|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***ИУК «Информатика и управление»***

**КАФЕДРА** \_\_***ИУК4 «Программное обеспечение ЭВМ, информационные технологии»***

**ДОМАШНЯЯ РАБОТА №2**

**«Метод разделения переменных для решения ДУЧП2»**

**ДИСЦИПЛИНА: «Моделирование»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-62Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Карельский М.К. )  (Подпись) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Никитенко У.В. )  (Подпись) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2023

**Цель:** овладеть навыками использования специализированных программных пакетов и библиотек для стандартных вычислений и визуализации результатов вычислений; навыками анализа возможностей построения и выделения наиболее важных свойств объектов моделей для моделирования; практических навыков решения ДУЧП2.

**Задачи:** решить методом разделения переменных задачи для ДУЧП2 гиперболического, параболического и эллиптического типов. Выбрать среду для проведения расчетов. Оценить результаты расчетов. Визуализировать результаты.

**Вариант 11**

**Задача 1.**

Решить методом Фурье и методом отражений начально-краевую задачу для волнового уравнения.

* 1. Отдельно выписать задачу Штурма-Лиувилля и решить её.
  2. Ответ представить в максимально компактной форме
  3. Построить профиль струны в различные моменты времени, начиная с нулевого.

**Решение:**

Из условия имеем:

Решим задачу методом Фурье

Последнее равенство возможно только при условии, что оно равно константе, положим .

Либо , что противоречит физическому смыслу задачи, либо и

Задача Штурма-Лиувилля:

Характеристическое уравнение:

Если , то

Если – действительные числа, то

Если – комплексные числа, то

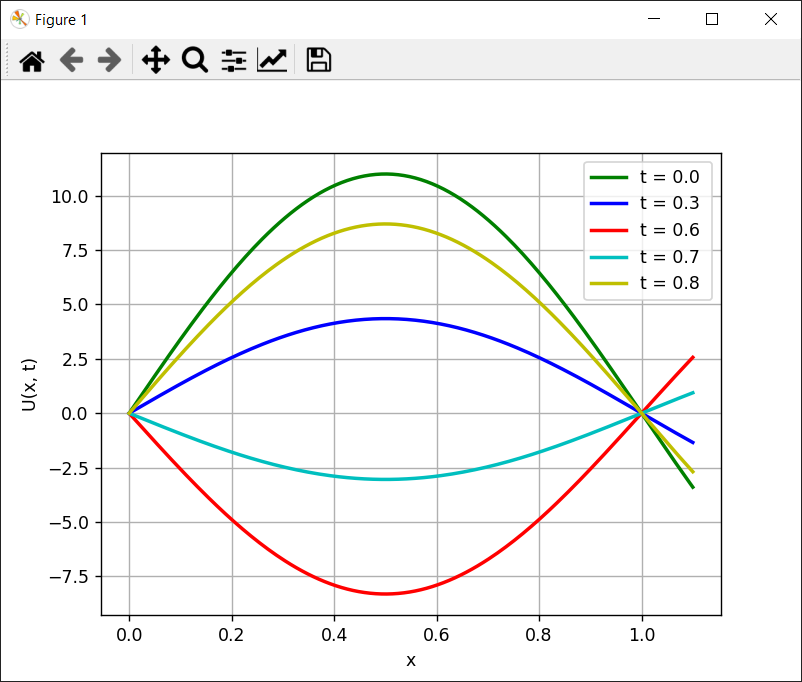
Собственные числа:

Собственные функции:

После подстановки получаем:

Решим относительно функции , составим характеристическое уравнение:

Проверка:

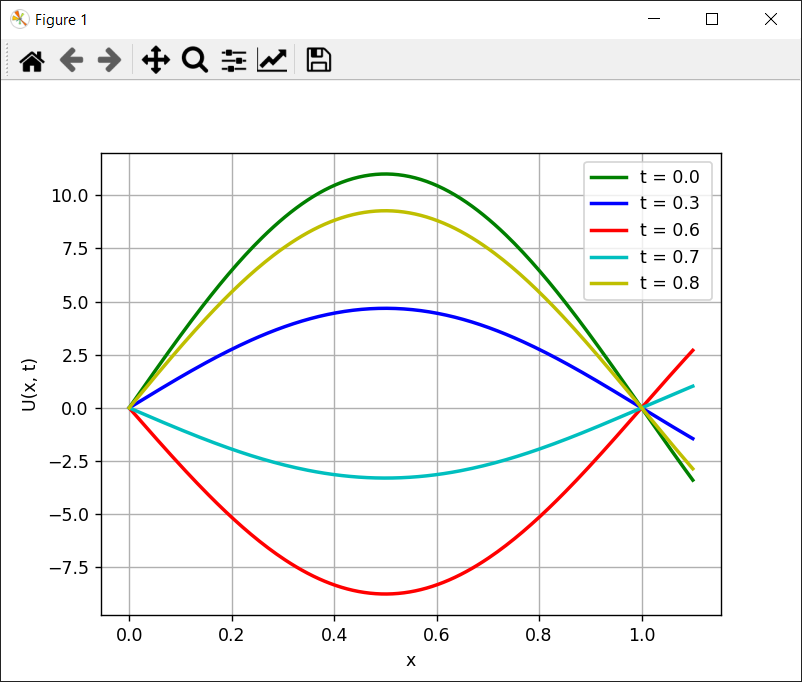


**Рис. 1.** Профиль струны

Теперь решим задачу методом отражений для струны с закрепленными концами.

Решение задачи:

Проверка:



**Рис. 2.** Профиль струны

**Задача 2.**

Решить методом Фурье начально-краевую задачу для уравнения теплопроводности.

* 1. Ответ представить в максимально компактной форме
  2. Построить графики изменения температуры в различные моменты времени, начиная с нулевого.

**Решение:**

Из условия имеем:

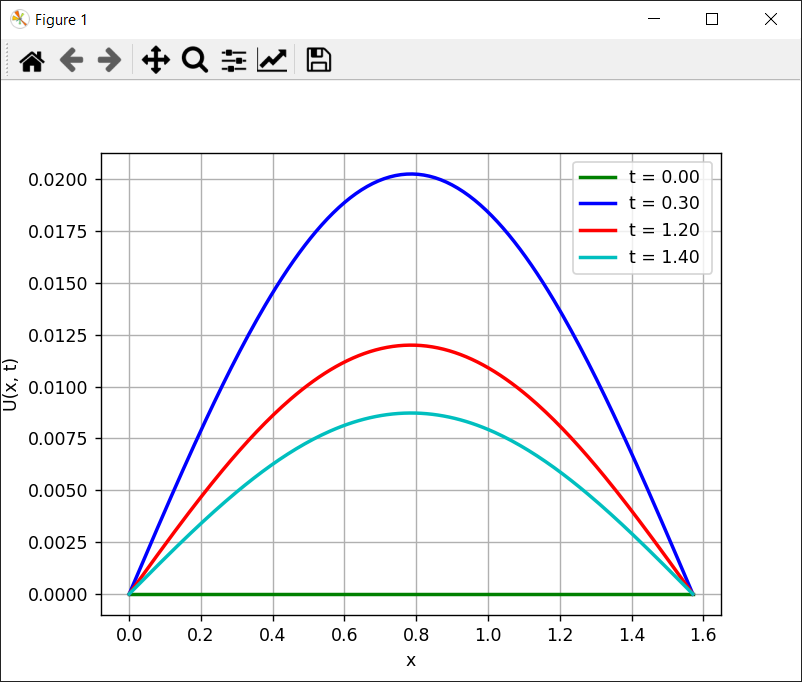
Решение задачи имеет вид:

После подстановки получаем:

Для каждого n должно выполняться равенство:

Пользуясь начальным условием для

Получаем начальное условие для . Решая обыкновенное дифференциальное уравнение с нулевым начальным условием, находим:



**Рис. 3.** Изменение температуры

**Задача 3.**

Решить внутреннюю задачу Дирихле для уравнения Лапласа в круговом секторе

Из условия имеем:

Решение уравнения имеет вид:

Подставляя выражение в уравнение Лапласа и разделяя переменные, получаем задачу Штурма-Лиувилля на отрезке для определения

А также задачу для определения

Решение задачи Штурма-Лиувилля имеет вид:

Подставив , получим:

Решение имеет вид:

После подстановки имеем:

Общее решение уравнения Эйлера:

Частные решения исходного уравнения:

Решение задачи Дирихле:

**Вывод:** в ходе выполнения домашней работы были получены практические навыки использования специализированных программных пакетов и библиотек для стандартных вычислений и визуализации результатов вычислений; навыками анализа возможностей построения и выделения наиболее важных свойств объектов моделей для моделирования; практических навыков решения ДУЧП2.

**ПРИЛОЖЕНИЯ**

**Листинг:**

**HW2\_1.py**

import matplotlib.pyplot as plt

import numpy as np

import math

def U(x, t):

    pi = np.pi

    r = 1/2\*(11\*math.sin(pi\*(x - 8\*t)) + 4\*(math.cos(pi\*(x - 8\*t)) - 1)) + \

        1/2\*(11\*math.sin(pi\*(x + 8\*t)) - 4\*(math.cos(pi\*(x + 8\*t)) - 1))

    print("r=",r)

    print(2/math.pi+t, "\n")

    return r

l = 6

a = 8

t = [0.0, 0.31, 0.61, 0.69, 0.79]

color = ['g-', 'b-', 'r-', 'c-', 'y-']

indx = 0

for tt in t:

     x = np.linspace(0, 1.1, num=200)

     print("x=",x)

     y = []

     print("tt=",tt)

     for i in x:

        y.append(U(i, tt))

     plt.plot(x, y, '%s' % color[indx], linewidth=2, label='t = %.1f '% tt)

     indx+=1

plt.xlabel('x')

plt.ylabel('U(x, t)')

plt.grid(True)

plt.legend(loc=0)

plt.show()

**HW2\_2.py**

import matplotlib.pyplot as plt

import numpy as np

import math

def U(x, t):

    pi = math.pi

    repetitions = 1

    r = 0.0

    for i in range(repetitions):

        n = i + 1

        temp = -(88\*n\*np.cos(pi\*n)\*(16\*n\*\*2\*np.sin(5\*t) - 5\*np.cos(5\*t) + 5\*np.exp(-16\*n\*\*2\*t))) \

            / (pi\*(4\*n\*\*2 - 81)\*(256\*n\*\*4 + 25))

        temp \*= np.sin(2\*n\*x)

        r += temp

    return r

t = [0.0, 0.3, 1.2, 1.4]

color = ['g-', 'b-', 'r-', 'c-']

indx = 0

for tt in t:

     x = np.linspace(0, math.pi/2, num=200)

     y = []

     for i in x:

        y.append(U(i, tt))

     plt.plot(x, y, '%s' % color[indx], linewidth=2, label='t = %.2f '% tt)

     indx+=1

plt.xlabel('x')

plt.ylabel('U(x, t)')

plt.grid(True)

plt.legend(loc=0)

plt.show()