

**TOMSK  
POLYTECHNIC  
UNIVERSITY**



**ТОМСКИЙ  
ПОЛИТЕХНИЧЕСКИЙ  
УНИВЕРСИТЕТ**

Министерство науки и высшего образования Российской Федерации  
федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский Томский политехнический университет» (ТПУ)

Инженерная школа природных ресурсов  
Направление подготовки Химическая технология  
Отделение химической инженерии

## **PYTHON ДЛЯ ЗАДАЧ ХИМИЧЕСКОЙ ТЕХНОЛОГИИ**

### **Отчет по лабораторной работе № 4 Минимизация функций**

Выполнил студент гр. 2ДМ24

\_\_\_\_\_  
(Подпись)

Иванцов П.С.

\_\_\_\_\_ 2023 г.

Отчет принят:

Преподаватель  
доцент ОХИ ИШПР, к.т.н.

\_\_\_\_\_  
(Подпись)

В.А. Чузлов

\_\_\_\_\_ 2023 г.

Томск 2023 г.

## Задание 1

Найдите минимум следующих функций, используя методы минимизации, доступные в функции `scipy.optimize.minimize()`.

Начальное приближение:  $x_0 = [0, 0]$ .

1. Функция Экли:

$$f(x, y) = -20 \exp \left[ -0.2 \sqrt{0.5 (x^2 + y^2)} \right] - \exp[0.5 (\cos(2\pi x) + \cos(2\pi y))] + e + 20$$

2. Функция Била:

$$f(x, y) = (1.5 - x + xy)^2 + (2.25 - x + xy^2)^2 + (2.625 - x + xy^3)^2$$

3. Функция Гольдшейна-Прайса:

$$f(x, y) = [1 + (x + y + 1)^2(19 - 14x + 3x^2 - 14y + 6xy + 3y^2)] \cdot [30 + (2x - 3y)^2 \cdot (18 - 32x + 12x^2 + 48y - 36xy + 27y^2)]$$

Функция Матьяса:

$$f(x, y) = 0.26 (x^2 + y^2) - 0.48xy$$

### Программная реализация:

```
from scipy.optimize import minimize
import numpy as np

def ackley_function(xy):
    x, y = xy
    return -20 * np.exp(-0.2 * np.sqrt(0.5 * (x**2 + y**2))) - np.exp(0.5 * (np.cos(2 * np.pi * x) + np.cos(2 * np.pi * y))) + np.e + 20

def beales_function(xy):
    x, y = xy
    return (1.5 - x + x * y)**2 + (2.25 - x + x * y**2)**2 + (2.625 - x + x * y**3)**2

def goldstein_price_function(xy):
    x, y = xy
    return (1 + (x + y + 1)**2 * (19 - 14 * x + 3 * x**2 - 14 * y + 6 * x * y + 3 * y**2)) * (30 + (2 * x - 3 * y)**2 * (18 - 32 * x + 12 * x**2 + 48 * y - 36 * x * y + 27 * y**2))

def matyas_function(xy):
    x, y = xy
    return 0.26 * (x**2 + y**2) - 0.48 * x * y

x0 = [0, 0]

result_ackley = minimize(ackley_function, x0, method='Nelder-Mead')
result_beales = minimize(beales_function, x0, method='Nelder-Mead')
result_goldstein_price = minimize(goldstein_price_function, x0, method='Nelder-Mead')
result_matyas = minimize(matyas_function, x0, method='Nelder-Mead')

print("Ackley Function Minimum:", result_ackley.x)
print("Beale's Function Minimum:", result_beales.x)
print("Goldstein-Price Function Minimum:", result_goldstein_price.x)
print("Matyas Function Minimum:", result_matyas.x)
```

**Ответ:**

Ackley Function Minimum: [0. 0.]

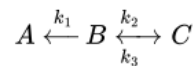
Beale's Function Minimum: [2.99994196 0.49998485]

Goldstein-Price Function Minimum: [-0.59995602 -0.40003256]

Matyas Function Minimum: [0. 0.]

**Задание 2**

Пусть дана схема химических превращений:



Необходимо определить с помощью генетического алгоритма и метода Нелдера-Мида (можно воспользоваться функцией `scipy.optimize.minimize()` , указав соответствующее значение опционального аргумента `method` ) константы скоростей реакций:  $k_1$ ,  $k_2$  и  $k_3$ , если известно, что к моменту времени  $t = 1$ (с) концентрации компонентов равны:  $CA = 0.1423$ ;  $CB = 1.5243$ ;  $CC = 0.5956$  моль/л. Начальные условия:  $CA(0) = 1.0$ ;  $CB = 0.0$ ;  $CC = 0.5$  моль/л. Область поиска для всех констант ограничьте интервалом  $[0; 2]$ .

**Программная реализация:**

```
from scipy.integrate import solve_ivp
from scipy.optimize import minimize
import numpy as np

def chemical_system(t, concentrations, k1, k2, k3):
    CA, CB, CC = concentrations
    dCAdt = -k1 * CA
    dCBdt = 2 * k1 * CA - 2 * k2 * CB**2 + 2 * k3 * CC
    dCCdt = k2 * CB**2 - k3 * CC
    return [dCAdt, dCBdt, dCCdt]

def objective_function(rate_constants):
    k1, k2, k3 = rate_constants
    result = solve_ivp(chemical_system, [0, 1], [1.0, 0.0, 0.5], args=(k1, k2, k3))
    CA, CB, CC = result.y[:, -1]
    error = (CA - 0.1423)**2 + (CB - 1.5243)**2 + (CC - 0.5956)**2
    return error

initial_guess = [1, 1, 1]

opt_result = minimize(objective_function, initial_guess, method='Nelder-Mead')

print("Optimized Rate Constants:", opt_result.x)
plt.show()
```

**Ответ:**

Optimized Rate Constants: [1.95009837 0.33183658 0.83200744]