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 **ПРОГРАМНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ (ИУ7)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.03 ПРОГРАММНАЯ ИНЖЕНЕРИЯ**

О Т Ч Е Т

По лабораторной работе № 4

Дисциплина: Моделирование

Студент ИУ7-62Б _______________ Н.А. Гарасев
(Группа) (Подпись, дата) (И.О. Фамилия)

Преподаватель _______________ В.М. Градов
(Подпись, дата) (И.О. Фамилия)

Москва, 2021

Тема: Программно-алгоритмическая реализация моделей на основе дифференциальных уравнений в частных производных с краевыми условиями II и III рода.

Цель работы. Получение навыков разработки алгоритмов решения смешанной краевой задачи при реализации моделей, построенных на квазилинейном уравнении параболического типа.

Исходные данные:

1. Дано:

$K(T) = a_1(b_1 + c_1T^{m1})$, Вт/см К,

$c(T) = a_2 + b_2\, T^{m_2} - \dfrac{c_2}{T^2}$, Дж/см^{3}К.

a_1=0.0134, b_1=1, c_1=4.35 10^{-4}, m_1=1,

a_2=2.049, b_2=0.563 10^{-3}, c_2=0.528 10^5, m_2=1.

$\alpha(x) = \dfrac{c}{x-d}$,

$\alpha_0 = 0.05$ Вт/см^2 К,

$\alpha_N = 0.01$ Вт/см^2 К,

$l =$ 10 см,

$T_0 = 300$К,

$R = 0.5$ см,

$F(t) = 50$ Вт/см^2 (для отладки принять постоянным).

2. Уравнение:

$$c\,(T)\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(k(T)\frac{\partial T}{\partial x}\right) - \frac{2}{R}\alpha(x)T + \frac{2T_0}{R}\alpha(x)$$

3. В обозначениях уравнения лекции

$$F = -k(T)\frac{\partial T}{\partial x}$$
$$p(x) = \frac{2}{R}\alpha(x)$$
$$f(x) = \frac{2T_0}{R}\alpha(x)$$

4. Краевые условия

$$\begin{cases} t = 0, \quad T(x,0) = T_0, \\ x = 0, \quad -k(T(0))\,\dfrac{\partial T}{\partial x} = F_0, \\ x = l, \quad -k(T(l))\,\dfrac{\partial T}{\partial x} = \alpha_N\,(T(l) - T_0) \end{cases}$$

Функция a(x) по условию:

$$\alpha(x) = \frac{c}{x-d},$$

5. Разностная схема

$$\hat{A}_n\,\hat{y}_{n-1} - \hat{B}_n\,\hat{y}_n + \hat{D}_n\,\hat{y}_{n+1} = -\hat{G}_n,$$

где

$$\hat{A}_n = \hat{\chi}_{n-1/2}\frac{\tau}{h},$$

$$\hat{D}_n = \hat{\chi}_{n+1/2}\frac{\tau}{h},$$

$$\hat{B}_n = \hat{A}_n + \hat{D}_n + \hat{c}_n\,h + p_n h\tau,$$

$$\hat{F}_n = f_n h\tau + \hat{c}_n\,y_n\,h.$$

6. Краевые условия

Разностные аналоги краевых условий при $x = 0$:

$$\left(\frac{h}{8}\hat{c}_{1/2}+\frac{h}{4}\hat{c}_0+\hat{\chi}_{1/2}\frac{\tau}{h}+\frac{\tau h}{8}p_{1/2}+\frac{\tau h}{4}p_0\right)\hat{y}_0+\left(\frac{h}{8}\hat{c}_{1/2}-\hat{\chi}_{1/2}\frac{\tau}{h}+\frac{\tau h}{8}p_{1/2}\right)\hat{y}_1=$$

$$=\frac{h}{8}\hat{c}_{1/2}\left(y_0+y_1\right)+\frac{h}{4}\hat{c}_0\,y_0+\hat{F}\tau+\frac{\tau h}{4}\left(\hat{f}_{1/2}+\hat{f}_0\right)$$

, взяты из лекций 10-11.

Разностные аналоги краевых условий при x=l:

$$c(T)\frac{\partial T}{\partial t}=\frac{\partial}{\partial x}\left(k(T)\frac{\partial T}{\partial x}\right)-\frac{2}{R}\alpha(x)T+\frac{2T_0}{R}\alpha(x)$$

$$F=-k(T)\frac{\partial T}{\partial x}$$

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

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

Проинтегрируем ур-е на отрезке $[x_{N-\frac{1}{2}}; x_N]$ и на временном интервале $[t_m, t_{m+1}]$

$$\int_{x_{N-\frac{1}{2}}}^{x_N}dx\int_{t_m}^{t_{m+1}}c(T)\frac{\partial T}{\partial t}dt=-\int_{t_m}^{t_{m+1}}dt\cdot\int_{x_{N-\frac{1}{2}}}^{x_N}\frac{\partial F}{\partial x}dx-$$

$$-\int_{x_{N-\frac{1}{2}}}^{x_N}dx\int_{t_m}^{t_{m+1}}p(x)T\,dt+\int_{x_{N-\frac{1}{2}}}^{x_N}dx\int_{t_m}^{t_{m+1}}f(T)\,dt$$

$$\int_{x_{N-\frac{1}{2}}}^{x_N}\hat{c}(\hat{T}-T)\,dx=-\int_{t_m}^{t_{m+1}}(F_N-F_{N-\frac{1}{2}})\,dt-\int_{x_{N-\frac{1}{2}}}^{x_N}p\hat{T}\tau\,dx+$$

$$+\int_{x_{N+\frac{1}{2}}}^{x_N}\hat{f}\tau\,dx$$

$$\left(\hat c_N(\hat y_N - y_N) + \hat c_{N-\frac12}(\hat y_{N-\frac12} - y_{N-\frac12})\right)\frac{h}{4} = -\tau(\hat F_N - \hat F_{N-\frac12}) -$$

$$-(p_N \hat y_N + p_{N-\frac12}\hat y_{N-\frac12})\tau\cdot\frac{h}{4} + (\hat f_N + \hat f_{N-\frac12})\cdot\tau\cdot\frac{h}{4}$$

$$\hat F_N = \alpha_N(\hat y_N - T_0); \quad \hat F_{N-\frac12} = \hat x_{N-\frac12}\frac{\hat y_{N-1} - \hat y_N}{h}$$

$$\hat c_N \hat y_N \frac{h}{4} - \hat c_N y_N \frac{h}{4} + \hat c_{N-\frac12}\cdot y_N \frac{h}{8} + \hat c_{N-\frac12} y_{N-\frac12}\frac{h}{8} - \hat c_{N-\frac12}\hat y_N \frac{h}{8} -$$

$$-\hat c_{N-\frac12}\hat y_{N-1}\frac{h}{8} = -\tau\cdot\alpha_N\cdot\hat y_N + \tau\cdot\alpha_N\cdot T_0 + \tau\cdot x_{N-\frac12}\cdot\frac{\hat y_{N-1}}{h} -$$

$$-\tau\cdot\hat x_{N-\frac12}\cdot\frac{\hat y_N}{h} - p_N\hat y_N\cdot\tau\cdot\frac{h}{4} - p_{N-\frac12}\cdot\hat y_N\cdot\frac{h}{8} - p_{N-\frac12}\hat y_{N-1}\cdot\tau\cdot\frac{h}{8} +$$

$$+(\hat f_N + \hat f_{N-\frac12})\cdot\tau\cdot\frac{h}{4}$$

$$\hat y_{N-1}\left(\hat c_{N-\frac12}\frac{h}{8} - \frac{\tau\hat x_{N-\frac12}}{h} + p_{N-\frac12}\tau\frac{h}{8}\right) +$$

$$+\hat y_N\left(\hat c_N\frac{h}{4} + \hat c_{N-\frac12}\frac{h}{8} + \tau\alpha_N + \frac{\tau\hat x_{N-\frac12}}{h} + p_N\tau\frac{h}{4} + p_{N-\frac12}\tau\frac{h}{8}\right) =$$

$$= \hat c_{N-\frac12}\left(y_N\frac{h}{8} + y_{N-1}\frac{h}{8}\right) + \hat c_N y_N\frac{h}{4} + \tau\alpha_N T_0 + (\hat f_N + \hat f_{N-\frac12})\tau\frac{h}{4}$$

Используя полученные краевые условия, можно найти коэффициенты *K0, M0, P0, KN, MN, PN*.

$$\begin{cases} \hat K_0\,\hat y_0 + \hat M_0\hat y_1 = \hat P_0, \\ \hat A_n\,\hat y_{n-1} - \hat B_n\,\hat y_n + \hat D_n\,\hat y_{n+1} = -\hat F_n, \quad 1 \le n \le N-1, \\ \hat K_N\,\hat y_N + \hat M_{N-1}\hat y_{N-1} = \hat P_N \end{cases}$$

7. Метод простых итераций.

$$\widehat{A}_n^{s-1}\,\widehat{y}_{n+1}^s - \widehat{B}_n^{s-1}\,\widehat{y}_n^s + \widehat{D}_n^{s-1}\,\widehat{y}_{n-1}^s = -\widehat{F}_n^{s-1},$$

, где s – номер итерации.

Итерации прекращаются при условии:

$$\max\left|\frac{y_n^s - y_n^{s-1}}{y_n^s}\right| \le \varepsilon_1, \text{ для всех } n = 0,1,\dots N$$

Листинг программы

Листинг 1. Данные.

```
import matplotlib.pyplot as plt
import numpy as np
from math import fabs

a1 = 0.0134
b1 = 1
c1 = 4.35e-4
m1 = 1
a2 = 2.049
b2 = 0.563e-3
c2 = 0.528e5
m2 = 1
alpha0 = 0.05
alphaN = 0.01
l = 10
T0 = 300
R = 0.5
F0 = 50
h = 0.01
t = 1
```

Листинг 2. Вспомогательные функции

```
def k(T):
    return a1 * (b1 + c1 * T ** m1)

def c(T):
    return a2 + b2 * T ** m2 - c2 / T ** 2

def alpha(x):
    d = (alphaN * l) / (alphaN - alpha0)
    c = - alpha0 * d
    return c / (x - d)

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

def f(x):
```

```
    return 2 * alpha(x) * T0 / R

def func_plus_1_2(x, step, func):
    return (func(x) + func(x + step)) / 2

def func_minus_1_2(x, step, func):
    return (func(x) + func(x - step)) / 2

def A(T):
    return func_minus_1_2(T, t, k) * t / h

def D(T):
    return func_plus_1_2(T, t, k) * t / h

def B(x, T):
    return A(T) + D(T) + c(T) * h + p(x) * h * t

def F(x, T):
    return f(x) * h * t + c(T) * T * h
```

Листинг 3. Расчет коэффициентов.

```
K0 = h / 8 * func_plus_1_2(prevT[0], t, c) + h / 4 * c(prevT[0]) + \
                    func_plus_1_2(prevT[0], t, k) * t / h + \
                    t * h / 8 * p(h / 2) + t * h / 4 * p(0)

M0 = h / 8 * func_plus_1_2(prevT[0], t, c) - \
                    func_plus_1_2(prevT[0], t, k) * t / h + \
                    t * h * p(h / 2) / 8

P0 = h / 8 * func_plus_1_2(prevT[0], t, c) * (prevT[0] + prevT[1]) + \
                    h / 4 * c(prevT[0]) * prevT[0] + \
                    F0 * t + t * h / 8 * (3 * f(0) + f(h))

KN = h / 8 * func_minus_1_2(prevT[-1], t, c) + h / 4 * c(prevT[-1]) + \
                    func_minus_1_2(prevT[-1], t, k) * t / h + t * alphaN + \
                    t * h / 8 * p(l - h / 2) + t * h / 4 * p(l)

MN = h / 8 * func_minus_1_2(prevT[-1], t, c) - \
                    func_minus_1_2(prevT[-1], t, k) * t / h + \
                    t * h * p(l - h / 2) / 8

PN = h / 8 * func_minus_1_2(prevT[-1], t, c) * (prevT[-1] + prevT[-2]) + \
                    h / 4 * c(prevT[-1]) * prevT[-1] + t * alphaN * T0 + \
                    t * h / 4 * (f(l) + f(l - h / 2))
```

Листинг 4. Прямой ход.

```
eps = [0, -M0 / K0]
eta = [0, P0 / K0]

x = h
n = 1
while (x + h < l):
    eps.append(D(prevT[n]) / (B(x, prevT[n]) - A(prevT[n]) * eps[n]))
    eta.append((F(x, prevT[n]) + A(prevT[n]) * eta[n]) / (B(x, prevT[n]) \
                                                    - A(prevT[n]) * eps[n]))
    n += 1
    x += h
```

Листинг 5. Обратный ход.

```
y = [0] * (n + 1)
y[n] = (PN - MN * eta[n]) / (KN + MN * eps[n])

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

return y
```

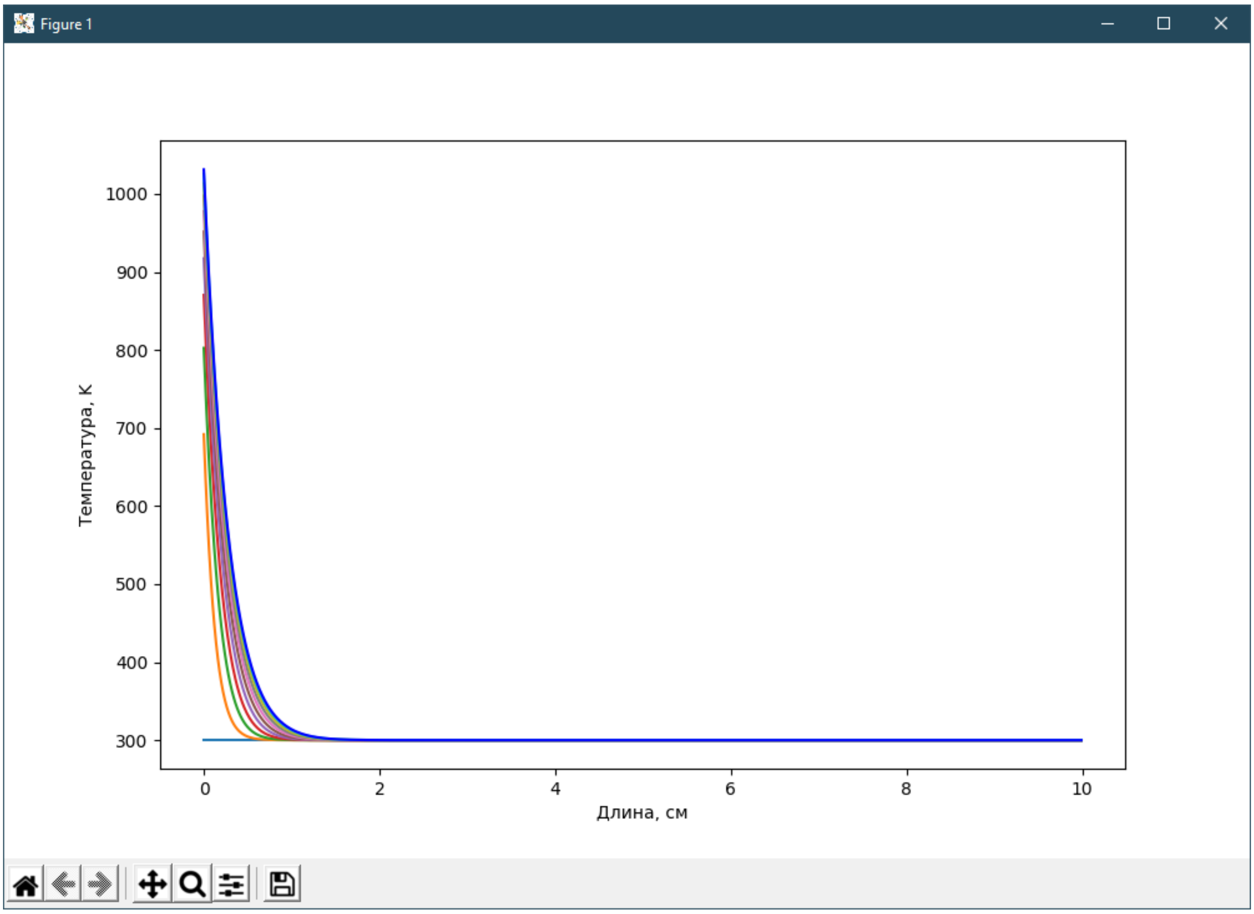
Листинг 6. Проверка выхода из цикла.

```
def check_eps(T, newT):
    for i, j in zip(T, newT):
        if fabs((i - j) / j) > 1e-2:
            return True
    return False

def check_iter(T, newT):
    max = fabs((T[0] - newT[0]) / newT[0])
    for i, j in zip(T, newT):
        d = fabs(i - j) / j
        if d > max:
            max = d
    return max < 1
```

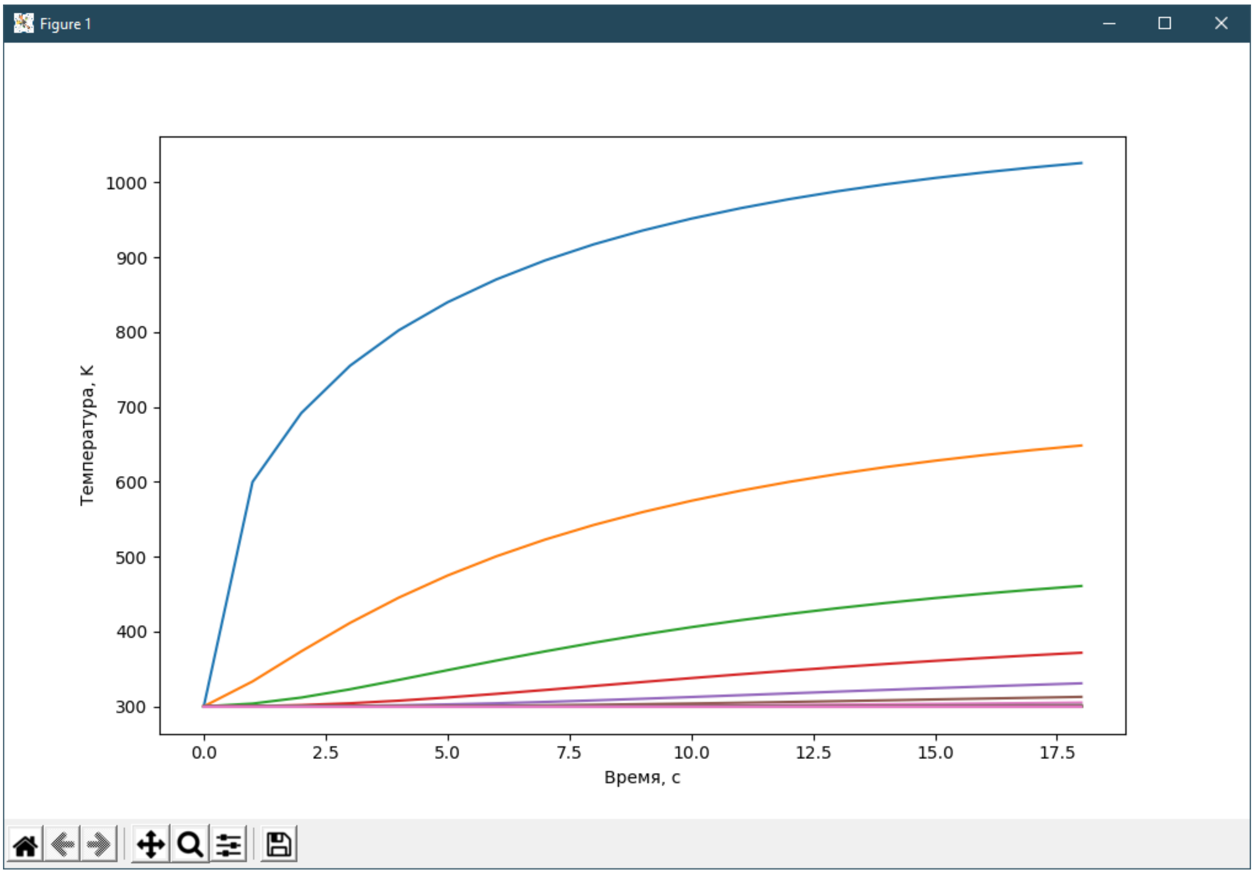
Результаты работы.

График зависимости температуры от координаты при нескольких фиксированных значениях времени.



При фиксированных значениях времени t = 0, 2, 4, 8, …. Синяя кривая (самая нижняя) соответствует установившемуся режиму, когда поле меняется с точность eps = 1e-3.

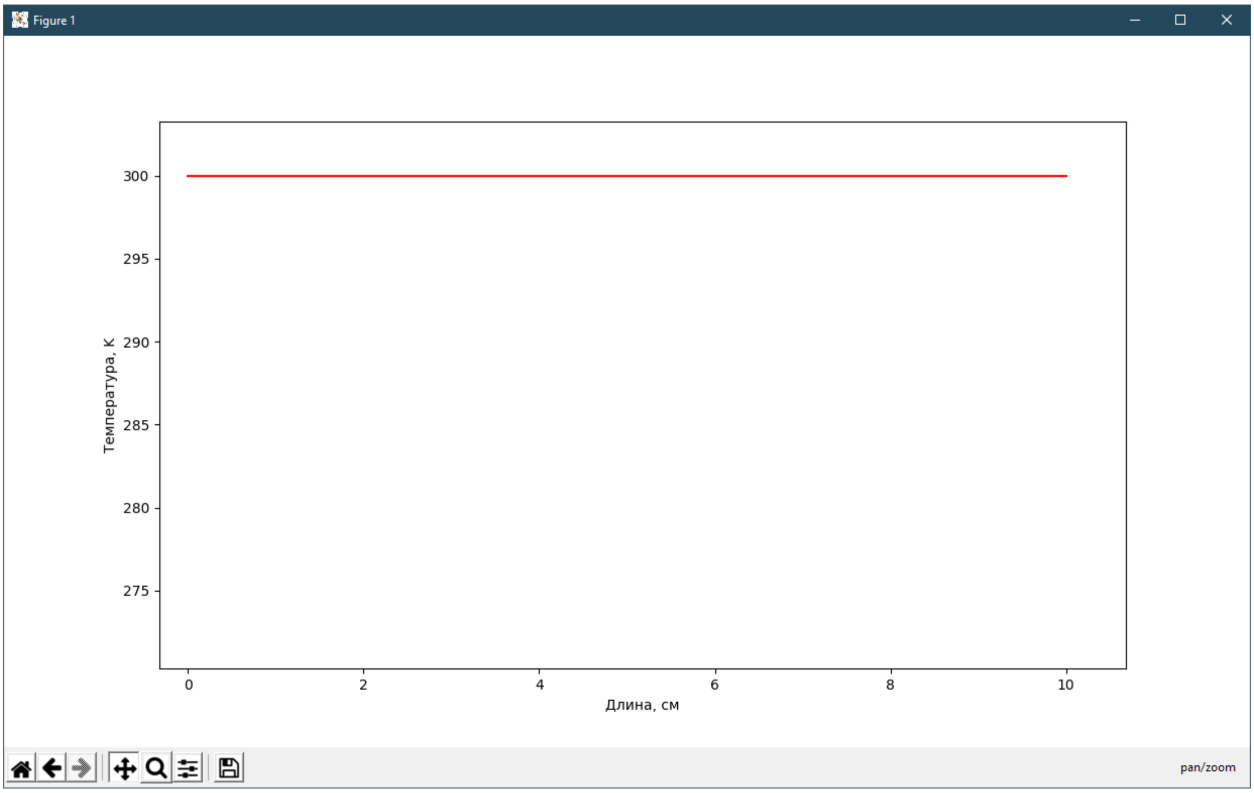
График зависимости температуры от времени при нескольких фиксированных значениях координаты.



Длина *x* изменяется от 0,2 до 3,2 с шагом 0,2.

Ответы на вопросы.

Протестируем программу на примерах из третьей лабораторной работы. При $F_0 = 0$, температура стержня должна оставаться неизменной и должна быть равна температуре внешней среды T_0.



При отрицательном тепловом потоке (F_0) происходит съем тепла. Чему соответствуют графики, представленные ниже.