|  |  |
| --- | --- |
| ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  Институт математики и информационных систем  Факультет автоматики и вычислительной техники  Кафедра систем автоматизации управления | |
| «Задача одномерной оптимизации»  Отчет по контрольной работе  «Методы оптимизации» | |
|  | Выполнил: |
|  | студент гр. УТб-2301 |
|  | Лесовой Н.С. |
|  | Проверил: |
|  | Шмакова Н.А. |
| Киров  2023 | |

**Задание:** решить задачу по поиску экстремума заданной функции изображенной (1) с использованием метода секущих.

***(1)***

Метод секущих является численным методом оптимизации, который позволяет приближенно находить экстремумы функций. Он основан на итеративном процессе, в котором на каждой итерации вычисляется новое приближение к экстремуму.

На рисунке 1 показаны итерации с приближением к экстремуму и значение экстремума.

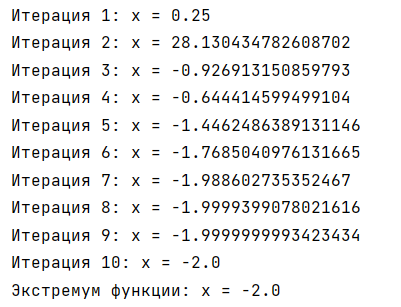


Рисунок 1 – Итерации со значением экстремума

На рисунке 2 и 3 показаны графики функции на разных промежутках.

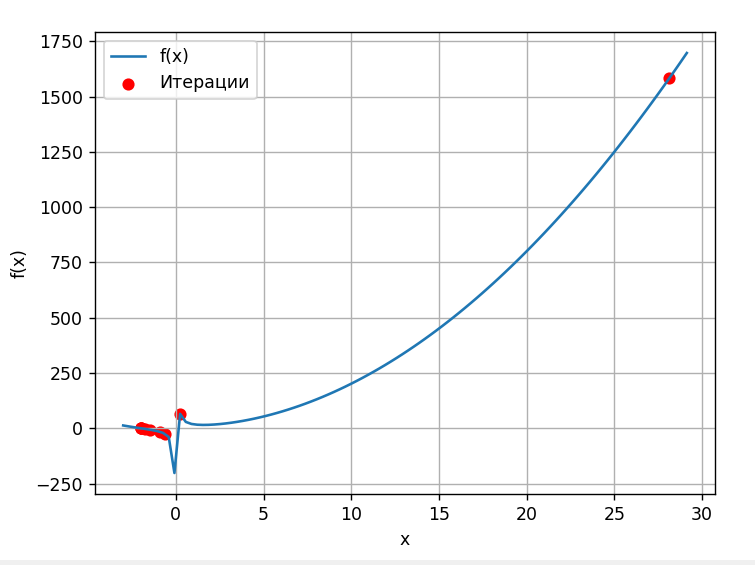


Рисунок 2 – График функции в промежутке от -3 до 30

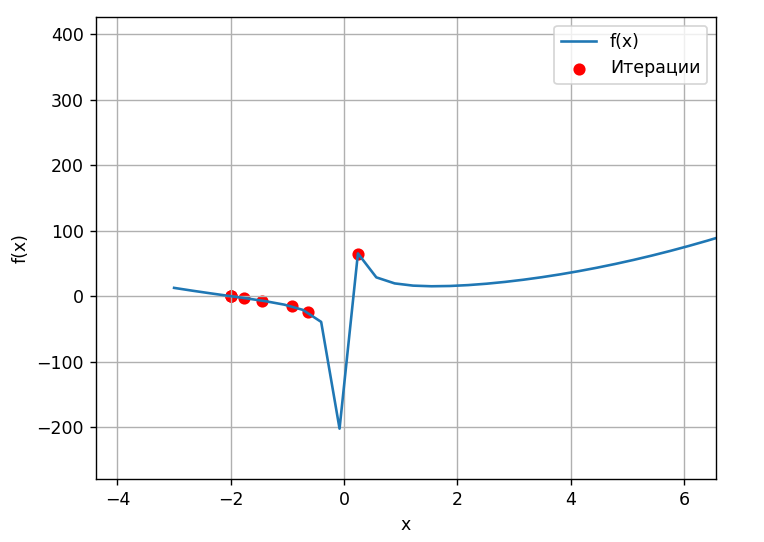


Рисунок 3 – График функции в промежутке от -3 до 6

**Вывод:** в данной работе был реализован метод секущих для поиска экстремума функции на Python. Был добавлен вывод графика функции и маркировка точек итераций. Кроме того, была проведена проверка на оптимальность найденной точки.

# Приложение А

# (обязательное)

**Листинг программного кода**

import matplotlib.pyplot as plt  
import numpy as np  
  
def f(x):  
 return 2\*x\*\*2 + 16/x  
  
  
def secant\_method(f, x0, x1, tol, max\_iter):  
 iter\_count = 0  
 x\_values = []  
 f\_values = []  
  
 while iter\_count < max\_iter:  
 df = (f(x1) - f(x0)) / (x1 - x0)  
  
 x2 = x1 - f(x1) / df  
  
 print(f"Итерация {iter\_count + 1}: x = {x2}")  
  
 x\_values.append(x2)  
 f\_values.append(f(x2))  
 if abs(x2 - x1) < tol:  
 if abs(df) < tol:  
 return x2, f(x2), x\_values, f\_values  
 else:  
 return x2, f(x2), x\_values, f\_values, "Не оптимально"  
  
 x0 = x1  
 x1 = x2  
 iter\_count += 1  
  
 return None, None, x\_values, f\_values, "Не сошелся"  
  
  
x0 = -1  
x1 = 5  
tolerance = 1e-6  
max\_iterations = 100  
  
result, f\_result, x\_values, f\_values, status = secant\_method(f, x0, x1, tolerance, max\_iterations)  
  
  
if result is not None:  
 x\_extremum = result  
 print(f"Экстремум функции: x = {x\_extremum}")  
else:  
 print(f"Метод секущих завершился с сообщением: {status}")  
  
x = np.linspace(min(x\_values) - 1, max(x\_values) + 1, 100)  
plt.plot(x, f(x), label='f(x)')  
plt.scatter(x\_values, f\_values, color='red', marker='o', label='Итерации')  
plt.xlabel('x')  
plt.ylabel('f(x)')  
plt.legend()  
plt.grid()  
plt.show()