Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 3

Тема: Программно- алгоритмическая реализация моделей на основе ОДУ второго порядка с краевыми условиями II и III рода

Студент Оберган Т.М.

Группа ИУ7-65Б

Оценка (баллы) ________________

Преподаватель Градов В.М.

Москва.
2020 г.

Оглавление

Цель работы: получение навыков разработки алгоритмов решения краевой задачи при реализации моделей, построенных на ОДУ второго порядка.

Исходные данные

1. Задана математическая модель.

Уравнение для функции $T(x)$:

$$\frac{d}{dx}\left(k(x)\frac{dT}{dx}\right) - \frac{2}{R}\alpha(x)T + \frac{2T_0}{R}\alpha(x) = 0 \tag{1}$$

Краевые условия:

$$\begin{cases} x = 0, -k(0)\frac{dT}{dx} = F_0 \\ x = l, -k(l)\frac{dT}{dx} = \alpha_N(T(l) - T_0) \end{cases} \tag{2}$$

2. Функции $k(x), \alpha(x)$ заданы константами:

$$k(x) = \frac{a}{x-b}$$
$$\alpha(x) = \frac{c}{x-d}$$

(3,4)

Константы a, b следует найти из условий $k(0) = k_0, k(l) = k_N$, а константы c, d из условий $\alpha(0) = \alpha_0, \alpha(l) = \alpha_N$. Получим:

$$a = -k_0 b$$
$$b = \frac{k_N l}{k_N - k_0}$$
$$c = -\alpha_0 d$$
$$d = \frac{\alpha_0 l}{\alpha_N - \alpha_0}$$

(5-8)

Величины $k_0, k_N, \alpha_0, \alpha_N$ задает пользователь.

3. Разностная схема с разностным краевым условием при $x = 0$.

$$A_n y_{n+1} - B_n y_n + C_n y_{n-1} = -D_n, \qquad 1 \le n \le N-1 \tag{9}$$

$$A_n = \frac{X_{n+\frac{1}{2}}}{h}$$

$$C_n = \frac{X_{n-\frac{1}{2}}}{h}$$

$$B_n = A_n + C_n + p_n h$$

$$D_n = f_n h$$

Система (9) совместно с краевыми условиями решается методом прогонки.

Для величин $X_{n+\frac{1}{2}}$ можно получить различные приближенные выражения, численно вычисляя интеграл методом трапеций или методом средних.

Для вычислений будет использоваться метод средних:

$$X_{n\pm\frac{1}{2}} = \frac{k_n + k_{n\pm1}}{2} \tag{10}$$

Разностный аналог краевого условия при $x = 0$:

$$\left(X_{\frac{1}{2}} + \frac{h^2}{8} p_{\frac{1}{2}} + \frac{h^2}{4} p_0\right) y_0 - \left(X_{\frac{1}{2}} - \frac{h^2}{8} p_{\frac{1}{2}}\right) y_1 = hF_0 + \frac{h^2}{4} (f_{\frac{1}{2}} + f_0) \tag{11}$$

Простая аппроксимация:

$$p_{\frac{1}{2}} = \frac{p_0 + p_1}{2}$$

$$f_{\frac{1}{2}} = \frac{f_0 + f_1}{2}$$

(12,13)

При выполнении лабораторной необходимо учесть, что

$$F_N = a_N (y_N - T_0)$$

$$F_{N-\frac{1}{2}} = X_{N-\frac{1}{2}} \frac{y_{N-1} - y_N}{h}$$

(14,15)

4. Значения параметров для отладки (все размерности согласованы)

$k_0 = 0.4$ Вт/см К,

$k_N = 0.1$ Вт/см К,

$\alpha_0 = 0.05$ Вт/см^2 К,

$\alpha_N = 0.01$ Вт/см^2 К,

$l =$ 10 см,

$T_0 = 300$К,

$R = 0.5$ см,

$F_0 = 50$ Вт/см^2.

Физическое содержание задачи

Сформулированная математическая модель описывает температурное поле $T(x)$ вдоль цилиндрического стержня радиуса R и длиной l, причем $R << l$ и температуру можно принять постоянной по радиусу цилиндра. Ось x направлена вдоль оси цилиндра и начало координат совпадает с левым торцом стержня. Слева при $x = 0$ цилиндр нагружается тепловым потоком F_0. Стержень обдувается воздухом, температура которого равна T_0. В результате происходит съем тепла с цилиндрической поверхности и поверхности правого торца при $x = l$.Функции $k(x), \alpha(x)$ являются, соответственно, коэффициентами теплопроводности материала стержня и теплоотдачи при обдуве,.

Листинг

```
# Лабораторная по моделированию №3
# Реализация модели на основу ОДУ второго
# порядка с краевыми условиями II и III рода.

import matplotlib.pyplot as plt
import numpy as np

# Исходные данные модели
def get_constants():
    b = (kN * N) / (kN - k0)
    d = (aN * N) / (aN - a0)
    a = -k0 * b
    c = -a0 * d
    return a, b, c, d

def k(x):
    return a / (x - b)

def alpha(x):
    return c / (x - d)

def p(x):
    return (2 * alpha(x)) / R

def f(x):
    return (2 * T0 * alpha(x)) / R


# Разностная схема
def A(n):
    return approc_plus_half(k, n) / h

def C(n):
    return approc_minus_half(k, n) / h

def B(n):
    return A(n) + C(n) + p(n) * h

def D(n):
    return f(n) * h


# Простая аппроксимация
def approc_plus_half(func, n):
    return (func(n) + func(n + h)) / 2

def approc_minus_half(func, n):
    return (func(n - h) + func(n)) / 2


# Краевые условия
# При x = 0
def left_boundary_condition():
    k0 = approc_plus_half(k, 0) + (h*h*approc_plus_half(p, 0) / 8) + ((h*h*p(0))
/ 4)
    M0 = -approc_plus_half(k, 0) + (h*h*approc_plus_half(p, 0) / 8)
    P0 = h*F0 + (h*h / 4) * (approc_plus_half(f, 0) + f(0))
    return k0, M0, P0

# При x = N
def right_boundary_condition():
    kN = (approc_minus_half(k, N) / h) - (h*approc_minus_half(p, N)/ 8)
```

```
    MN = -aN - (approc_minus_half(k, N) / h) - (h*p(N) / 4) -
(h*approc_minus_half(p, N) / 8)
    PN = -(h/4) * (f(N) + approc_minus_half(f, N)) - T0*aN
    return kN, MN, PN


if __name__ == "__main__":
    # Исходные данные
    k0 = 0.4
    kN = 0.1
    a0 = 0.05
    aN = 0.01
    N = 10
    T0 = 300
    R = 0.5
    F0 = 50
    h = 1e-3

    a, b, c, d = get_constants()

    k0, M0, P0 = left_boundary_condition()
    kN, MN, PN = right_boundary_condition()

    # Прямой ход
    eps = [0, -M0 / k0]
    eta = [0, P0 / k0]

    x = h
    n = 1
    while x + h < N:
        eps.append(C(x) / (B(x) - A(x) * eps[n]))
        eta.append((A(x) * eta[n] + D(x)) / (B(x) - A(x) * eps[n]))
        n += 1
        x += h

    # Обратный ход
    t = [0] * (n + 1)

    t[n] = (PN - MN * eta[n]) / (kN + MN * eps[n])

    for i in range(n - 1, -1, -1):
        t[i] = eps[i + 1] * t[i + 1] + eta[i + 1]

    # График
    x = [i for i in np.arange(0, N, h)]

    plt.plot(x, t[:-1], 'r-')
    plt.xlabel("x, cm")
    plt.ylabel("temperature, K")
    plt.grid()
    plt.show()
```

Результаты работы

1. Представить разностный аналог краевого условия при $x = l$ и его краткий вывод интегро-интерполяционным методом.

Пусть

$$F = -k(x)\frac{dT}{dx}$$

$$p(x) = \frac{2}{R}\alpha(x)$$

$$f(x) = \frac{2T_0}{R}\alpha(x)$$

Тогда (1) можно записать как

$$-\frac{d}{dx}(F) - p(x) + f(x) = 0 \tag{16}$$

Проинтегрируем (15) на отрезке $[x_{N-\frac{1}{2}}; x_N]$:

$$-\int_{X_{N-\frac{1}{2}}}^{X_N} \frac{dF}{dx}dx - \int_{X_{N-\frac{1}{2}}}^{X_N} p(x)udx + \int_{X_{N-\frac{1}{2}}}^{X_N} f(x)dx = 0 \tag{17}$$

Второй и третий интегралы вычислим методом трапеций:

$$-\left(F_N - F_{N-\frac{1}{2}}\right) - \frac{h}{4}\left(p_N y_N + p_{N-\frac{1}{2}} y_{N-\frac{1}{2}}\right) + \frac{h}{4}\left(f_N + f_{N-\frac{1}{2}}\right) = 0 \tag{18}$$

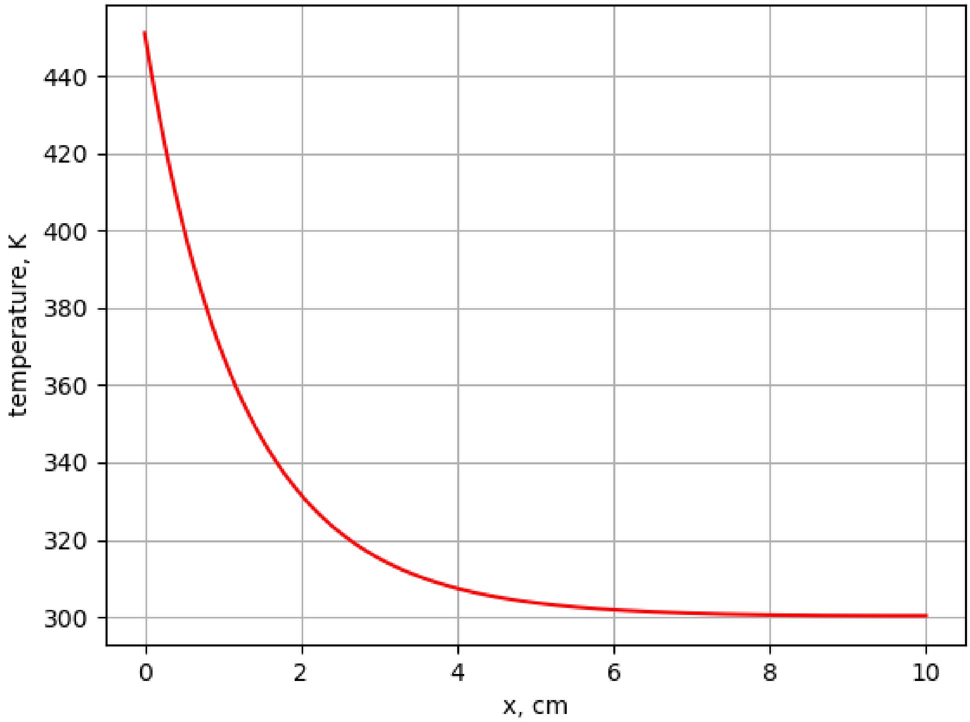
Применив (14, 15) к (18) получим:

$$-\left(a_N(y_N - T_0) - X_{N-\frac{1}{2}}\frac{y_{N-1} - y_N}{h}\right) - \frac{h}{4}\left(p_N y_N + p_{N-\frac{1}{2}}\frac{y_N + y_{N-1}}{2}\right) + \frac{h}{4}\left(f_N + f_{N-\frac{1}{2}}\right) = 0$$

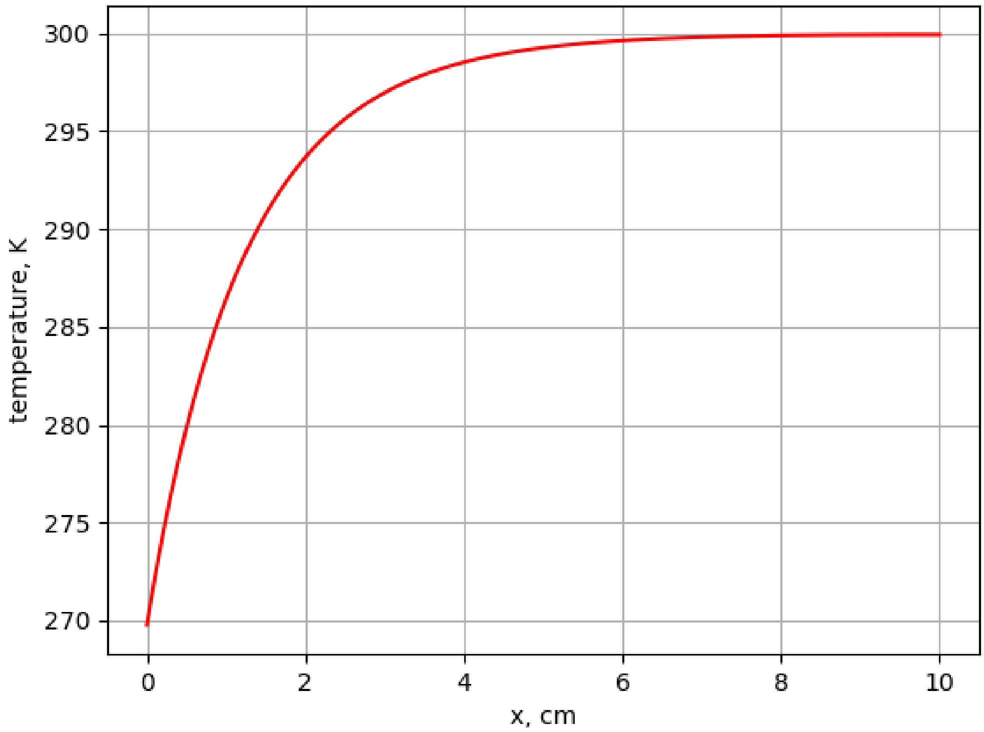
$$X_{N-\frac{1}{2}}\frac{y_{N-1} - y_N}{h} - \frac{h}{4}p_N y_N - \frac{h}{4}p_{N-\frac{1}{2}}\frac{y_N + y_{N-1}}{2} = -\frac{h}{4}\left(f_N + f_{N-\frac{1}{2}}\right) + a_N(y_N - T_0)$$

$$y_{N-1}\left(\frac{X_{N-\frac{1}{2}}}{h} - \frac{h}{8}p_{N-\frac{1}{2}}\right) + y_N\left(-a_N - \frac{X_{N-\frac{1}{2}}}{h} - \frac{h}{4}p_N - \frac{h}{8}p_{N-\frac{1}{2}}\right) = -\frac{h}{4}\left(f_N + f_{N-\frac{1}{2}}\right) - T_0 a_N \tag{19}$$

2. График зависимости температуры $T(x)$ от координаты x при заданных выше параметрах.

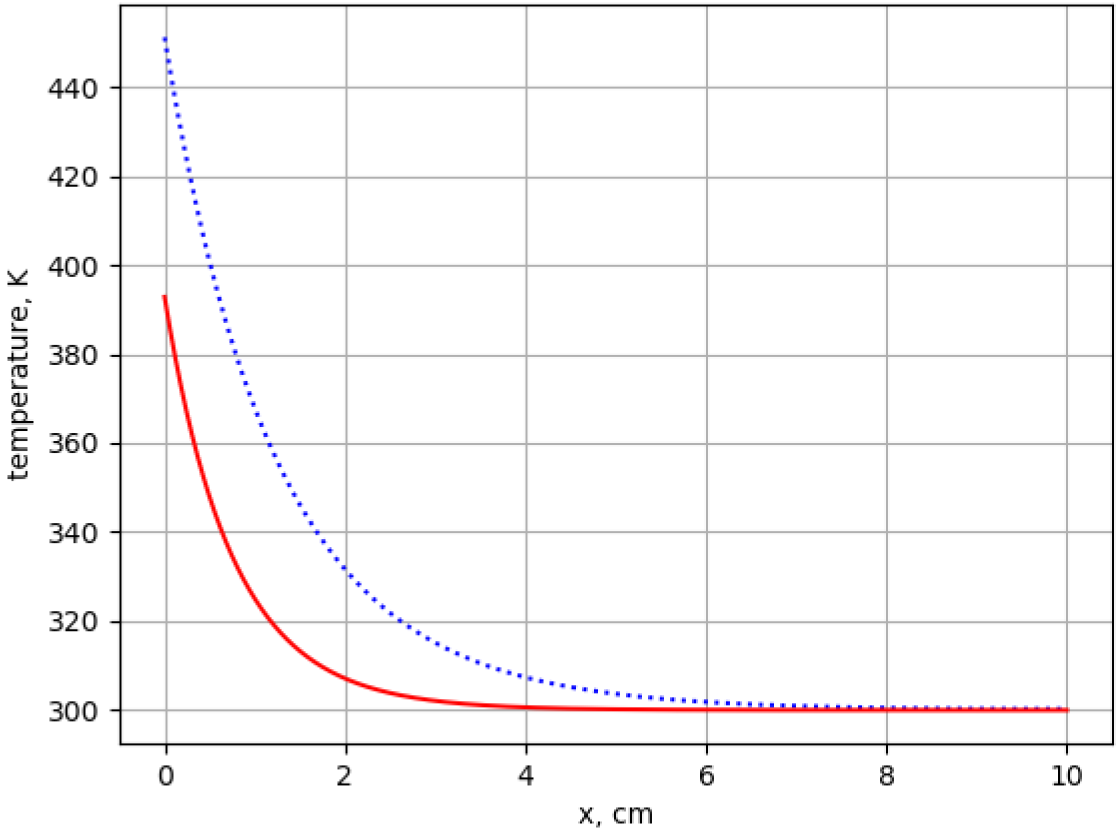


3. График зависимости $T(x)$ при F_0 =-10 Вт/см^2.



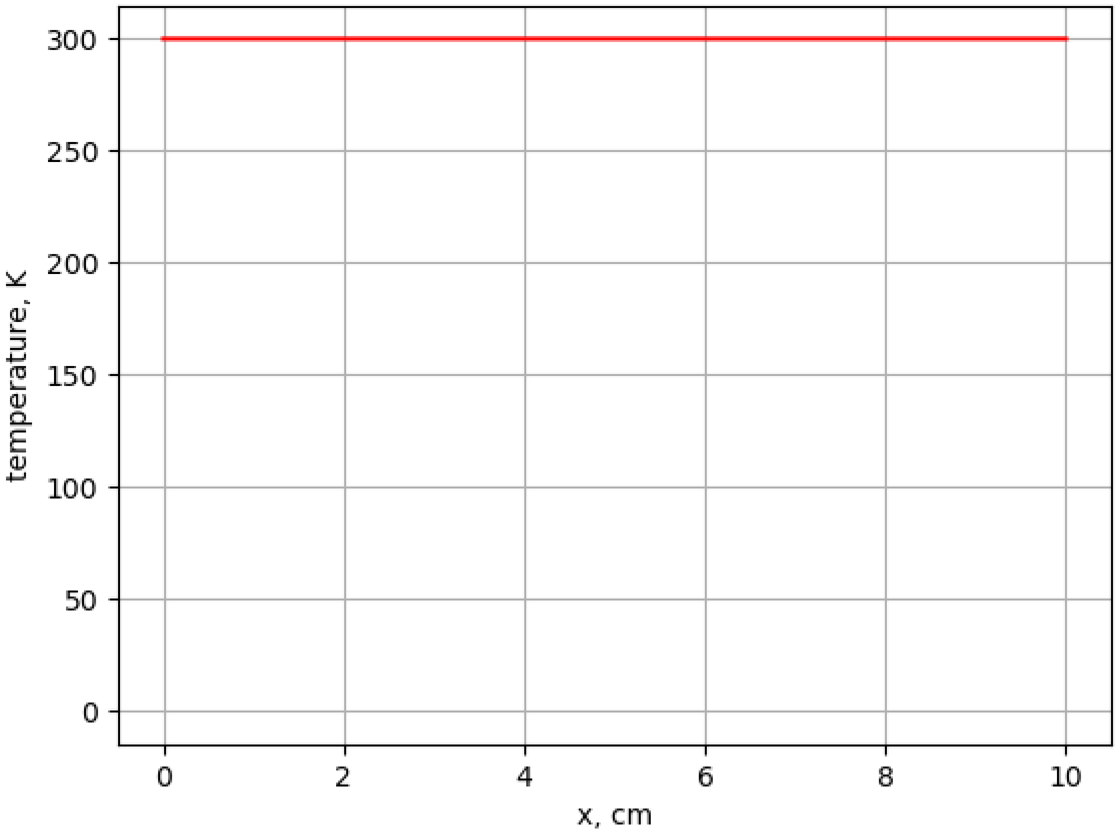
Справка. При отрицательном тепловом потоке слева идет съем тепла, поэтому производная $T'(x)$ должна быть положительной.

4. График зависимости $T(x)$ при увеличенных значениях $\alpha(x)$ (например, в 3 раза). Сравнить с п.2.

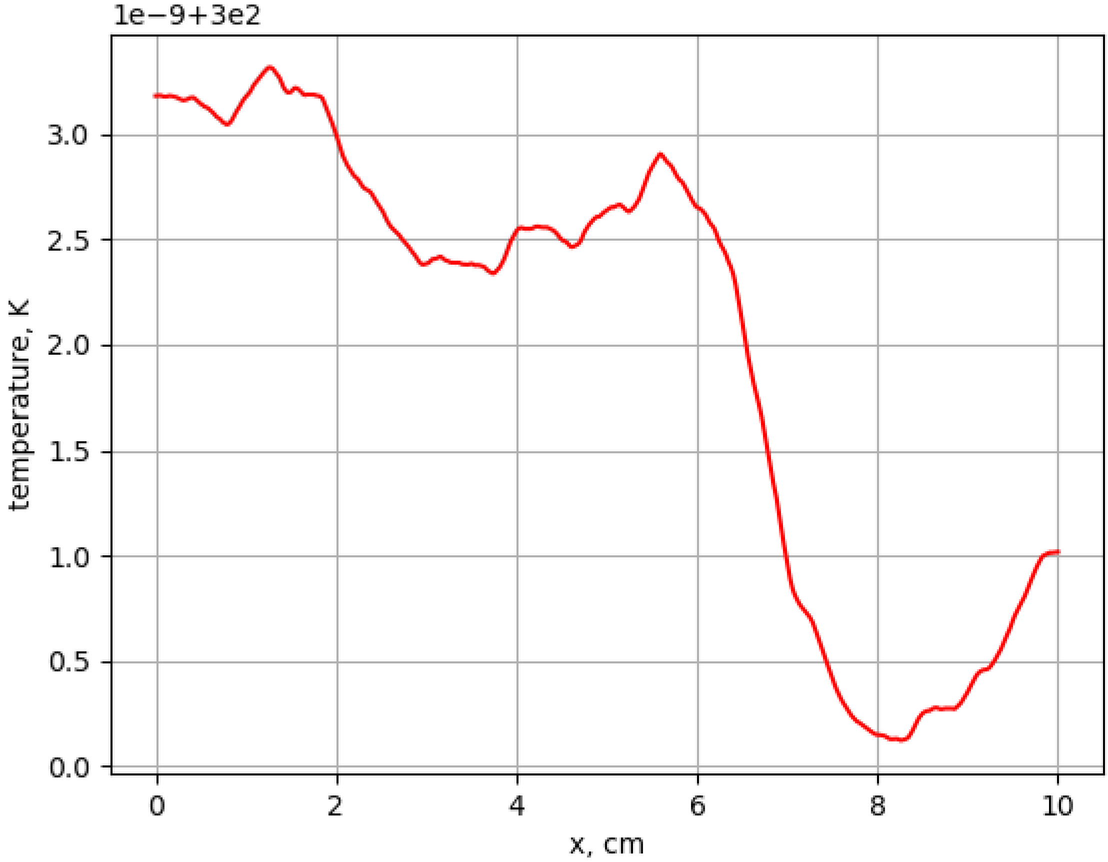


Справка. При увеличении теплосъема и неизменном потоке F_0 уровень температур $T(x)$ должен снижаться, а градиент увеличиваться.

5. График зависимости $T(x)$ при $F_0 = 0$.



На следующем графике видна погрешность:



Справка. В данных условиях тепловое нагружение отсутствует, причин для нагрева нет, температура стержня должна быть равна температуре окружающей среды T_0 (разумеется, с некоторой погрешностью, определяемой приближенным характером вычислений).

Вопросы при защите лабораторной работы

Ответы на вопросы дать письменно в Отчете о лабораторной работе.

1. Какие способы тестирования программы можно предложить?

Меняя коэффициенты, можно оценивать результат исходя из физического смысла задачи. Например, как было продемонстрировано в лабораторной, при F0 = 0 температура стержня должна быть равна температуре среды, а при отрицательных значениях идет съем тепла, температура должна увеличиваться.

2. Получите простейший разностный аналог нелинейного краевого условия при

$$x = l, -k(l)\frac{dT}{dx} = \alpha_n(T(l) - T_0) + \varphi(T)$$

где $\varphi(T)$ – заданная функция.

Производную аппроксимируйте односторонней разностью.

$$-k_n\frac{y_N - y_{N-1}}{h} = \alpha_N(y_N - T_0) + \varphi(y_N) \tag{20}$$

3. Опишите алгоритм применения метода прогонки, если при $x = 0$ **краевое условие линейное (как в настоящей работе), а при** $x = l$, **как в п.2.**

Будет использоваться правая прогонка.

$$y_n = \varepsilon_{n+1}y_{n+1} + \eta_{n+1} \tag{21}$$

Начальные прогоночные коэффициенты:

$$\varepsilon_1 = 1$$

$$\eta_1 = \frac{hF_0}{k_0} \tag{22}$$

Прогоночные коэффициенты:

$$\varepsilon_{n+1} = \frac{C_n}{B_n - A_n \varepsilon_n}$$

$$\eta_{n+1} = \frac{D_n + A_n \eta_n}{B_n - A_n \varepsilon_n}$$

(23)

Зная, что $y_{n-1} = \varepsilon_n y_n + \eta_n$, найдем y_n:

В (20) подставим (21):

$$-k_N \frac{y_N - (\varepsilon_N y_N + \eta_N)}{h} = \alpha_N y_N - \alpha_N T_0 + \varphi(y_N)$$

$$-k_N y_N + \varepsilon_N k_N y_N + k_N \eta_N = h\alpha_N y_N - h\alpha_N T_0 + h\varphi(y_N)$$

$$y_N(-k_N + k_N \varepsilon_N - h\alpha_N) + \eta_N + h\alpha_N T_0 - h\varphi(y_N) = 0$$

(24)

4. Опишите алгоритм определения единственного значения сеточной функции y_p в одной заданной точке p. Использовать встречную прогонку, т.е. комбинацию правой и левой прогонок (лекция №8). Краевые условия линейные.

Левая прогонка:

$$y_n = \alpha_{n+1} y_{n+1} + \beta_{n+1}, \qquad 0 \le n \le p$$

Прогоночные коэффициенты:

$$\alpha_{n-1} = \frac{C_n}{B_n - A_n \alpha_n}$$

$$\beta_{n-1} = \frac{D_n + A_n \beta_n}{B_n - A_n \alpha_n}$$

Объединив левую и правую (21), (23) прогонки

Получим систему:

$$\begin{cases} y_p = \varepsilon_{p+1}\, y_{p+1} + \eta_{p+1} \\ y_{p+1} \qquad = \alpha_p\, y_p + \beta_p \end{cases}$$

Подставив второе уравнение в первое:

$$y_n = \frac{\varepsilon_{p+1}\beta_p + \eta_{p+1}}{1 - \varepsilon_{p+1}\alpha_p}$$