

Расчетная работа №2. Реализация алгоритма поисковой оптимизации

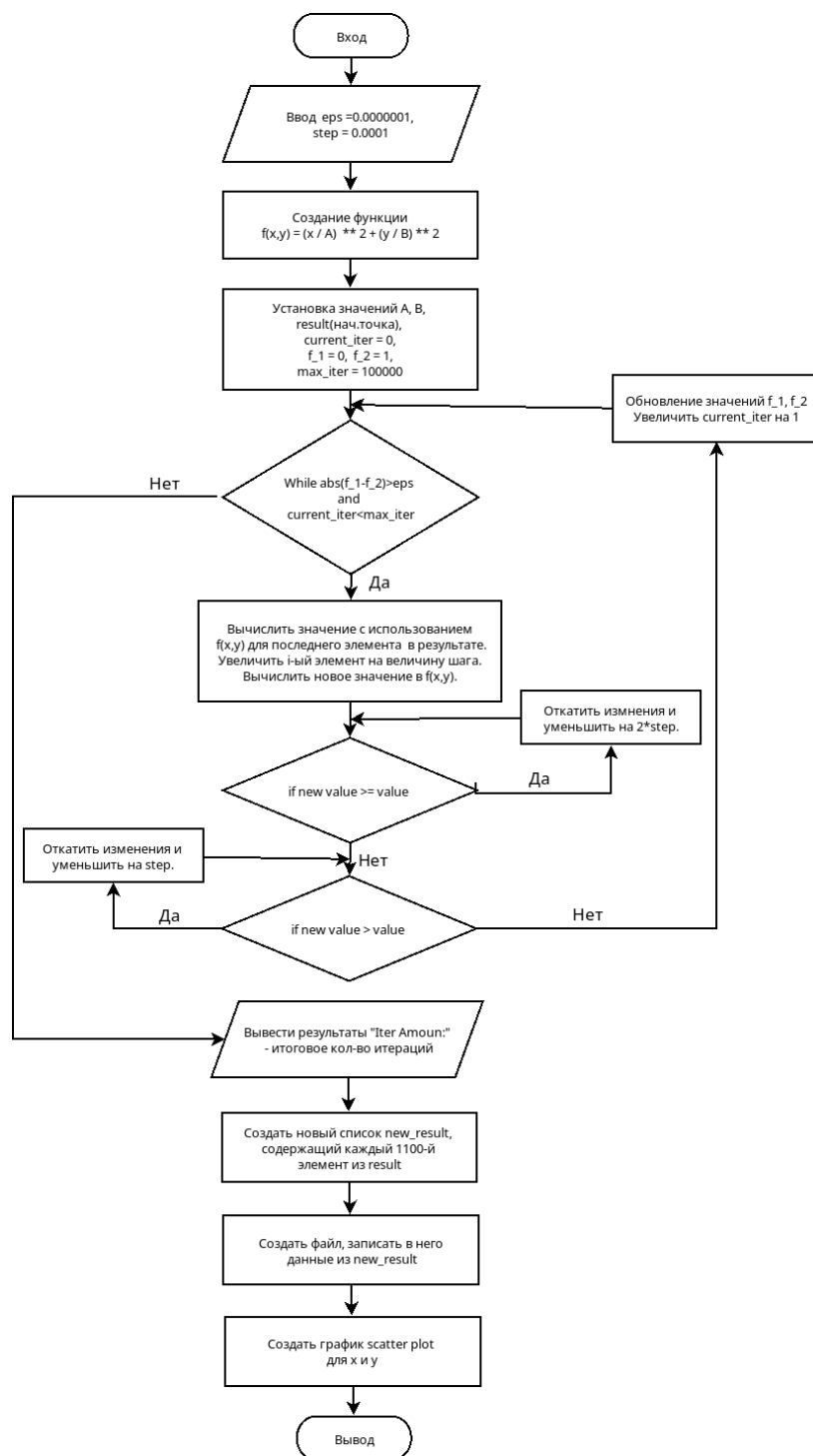
- 1. Вариант 2 (10). Реализация метода покоординатного спуска с постоянным шагом.**
- 2. Описание общего алгоритма поиска.**

Метод покоординатного спуска с постоянным шагом является одним из простейших методов оптимизации. Его основная идея заключается в том, что на каждой итерации мы выбираем одно из направлений (координат) и опускаемся по нему на фиксированное расстояние (шаг).

Реализация данного алгоритма требует учета особенностей конкретной задачи оптимизации (например, выбор критерия остановки, определение начального приближения и шага, обработка ограничений на переменные). Также важно правильно выбрать координатную ось для опускания на каждой итерации, чтобы обеспечить сходимость алгоритма.

В целом, метод покоординатного спуска с постоянным шагом является простым и интуитивно понятным методом оптимизации, однако он часто требует большого количества итераций для сходимости, особенно в случае высокой размерности пространства. Рекомендуется использовать более эффективные методы оптимизации, такие как градиентный спуск или метод Ньютона, если это возможно.

3. Блок-схема общего алгоритма метода покоординатного поиска



4. Описание условий окончания алгоритма поиска.

Условия окончания алгоритма реализации метода покоординатного спуска с постоянным шагом могут быть заданы различными способами, но обычно они включают в себя одно или несколько из следующих условий:

- Достижение определенного числа итераций - алгоритм останавливается после выполнения заданного количества шагов.
- Достижение заданного значения функции потерь или точности - алгоритм завершается, когда значение функции потерь становится меньше определенного порога или когда изменение значения функции потерь меньше определенного значения.
- Обнаружение сходимости - алгоритм завершает работу, если на очередной итерации значения функции потерь не меняются существенно или изменяются медленно.

При реализации метода покоординатного спуска с постоянным шагом, после каждого обновления координаты проверяются условия остановки и при их выполнении алгоритм завершает работу.

5. Функция №1. Функция эллипсоида.

$$f(x, y) = \left(\frac{x}{A}\right)^2 + \left(\frac{y}{B}\right)^2$$

Условие №1

А: 3.0, В: 2.0;

Начальная точка: (2.0, 1.0);

Точность: 0.01;

Шаг: 0.1;

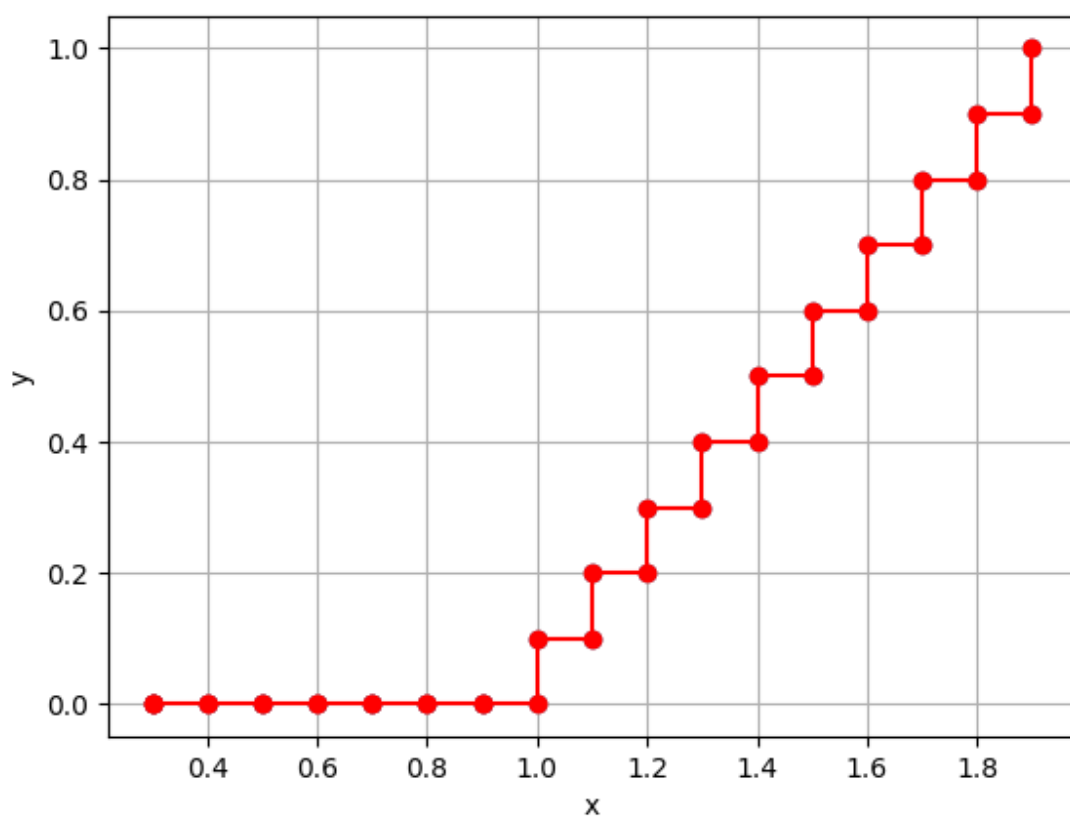
Заданное кол-во итераций: 100;

Таблица 1: Таблица результатов

х	у	f(х,у)
2.0	1.0	0.6511111111111112
1.9000000000000001	0.9000000000000001	0.6036111111111113
1.8	0.9000000000000001	0.5625
1.8	0.8000000000000003	0.5200000000000001
1.7000000000000002	0.8000000000000003	0.4811111111111114
1.7000000000000002	0.7000000000000002	0.4436111111111113
1.6000000000000003	0.7000000000000002	0.4069444444444446
1.6000000000000003	0.6000000000000001	0.3744444444444446
1.5000000000000004	0.6000000000000001	0.34000000000000014
1.5000000000000004	0.5	0.3125000000000001
1.4000000000000006	0.5	0.28027777777777796
1.4000000000000006	0.3999999999999997	0.25777777777777794
1.3000000000000007	0.3999999999999997	0.22777777777777797
1.3000000000000007	0.3	0.21027777777777798
1.2000000000000008	0.3	0.18250000000000025
1.2000000000000008	0.2	0.17000000000000026
1.1000000000000001	0.2	0.14444444444444468
1.1000000000000001	0.10000000000000003	0.13694444444444467
1.0000000000000001	0.10000000000000003	0.11361111111111136
1.0000000000000001	2.7755575615628914e-17	0.11111111111111135
0.90000000000000012	2.7755575615628914e-17	0.09000000000000026

Продолжение на след. странице

x	y	f(x,y)
0.90000000000000012	2.7755575615628914e-17	0.09000000000000026
0.80000000000000014	2.7755575615628914e-17	0.07111111111111135
0.80000000000000014	2.7755575615628914e-17	0.07111111111111135
0.70000000000000013	2.7755575615628914e-17	0.05444444444444464
0.70000000000000013	2.7755575615628914e-17	0.05444444444444464
0.60000000000000012	2.7755575615628914e-17	0.04000000000000016
0.60000000000000012	2.7755575615628914e-17	0.04000000000000016
0.50000000000000011	2.7755575615628914e-17	0.027777777777777905
0.50000000000000011	2.7755575615628914e-17	0.027777777777777905
0.40000000000000011	2.7755575615628914e-17	0.017777777777777875
0.40000000000000011	2.7755575615628914e-17	0.017777777777777875
0.30000000000000011	2.7755575615628914e-17	0.010000000000000073
0.30000000000000011	2.7755575615628914e-17	0.010000000000000073
0.30000000000000011	2.7755575615628914e-17	0.010000000000000073



Количество итераций - 17.

Точка минимума - (0.30000000000000011, 2.7755575615628914e-17)

Условие №2

A: 2.0, B: 3.0;

Начальная точка: (1.0, 1.0);

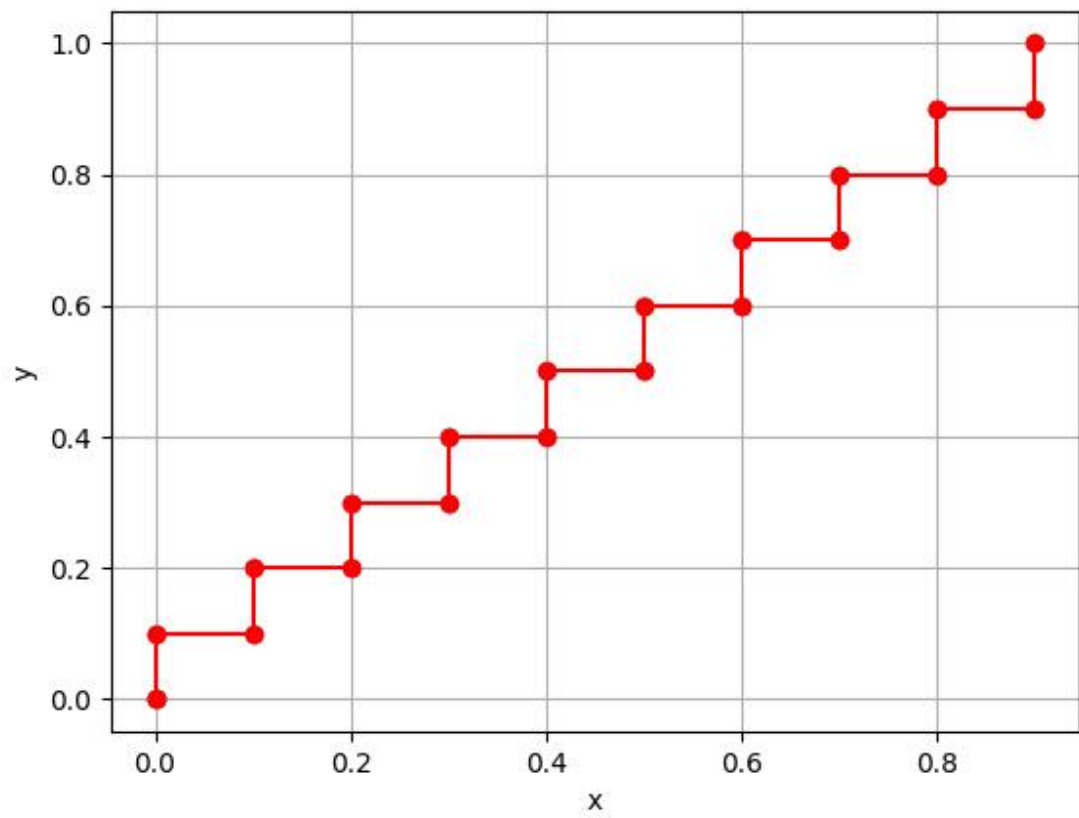
Точность: 0.01;

Шаг: 0.1;

Заданное кол-во итераций: 100.

Таблица 2: Таблица результатов

x	y	f(x,y)
1.0	1.0	0.3136111111111112
0.9000000000000001	0.9000000000000001	0.2925000000000001
0.8000000000000003	0.9000000000000001	0.2500000000000001
0.8000000000000003	0.8000000000000003	0.23111111111111127
0.7000000000000002	0.8000000000000003	0.19361111111111123
0.7000000000000002	0.7000000000000002	0.17694444444444454
0.6000000000000001	0.7000000000000002	0.1444444444444445
0.6000000000000001	0.6000000000000001	0.13000000000000003
0.5	0.6000000000000001	0.10250000000000001
0.5	0.5	0.09027777777777778
0.3999999999999997	0.5	0.06777777777777777
0.3999999999999997	0.3999999999999997	0.05777777777777775
0.3	0.3999999999999997	0.04027777777777777
0.3	0.3	0.0325
0.2	0.3	0.02
0.2	0.2	0.01444444444444447
0.10000000000000003	0.2	0.00694444444444446
0.10000000000000003	0.10000000000000003	0.003611111111111135
2.7755575615628914e-17	0.10000000000000003	0.001111111111111112
2.7755575615628914e-17	2.7755575615628914e-17	2.7818988085593406e-34
2.7755575615628914e-17	2.7755575615628914e-17	2.7818988085593406e-34



Количество итераций - 10.

Точка минимума - (2.7755575615628914e-17, 2.7755575615628914e-17)

Условие №3

A: 3.0, B: 2.0;

Начальная точка: (-1.0, 1.0);

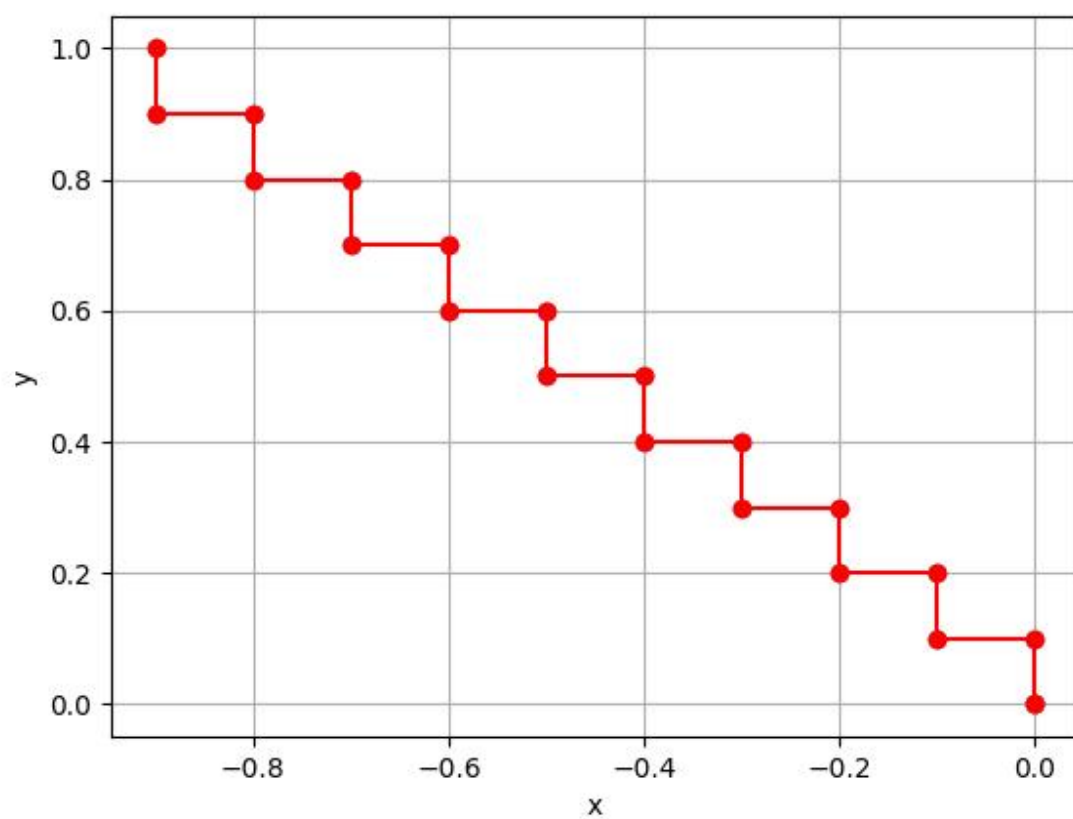
Точность: 0.01;

Шаг: 0.1;

Заданное кол-во итераций: 100;

Таблица 3: Таблица результатов

x	y	f(x,y)
-1.0	1.0	0.33999999999999997
-0.9	0.90000000000000001	0.29250000000000001
-0.8	0.90000000000000001	0.27361111111111112
-0.8	0.80000000000000003	0.23111111111111112
-0.70000000000000001	0.80000000000000003	0.214444444444444458
-0.70000000000000001	0.70000000000000002	0.176944444444444452
-0.60000000000000001	0.70000000000000002	0.16250000000000001
-0.60000000000000001	0.60000000000000001	0.13000000000000003
-0.50000000000000001	0.60000000000000001	0.117777777777777781
-0.50000000000000001	0.5	0.09027777777777779
-0.400000000000000013	0.5	0.08027777777777778
-0.400000000000000013	0.39999999999999997	0.05777777777777779
-0.300000000000000016	0.39999999999999997	0.05
-0.300000000000000016	0.3	0.03250000000000001
-0.200000000000000015	0.3	0.026944444444444445
-0.200000000000000015	0.2	0.014444444444444454
-0.100000000000000014	0.2	0.011111111111111117
-0.100000000000000014	0.10000000000000003	0.003611111111111116
-1.3877787807814457e-16	0.10000000000000003	0.002500000000000002
-1.3877787807814457e-16	2.7755575615628914e-17	2.332515154868986e-33
-1.3877787807814457e-16	2.7755575615628914e-17	2.332515154868986e-33



Количество итераций - 10.

Точка минимума - $(-1.3877787807814457e-16, 2.7755575615628914e-17)$

6. Функция №2. Функция Розенброка.

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2$$

Условие №1

Начальная точка: (2, 1);

Точность: 0.001;

Шаг: 0.01;

Заданное кол-во итераций: 100.

Таблица 4: Таблица результатов

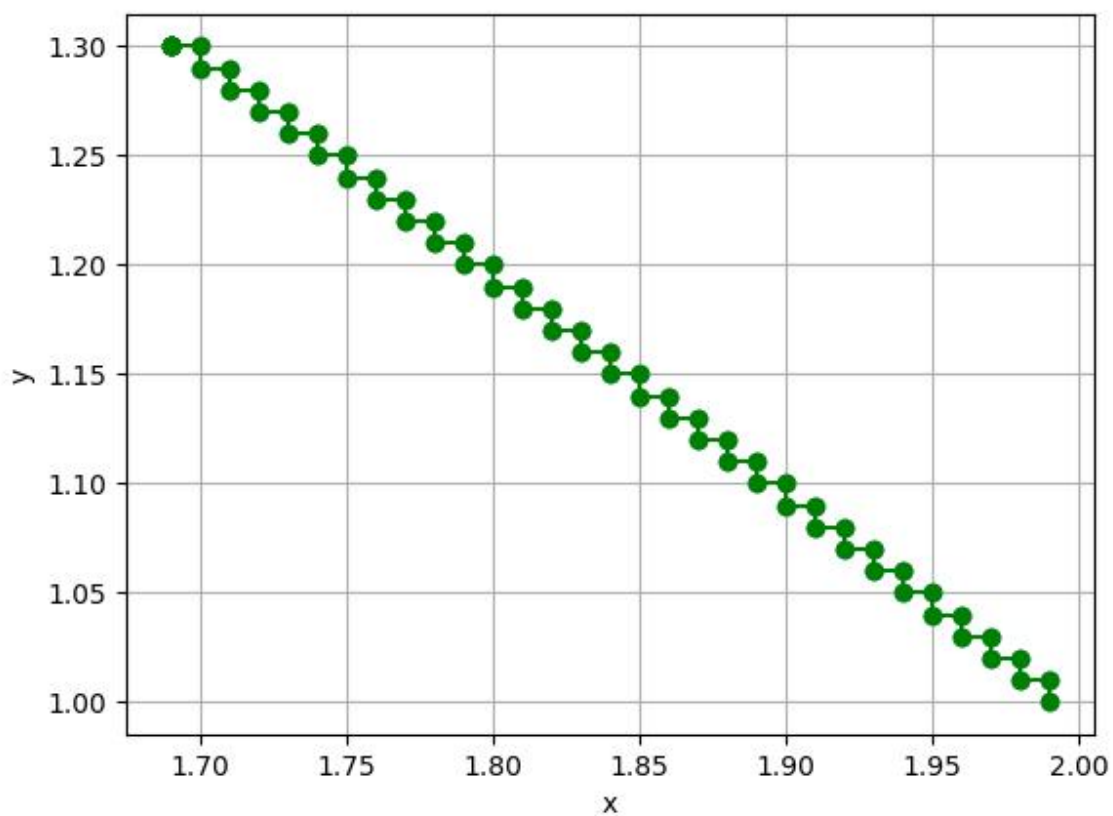
х	у	f(x,y)
1.989999999999998	1.0	98.00999999999995
1.989999999999998	1.01	94.07070099999996
1.979999999999998	1.01	92.14090099999996
1.979999999999998	1.02	88.28521599999995
1.969999999999998	1.02	86.41601599999996
1.969999999999998	1.03	82.64718099999996
1.959999999999997	1.03	80.83898099999996
1.959999999999997	1.04	77.16025599999993
1.949999999999997	1.04	75.41345599999993
1.949999999999997	1.05	71.82812499999994
1.939999999999997	1.05	70.14312499999994
1.939999999999997	1.06	66.65449599999994
1.929999999999997	1.06	65.03169599999993
1.929999999999997	1.07	61.643100999999945
1.919999999999997	1.07	60.08290099999995
1.919999999999997	1.08	56.79769599999994
1.909999999999997	1.08	55.30049599999994
1.909999999999997	1.09	52.12206099999993
1.899999999999997	1.09	50.68826099999993
1.899999999999997	1.10	47.619999999999926
1.889999999999997	1.10	46.24999999999993
1.889999999999997	1.11	43.29534099999993
1.879999999999997	1.11	41.98954099999993

Продолжение на след. странице

x	y	f(x,y)
1.879999999999997	1.12	39.151935999999935
1.869999999999997	1.12	37.910735999999936
1.869999999999997	1.1300000000000001	35.193660999999992
1.859999999999997	1.1300000000000001	34.017460999999992
1.859999999999997	1.1400000000000001	31.424415999999926
1.849999999999996	1.1400000000000001	30.313615999999993
1.849999999999996	1.1500000000000001	27.848124999999994
1.839999999999996	1.1500000000000001	26.803124999999937
1.839999999999996	1.1600000000000001	24.468735999999993
1.829999999999996	1.1600000000000001	23.489935999999933
1.829999999999996	1.1700000000000002	21.290220999999924
1.819999999999996	1.1700000000000002	20.378020999999926
1.819999999999996	1.1800000000000002	18.316575999999994
1.809999999999996	1.1800000000000002	17.471375999999943
1.809999999999996	1.1900000000000002	15.551820999999994
1.799999999999996	1.1900000000000002	14.774020999999943
1.799999999999996	1.2000000000000002	12.999999999999945
1.789999999999996	1.2000000000000002	12.289999999999946
1.789999999999996	1.2100000000000002	10.665180999999947
1.779999999999996	1.2100000000000002	10.023380999999949
1.779999999999996	1.2200000000000002	8.551455999999954
1.769999999999996	1.2200000000000002	7.978255999999955
1.769999999999996	1.2300000000000002	6.6629409999999485
1.759999999999996	1.2300000000000002	6.15874099999995
1.759999999999996	1.2400000000000002	5.003775999999958
1.749999999999996	1.2400000000000002	4.568975999999958
1.749999999999996	1.2500000000000002	3.5781249999999667
1.739999999999995	1.2500000000000002	3.213124999999968
1.739999999999995	1.2600000000000002	2.390175999999969
1.729999999999995	1.2600000000000002	2.095375999999971
1.729999999999995	1.2700000000000002	1.4441409999999737
1.719999999999995	1.2700000000000002	1.2199409999999757
1.719999999999995	1.2800000000000002	0.7442559999999803
1.709999999999995	1.2800000000000002	0.5910559999999826
1.709999999999995	1.2900000000000003	0.2947809999999904

Продолжение на след. странице

x	y	f(x,y)
1.6999999999999995	1.2900000000000003	0.212980999999999248
1.6999999999999995	1.3000000000000003	0.099999999999999797
1.6899999999999995	1.3000000000000003	0.090000000000000016
1.6899999999999995	1.3000000000000003	0.090000000000000016
1.6899999999999995	1.3000000000000003	0.090000000000000016
1.6899999999999995	1.3000000000000003	0.090000000000000016
1.6899999999999995	1.3000000000000003	0.090000000000000016
1.6899999999999995	1.3000000000000003	0.090000000000000016



Количество итераций - 32.

Точка минимума - (1.6899999999999995, 1.3000000000000003).

Условие №2Начальная точка: $(-1, 2)$;

Точность: 0.01;

Шаг: 0.1;

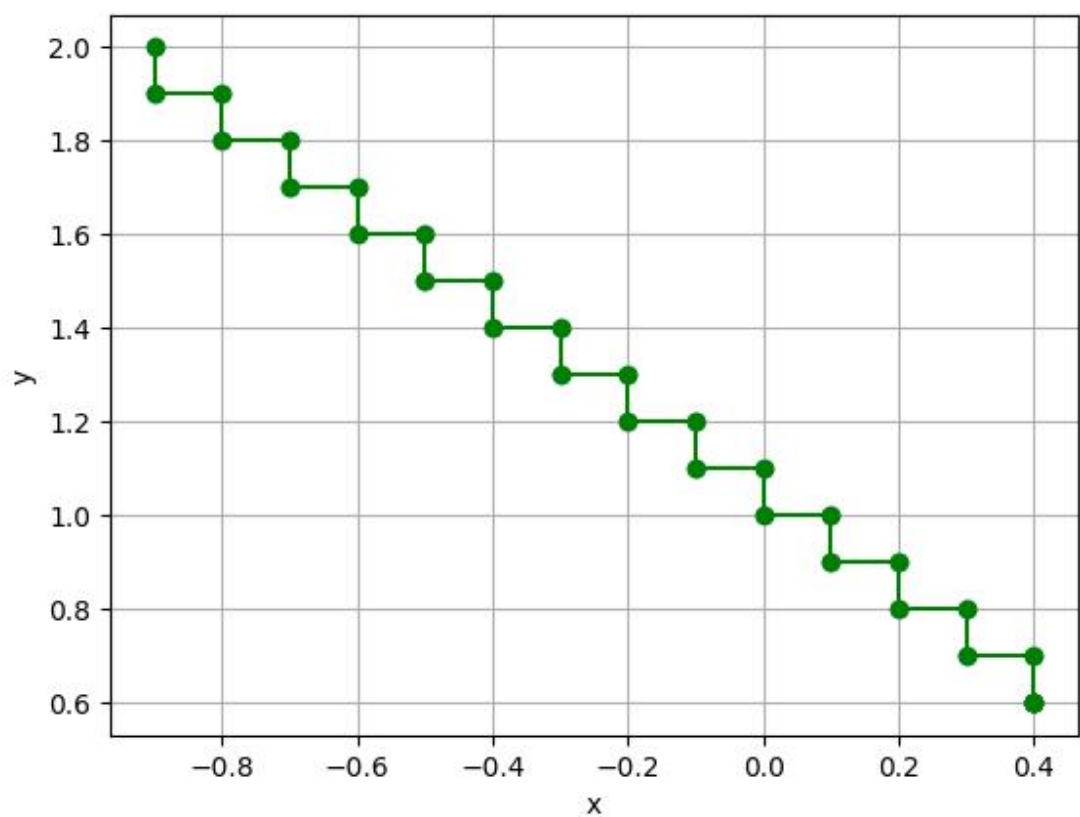
Заданное кол-во итераций: 100.

Таблица 5: Таблица результатов

x	y	f(x,y)
-1.0	2.0	2402.0000000000005
-0.9	1.9000000000000001	2034.8200000000006
-0.8	1.9000000000000001	1945.62
-0.8	1.8	1632.8000000000002
-0.7000000000000001	1.8	1553.0000000000005
-0.7000000000000001	1.7000000000000002	1289.3000000000004
-0.6000000000000001	1.7000000000000002	1218.5000000000005
-0.6000000000000001	1.6000000000000003	998.9200000000008
-0.5000000000000001	1.6000000000000003	936.7200000000006
-0.5000000000000001	1.5000000000000004	756.5000000000007
-0.40000000000000013	1.5000000000000004	702.5000000000006
-0.40000000000000013	1.4000000000000006	557.1200000000007
-0.30000000000000016	1.4000000000000006	510.9200000000007
-0.30000000000000016	1.3000000000000007	396.10000000000076
-0.20000000000000015	1.3000000000000007	357.30000000000075
-0.20000000000000015	1.2000000000000008	269.00000000000074
-0.10000000000000014	1.2000000000000008	237.2000000000006
-0.10000000000000014	1.1000000000000001	171.62000000000057
-1.3877787807814457e-16	1.1000000000000001	146.42000000000056
-1.3877787807814457e-16	1.0000000000000001	100.00000000000048
0.09999999999999987	1.0000000000000001	81.00000000000043
0.09999999999999987	0.90000000000000012	50.42000000000034
0.19999999999999987	0.90000000000000012	37.2200000000003
0.19999999999999987	0.80000000000000014	19.400000000000208
0.2999999999999999	0.80000000000000014	11.60000000000016
0.2999999999999999	0.70000000000000013	3.700000000000735
0.3999999999999999	0.70000000000000013	0.900000000000337

Продолжение на след. странице

x	y	f(x,y)
0.3999999999999999	0.6000000000000012	0.31999999999998685
0.3999999999999999	0.6000000000000012	0.31999999999998685
0.3999999999999999	0.6000000000000011	0.31999999999998785
0.3999999999999999	0.6000000000000011	0.31999999999998785



Количество итераций - 15.

Точка минимума - (0.3999999999999999, 0.6000000000000011).

Условие №3 Начальная точка: (3, -5);

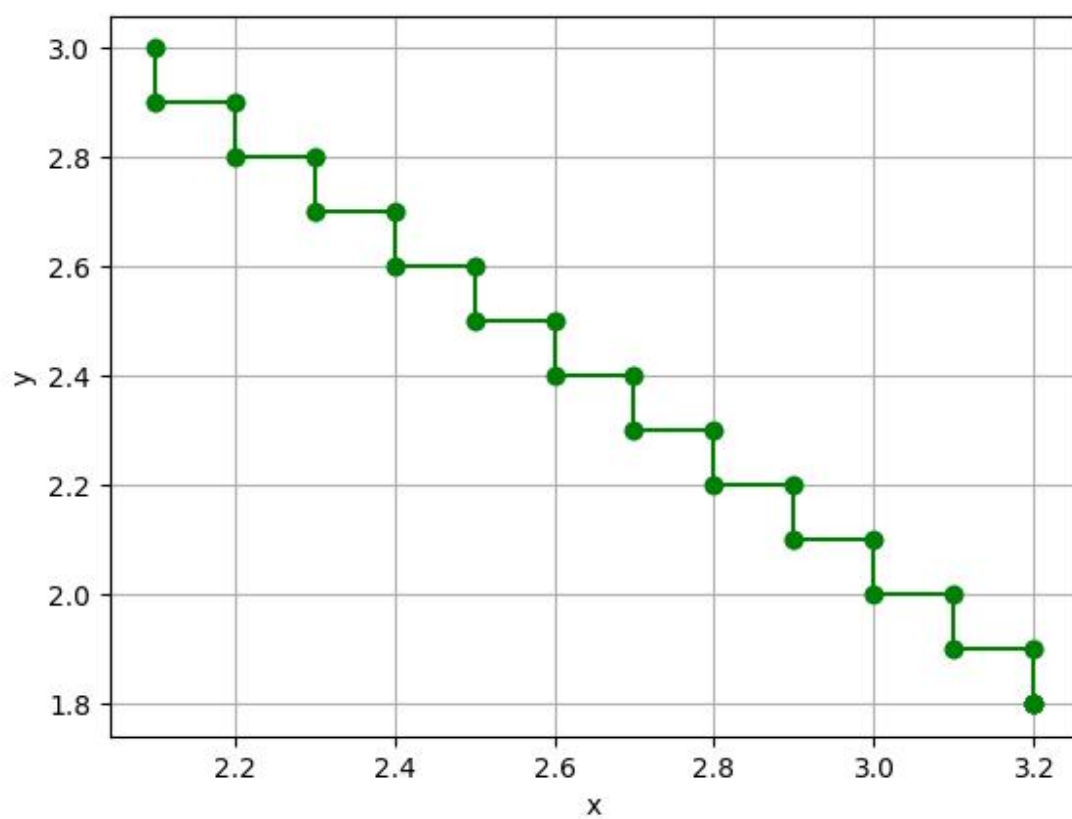
Точность: 0.01;

Шаг: 0.1;

Заданное кол-во итераций: 100.

Таблица 6: Таблица результатов

х	у	f(x,y)
2.0	3.0	4765.000000000001
2.1	2.9	3985.2200000000007
2.2	2.9	3860.02
2.2	2.8	3184.199999999985
2.3000000000000003	2.8	3072.399999999987
2.3000000000000003	2.699999999999997	2492.899999999983
2.4000000000000004	2.699999999999997	2394.09999999998
2.4000000000000004	2.599999999999996	1903.519999999982
2.5000000000000004	2.599999999999996	1817.319999999983
2.5000000000000004	2.499999999999996	1408.499999999984
2.6000000000000005	2.499999999999996	1334.499999999984
2.6000000000000005	2.399999999999995	1000.519999999979
2.7000000000000006	2.399999999999995	938.319999999979
2.7000000000000006	2.299999999999994	672.499999999984
2.8000000000000007	2.299999999999994	621.699999999983
2.8000000000000007	2.199999999999993	417.599999999986
2.9000000000000001	2.199999999999993	377.799999999986
2.9000000000000001	2.099999999999999	229.2199999999872
3.0000000000000001	2.099999999999999	200.019999999988
3.0000000000000001	1.999999999999993	100.9999999999929
3.1000000000000001	1.999999999999993	81.9999999999935
3.1000000000000001	1.899999999999992	26.81999999999617
3.2000000000000001	1.899999999999992	17.6199999999968
3.2000000000000001	1.799999999999994	0.79999999999708
3.2000000000000001	1.799999999999994	0.79999999999708
3.2000000000000001	1.799999999999996	0.79999999999783
3.2000000000000001	1.799999999999996	0.79999999999783



Количество итераций - 13.

Точка минимума - (3.2000000000000001, 1.7999999999999996)

7. Вывод по расчетной работе №2

В настоящем исследовании был рассмотрен метод покоординатного спуска с постоянным шагом. Было проведено тестирование метода на вычислении двух функций: функции эллипсоида с минимальным значением в точке (0,0) и функции Розенброка с минимальным значением в точке (1,1).

Недостатком этого метода является большое количество итераций, необходимых для достижения минимума функций, что делает его достаточно затратным по ресурсам и времени.

8. Код программы

```
1 from matplotlib import pyplot as plt
2 import pandas as pd
3 import csv
4
5 eps = 0.01
6 step = 0.1
7
8 def ellipsoid(x: list):
9     A = 3.0
10    B = 2.0
11    return (float(x[0]) / A) ** 2 + (float(x[1]) / B) ** 2
12
13 result = [[2, 1]]
14 current_iter = 0
15 f_1 = 0
16 f_2 = 1
17 max_iter = 100
18
19 while abs(f_1 - f_2) > eps and current_iter < max_iter:
20     for i in range(len(result[-1])):
21         value = ellipsoid(result[-1])
22         result[-1][i] += step
23         new_value = ellipsoid(result[-1])
24
25         if new_value >= value:
26             result[-1][i] -= 2 * step
27             new_value = ellipsoid(result[-1])
28             if new_value > value:
```

```

29         result[-1][i] += step
30     result.append(result[-1][:])
31     f_1 = f_2
32     f_2 = ellipsoid(result[-1])
33     current_iter += 1
34
35     print("Iter Amount: ", current_iter)
36
37     new_result = result[0::]
38     columns = ['x', 'y', 'z']
39     with open('fun1_1.csv', 'w') as f:
40         write = csv.writer(f)
41         write.writerow(columns)
42         for coord in new_result:
43             coord.append(ellipsoid(coord))
44             write.writerow(coord)
45
46     df = pd.DataFrame(new_result, columns=columns)
47     df.to_csv('fun1_1.csv', index=False)
48
49     x = df['x']
50     y = df['y']
51     plt.xlabel('x')
52     plt.ylabel('y')
53     plt.grid(True)
54
55     plt.scatter(x, y)
56     plt.plot(x, y, '-o', color='r')
57     plt.savefig('fun1_1.jpeg')
58     plt.show()

```

```

1 from matplotlib import pyplot as plt
2 import pandas as pd
3 import csv
4
5 eps = 0.01
6 step = 0.1
7
8 def rosenbrok(x: list):
9     return (1 - x[1]) ** 2 + 100 * (x[0] - x[1] ** 2) ** 2
10

```

```

11 result = [[2, 1]]
12 current_iter = 0
13 f_1 = 0
14 f_2 = 1
15 max_iter = 100
16
17 while abs(f_1 - f_2) > eps and current_iter < max_iter:
18     for i in range(len(result[-1])):
19         value = rosenbrok(result[-1])
20         result[-1][i] += step
21         new_value = rosenbrok(result[-1])
22
23         if new_value >= value:
24             result[-1][i] -= 2 * step
25             new_value = rosenbrok(result[-1])
26             if new_value > value:
27                 result[-1][i] += step
28             result.append(result[-1][:])
29         f_1 = f_2
30         f_2 = rosenbrok(result[-1])
31         current_iter += 1
32
33 print("Iter Amount: ", current_iter)
34
35 new_result = result[0::]
36 columns = ['x', 'y', 'z']
37 with open('fun2_1.csv', 'w') as f:
38     write = csv.writer(f)
39     write.writerow(columns)
40     for coord in new_result:
41         coord.append(rosenbrok(coord))
42         write.writerow(coord)
43
44 df = pd.DataFrame(new_result, columns=columns)
45 df.to_csv('fun2_1.csv', index=False)
46
47 x = df['x']
48 y = df['y']
49 plt.xlabel('x')
50 plt.ylabel('y')
51 plt.grid(True)

```

```
52 |
53 | plt.scatter(x, y)
54 | plt.plot(x, y, '-o', color='g')
55 | plt.savefig('fun2_1.jpeg')
56 | plt.show()
```