import numpy as np

import matplotlib.pyplot as plt

# Задаем параметры

L = 1 # Длина стержня

T = 1 # Время

Nx = 10 # Количество узлов по x

Nt = 100 # Количество шагов по времени

alpha = 0.1 # Коэффициент теплопроводности

dx = L / Nx # Шаг по x

dt = T / Nt # Шаг по времени

# Инициализируем массив температуры

u = np.zeros((Nx+1, Nt+1))

# Задаем начальное и граничные условия

u[:, 0] = 1 # Начальное распределение температуры

u[0, :] = 0 # Граничное условие при x=0

u[-1, :] = 0 # Граничное условие при x=L

# Неявная схема

A = np.zeros((Nx-1, Nx-1))

np.fill\_diagonal(A, 1 + 2\*alpha\*dt/dx\*\*2)

np.fill\_diagonal(A[1:], -alpha\*dt/dx\*\*2)

np.fill\_diagonal(A[:, 1:], -alpha\*dt/dx\*\*2)

print(A)

for n in range(Nt):

b = u[1:Nx, n]

u[1:Nx, n+1] = np.linalg.solve(A, b)

print(f"Решенная система \n {u}")

Вывод:

[[ 1.2 -0.1 0. 0. 0. 0. 0. 0. 0. ]

[-0.1 1.2 -0.1 0. 0. 0. 0. 0. 0. ]

[ 0. -0.1 1.2 -0.1 0. 0. 0. 0. 0. ]

[ 0. 0. -0.1 1.2 -0.1 0. 0. 0. 0. ]

[ 0. 0. 0. -0.1 1.2 -0.1 0. 0. 0. ]

[ 0. 0. 0. 0. -0.1 1.2 -0.1 0. 0. ]

[ 0. 0. 0. 0. 0. -0.1 1.2 -0.1 0. ]

[ 0. 0. 0. 0. 0. 0. -0.1 1.2 -0.1]

[ 0. 0. 0. 0. 0. 0. 0. -0.1 1.2]]

Решенная система

[[0. 0. 0. ... 0. 0. 0. ]

[1. 0.91607978 0.84515425 ... 0.15034892 0.1488825 0.14743097]

[1. 0.99295739 0.98105321 ... 0.2858826 0.28310076 0.2803467 ]

...

[1. 0.99295739 0.98105321 ... 0.2858826 0.28310076 0.2803467 ]

[1. 0.91607978 0.84515425 ... 0.15034892 0.1488825 0.14743097]

[0. 0. 0. ... 0. 0. 0. ]]

# Визуализация результатов

x = np.linspace(0, L, Nx+1)

t = np.linspace(0, T, Nt+1)

X, T = np.meshgrid(x, t)

fig = plt.figure()

ax = fig.add\_subplot(111, projection='3d')

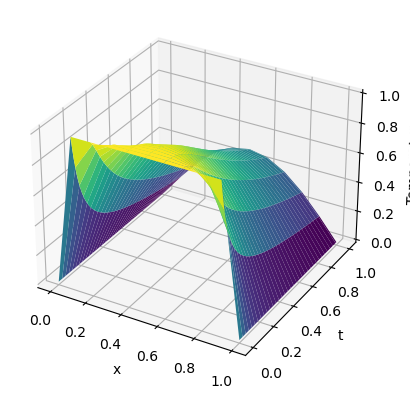
surf = ax.plot\_surface(X, T, u.T, cmap='viridis')

ax.set\_xlabel('x')

ax.set\_ylabel('t')

ax.set\_zlabel('Temperature')

plt.show()



# Визуализация зависимости температуры от времени в точке x=0.5

x\_index = int(0.5 / dx) # Индекс ближайшего узла к x=0.5

plt.figure()

plt.plot(t, u[x\_index, :], label='Temperature at x=0.5')

plt.xlabel('Time')

plt.ylabel('температура')

plt.title('Изменение температуры в x=0.5 за все время')

plt.legend()

plt.grid(True)

plt.show()

