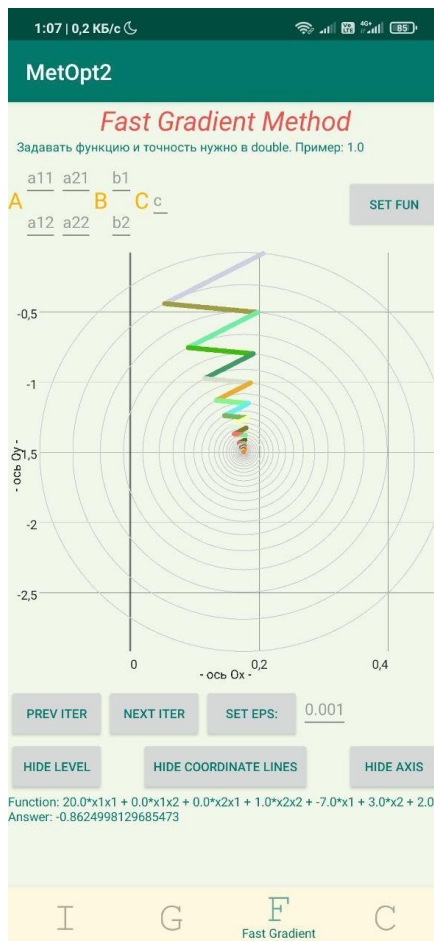
вектор минимума: [0,175976, -1,500425]

- Итерации поиска решения в виде таблицы приведены в **Приложении 1**.

- Иллюстрация работы метода:



- Метод наискорейшего спуска:
  - Вычислительная схема данного метода:  
 $p^k = -\nabla f(x^k)$ ,  $\alpha_k$  – находится из решения задачи одномерной минимизации:  
 $\Phi_k(\alpha) \rightarrow \min, \Phi_k(\alpha) = f(x^k - \alpha \cdot \nabla f(x^k))$ ,  $\alpha > 0$
  - Задача минимизации:  
 $f(x_1, x_2) = 20 \cdot (x_1)^2 + (x_2)^2 - 7 \cdot x_1 + 3 \cdot x_2 + 2$   
 $\varepsilon = 0.001$
  - 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:

$$p^0 = -\nabla f(x^0), x^0 \in E_n$$

для квадратичной функции:

$$\alpha_k = -\frac{(\nabla f(x_k), p_k)}{(Ap_k, p_k)};$$

$$p^{k+1} = -\nabla f(x^{k+1}) + b_k p^k;$$

$$b_k = -\frac{(A\nabla f(x_{k+1}), p_k)}{(Ap_k, p_k)}$$

- Задача минимизации:

$$f(x_1, x_2) = 20 \cdot (x_1)^2 + (x_2)^2 - 7 \cdot x_1 + 3 \cdot x_2 + 2$$

$$\varepsilon = 0.001$$

- Численный результат решения:

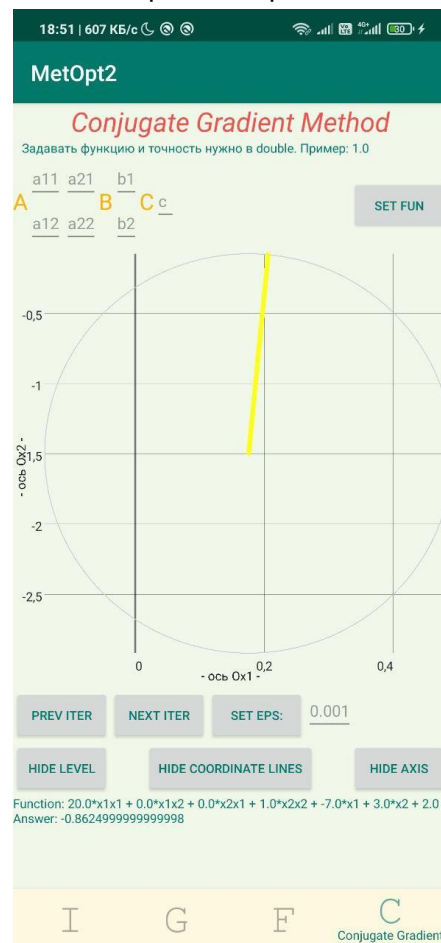
минимум функции: -0,862500

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:

$$f(x_1, x_2) = 20 \cdot (x_1)^2 + (x_2)^2 - 7 \cdot x_1 + 3 \cdot x_2 + 2$$

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

## Задача 2.

### а. Постановка задачи:

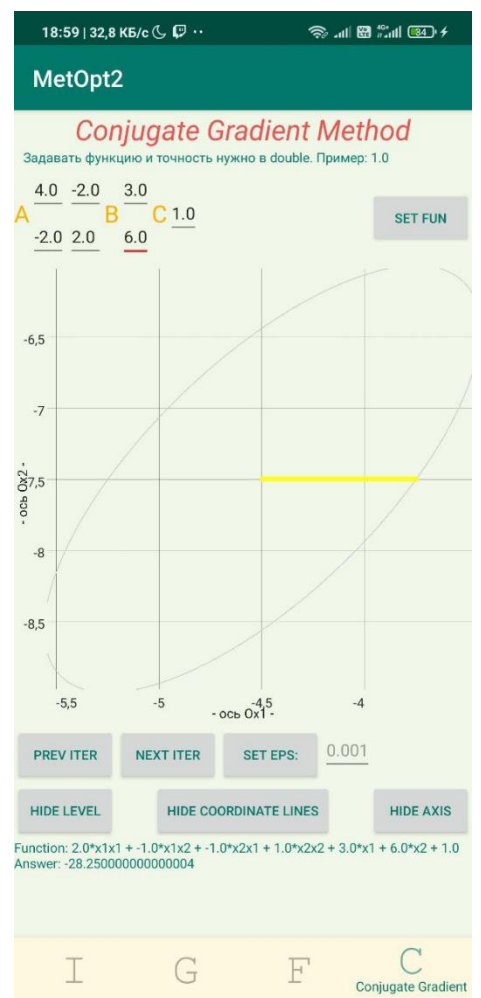
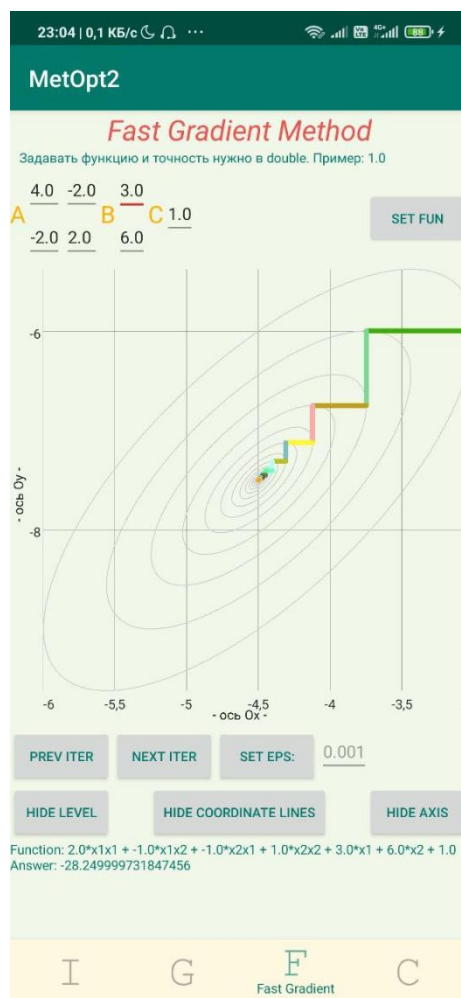
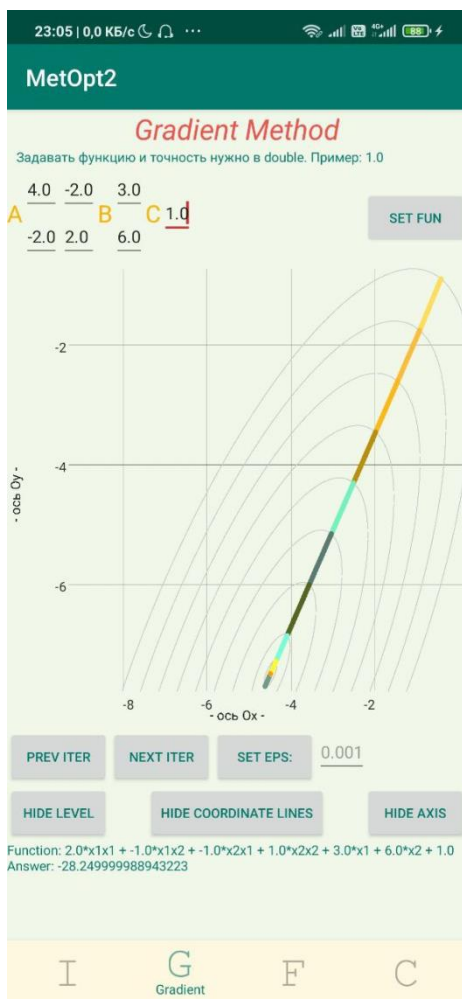
Проанализируйте траектории методов для нескольких квадратичных функций: придумайте две-три квадратичные двумерные функции, на которых работа каждого из методов будет отличаться. Нарисуйте графики с линиями уровня функций и траекториями методов.

### б. Решение задачи:

$$f(x_1, x_2) = 2 \cdot (x_1)^2 - 2 \cdot x_1 \cdot x_2 + (x_2)^2 + 3 \cdot x_1 + 6 \cdot x_2 + 1$$

$$\alpha = 1.0$$

$$\varepsilon = 0.001$$

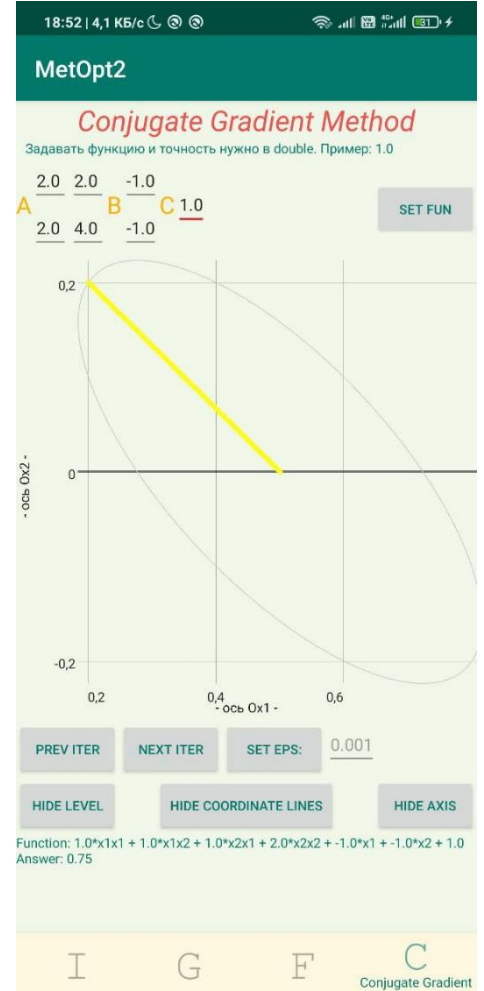
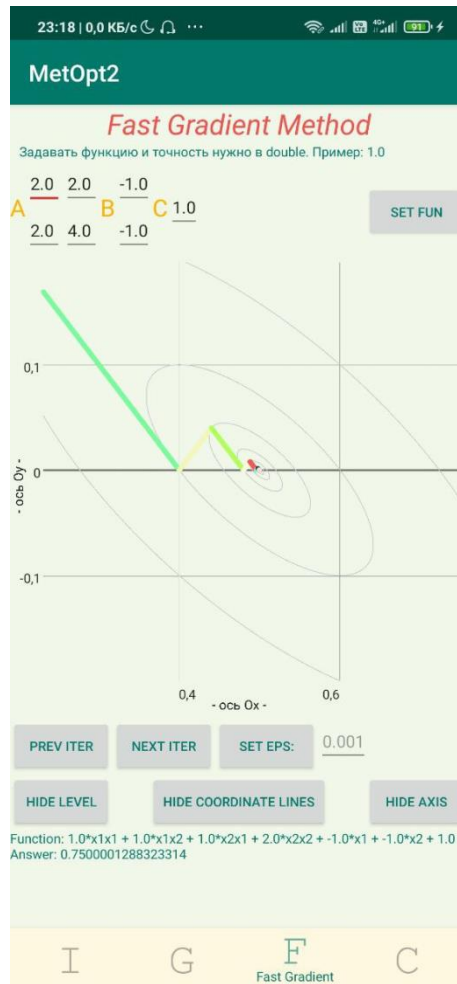
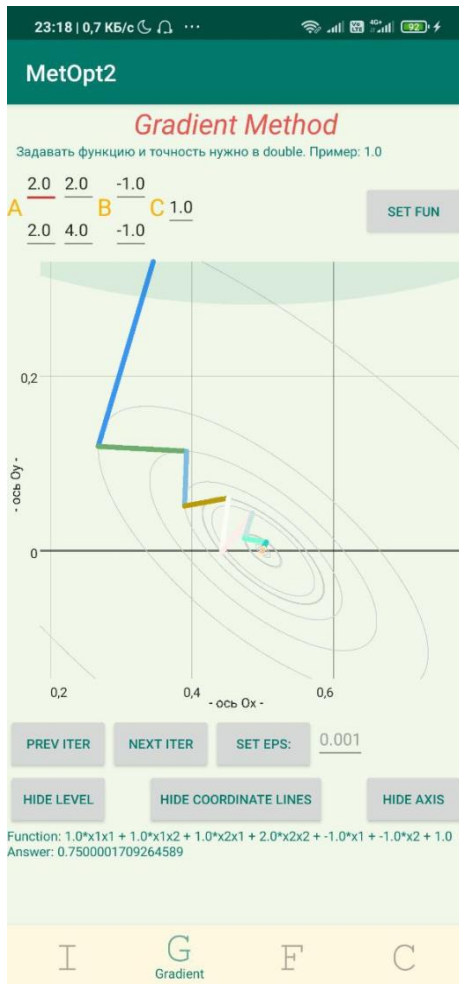


Можно заметить, что даже на одной функции все методы работают по-разному и имеют кардинально разные траектории.

$$f(x_1, x_2) = (x_1)^2 + 2 \cdot x_1 \cdot x_2 + 2 \cdot (x_2)^2 - x_1 - x_2 + 1$$

$$\alpha = 1.0$$

$$\varepsilon = 0.001$$



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 накапливает информацию, делая не очень точные шаги, затем делает точный шаг и сбрасывает память, после чего все повторяется.



### Задача 3.

#### а. Постановка задачи:

Исследуйте, как зависит число итераций, необходимое методам для сходимости, от следующих двух параметров:

- числа обусловленности  $k \geq 1$  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.

#### б. Решение задачи

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

$n \backslash k$	10	100	300	700	1000	1200
10	31	550	2241	5376	6998	7796
$10^2$	54	551	2401	5153	7551	8661
$10^3$	134	657	2477	5781	7761	8544
$10^4$	398	680	2568	5379	8042	8897

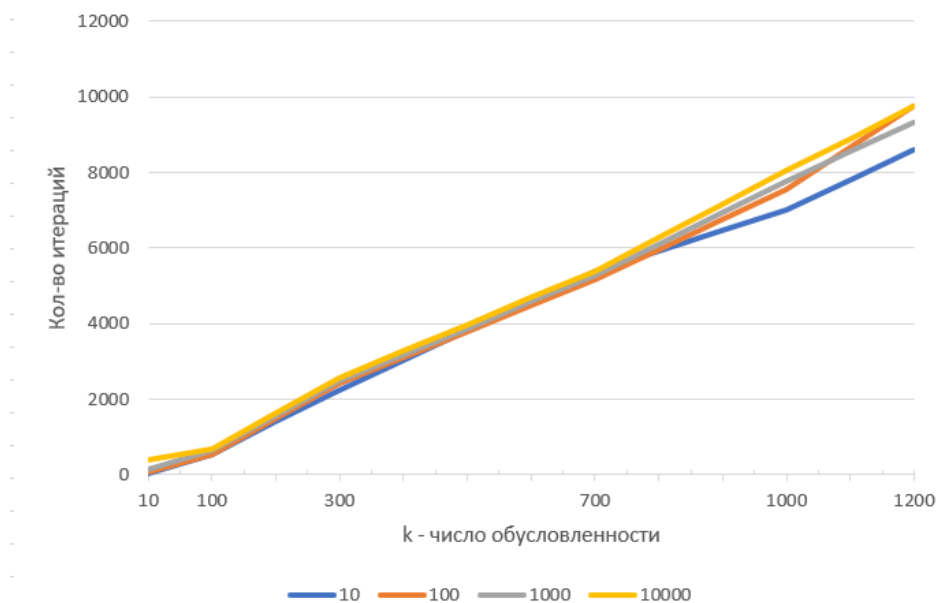


Таблица зависимости количества итераций от  $n$  и  $k$  для метода наискорейшего спуска:

$n \backslash k$	10	100	300	700	1000	1200
10	44	377	1212	2882	3851	4971
$10^2$	43	425	1239	2701	4021	4924
$10^3$	48	426	1280	2597	3769	4767
$10^4$	53	467	1297	2668	4134	5016

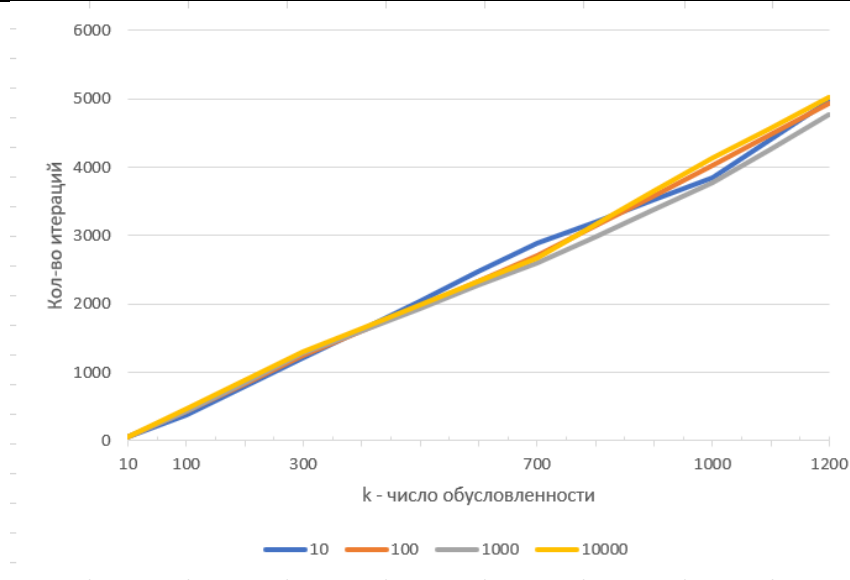
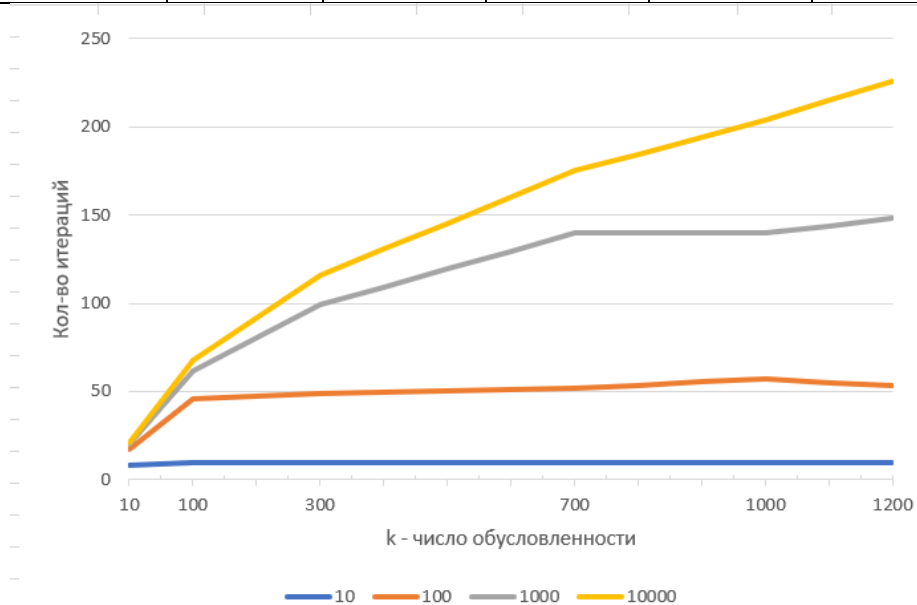


Таблица зависимости количества итераций от  $n$  и  $k$  для метода сопряженных градиентов:

$n \backslash k$	10	100	300	700	1000	1200
10	8	10	10	10	10	10
$10^2$	17	46	49	52	57	53
$10^3$	20	62	99	140	140	148
$10^4$	21	68	116	175	204	226



Вывод: По графикам видно, что в методах градиентного спуска и наискорейшего спуска количества итераций прямо пропорционально зависит от  $k$  – числа обусловленности и в большинстве случаев пропорционально от  $n$ . Скорость возрастания количества итераций в методе градиентного спуска примерно в 2 раза выше, чем в методе наискорейшего спуска. Также можно заметить, что метод сопряженных градиентов совершает не более  $n$  итераций для квадратичных функций при любом  $k$ . В этом методе количество итераций плавно возрастает при возрастании  $n$  и  $k$  (до определенного момента, далее при возрастании  $n$  количество итераций не меняется).

#### Задача 4.

##### а. Постановка задачи:

- I. Для разработанного программного кода в отчете привести код основных модулей, диаграмму классов, сделать текстовое описание.
- II. 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.

##### б. Решение задачи

- I. Код основных модулей и текстовое описание представлены по ссылке <https://github.com/valrun/MetOpt2> . Диаграмма классов приведена в **Приложение 4**.
- II. 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

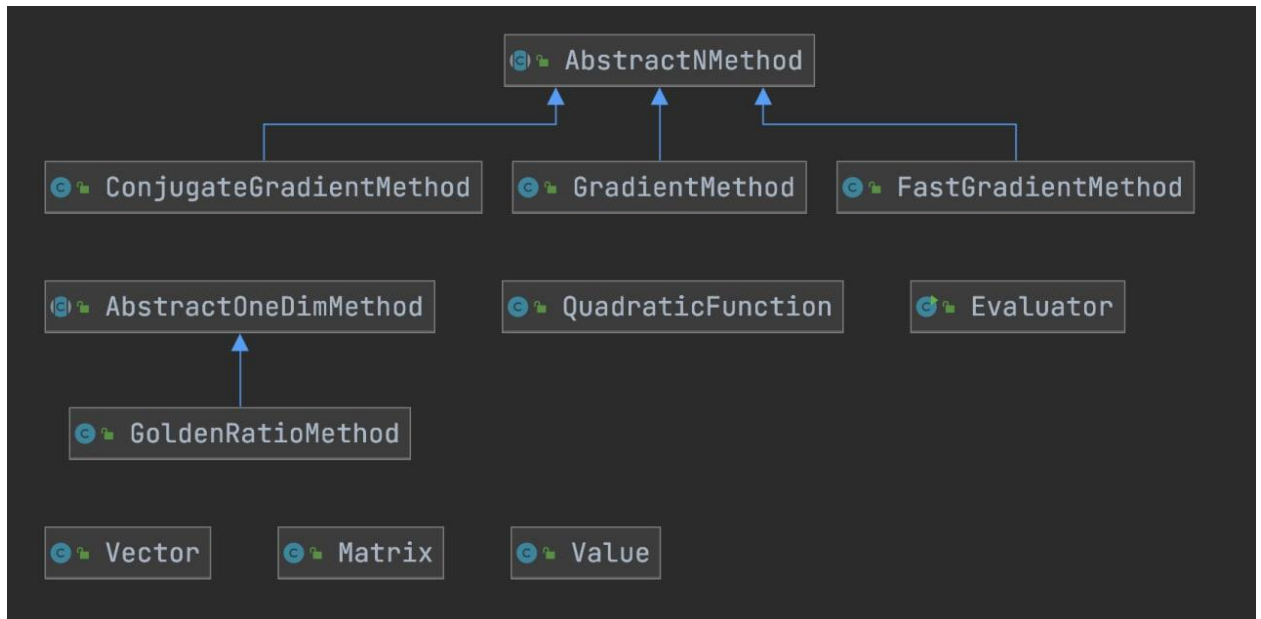
№	Вектор минимума	Значение минимума	№	Вектор минимума	Значение минимума
0	[0,000000, 0,000000]	2,000000	39	[0,175038, -1,498240]	-0,862497
1	[0,205291, -0,087982]	1,149646	40	[0,174847, -1,498685]	-0,862498
2	[0,052069, -0,445102]	0,552552	41	[0,175027, -1,498762]	-0,862498
3	[0,196308, -0,506989]	0,132651	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

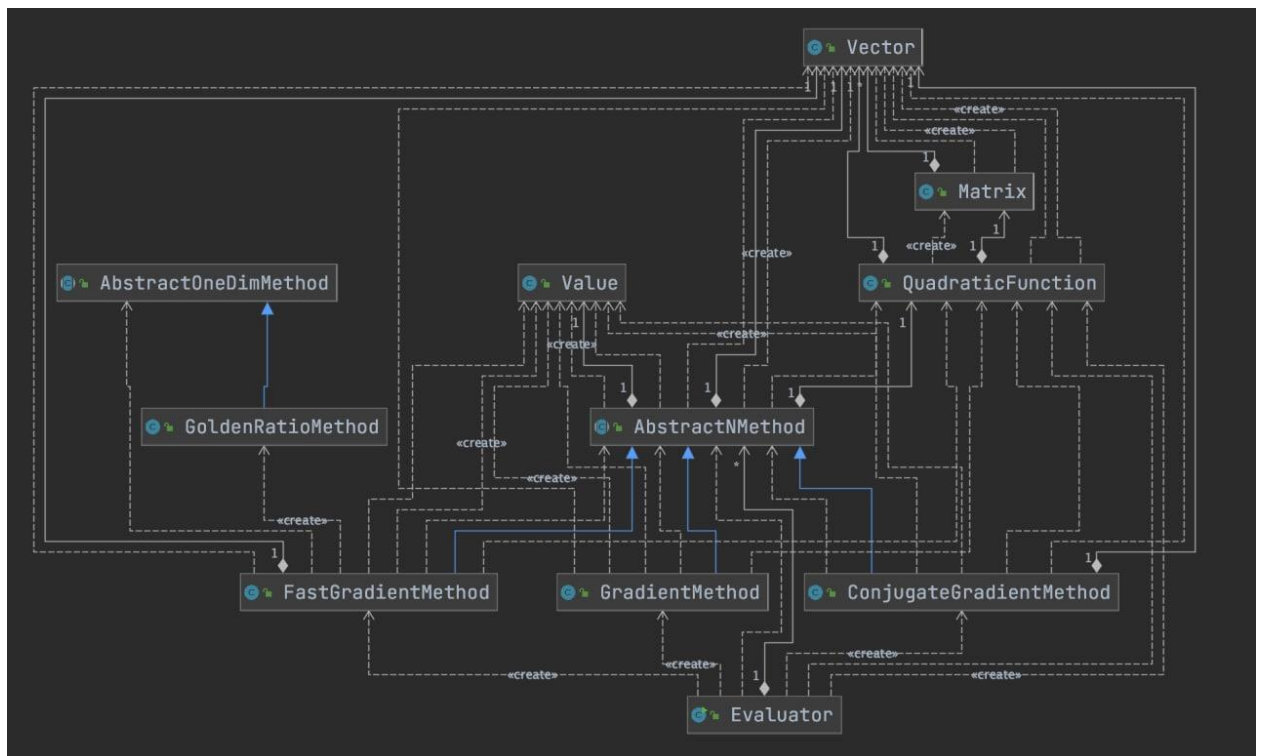
№	Вектор минимума	Значение минимума
0	[0,000000, 0,000000]	2.000000
1	[0,205258, -0,087968]	1,149646
2	[0,175000, -1,500000]	-0,862500

## Приложение 4

Краткая диаграмма классов:



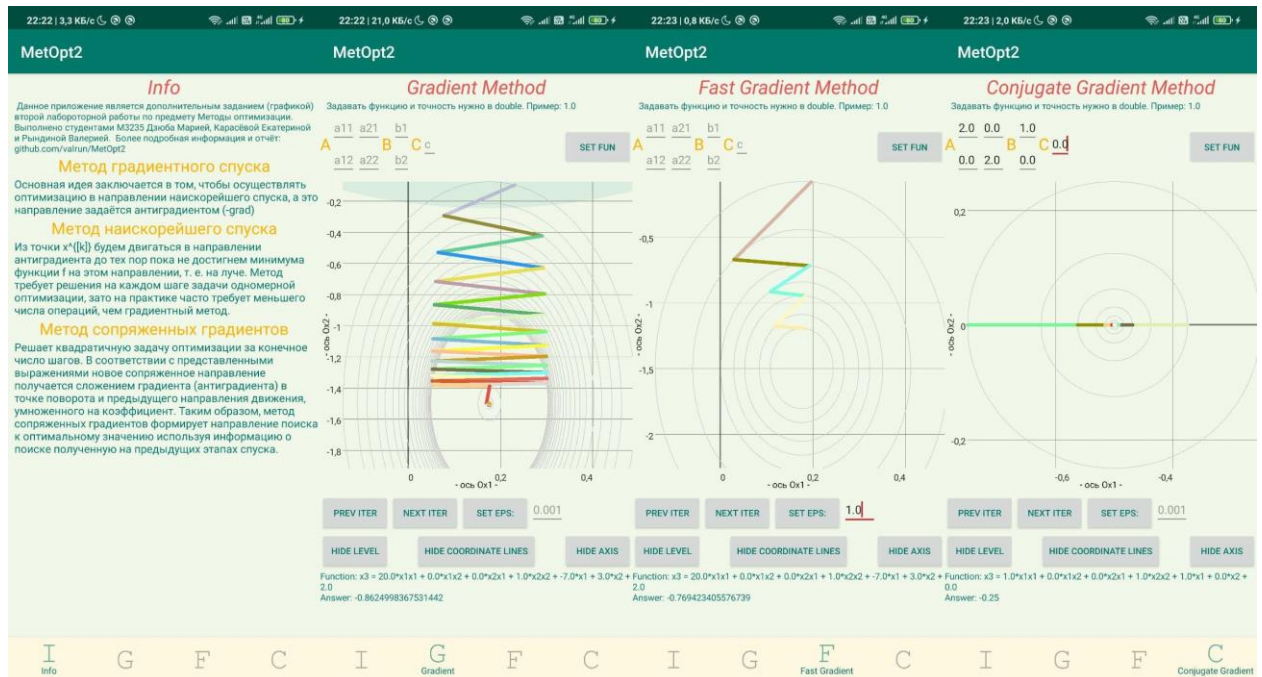
Развернутая диаграмма классов:





## 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 Oх2, если двигать пальцами вертикально в противоположные стороны или равномерно, если по диагонали (то есть масштабирование), аналогично вдоль Oх1, 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 показана 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.