МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО

Физико-механический институт

Высшая школа прикладной математики и вычислительной физики

Отчет по курсовому проекту  
по курсу:

ЧИСЛЕННОЕ РЕШЕНИЕ КРАЕВЫХ ЗАДАЧ

на тему:

Исследование нестационарного поля температур в плоской неограниченной пластине с использованием неявного метода конечных разностей

Работу выполнил:

студент группы 5030301/10002

Тугай В.В.

Преподаватель:

к.т.н., доц. Плетнев А.А.

Санкт-Петербург

2023

# Содержание

1. Физическая постановка задачи
2. Математическая постановка задачи
3. Метод решения
4. Тестовый расчет
5. Результаты решения задачи
6. Выводы
7. Приложение

# Физическая постановка задачи

Плоская неограниченная пластина из бронзы толщиной  60 см испытывает конвективный теплообмен с окружающей средой (с обеих сторон пластины интенсивность конвективного теплообмена одинакова). В начальный момент времени температура пластины постоянна во всем сечении и равна  500 °С. Температура окружающей среды  130 °С. Найти распределение температуры пластины в зависимости от координаты и времени для трех значений коэффициента конвективной теплоотдачи:

α1 = 35 Вт/(м²⋅К);   α2= 400 Вт/(м²⋅К);   α3= 25000 Вт/(м²⋅К).

*T*– температура пластины, K

*Tw*– температура на границе пластины, K

*Te*– температура окружающей среды, K

*T0*– начальная температура пластины, K

*τ* – время, с

*x*– координата, м

*δ* – толщина пластины, м

*q*– плотность теплового потока, Вт/м2

a – коэффициент конвективной теплоотдачи, Вт/(м2×К)

Материал пластины: бронза. Физические свойства материала и значения физический величин (Вариант-7):

*p*(1)=8600 кг/м2

*c*(2)=380 Дж/(К⋅кг)

*λ*=110 Вт/(м⋅К)

*δ*=0.6 м

*Te*=130 С˚

*T0*=500 С˚

(1) http://ru.solverbook.com/spravochnik/ximiya/plotnost/plotnost-bronzy/

(2) http://thermalinfo.ru/eto-interesno/tablitsy-udelnoj-teploemkosti-veshhestv

# Математическая постановка задачи

Diagram, engineering drawing

Description automatically generatedНачало координат – в центре пластины.  
Искомая функция – температура, которая зависит от двух переменных: координаты и времени

Плотность теплового потока (закон Фурье) : , где — коэффициент теплопроводности, Вт/(м⋅К)

Начальное условие (НУ):  
 – однородный профиль температуры

Граничные условия (ГУ):

– ГУ симметрии

– ГУ 3-го рода

Переход к безразмерным величинам:

— безразмерная координата ( X∈[0,1] )

— безразмерная избыточная температура( θ∈[0,1] )

— критерий Фурье (безразмерное время)

— число Био (безразмерный коэффициент теплоотдачи)

Уравнение теплопроводности в безразмерных величинах:

НУ и ГУ в безразмерных величинах:

# Метод решения

Чтобы снять ограничение для явного метода конечных разностей , аппроксимируем производную по Х на «старшем» временном слое :

При записи и аппроксимации уравнения баланса тепла для граничных узлов нам необходимо использовать выражения для теплового потока на границах, заданные согласно граничным условиям, а именно:

ГУ на левой границе:

ГУ на правой границе:

Тогда на левой границе (i=1) вместо (1) будем иметь сеточное уравнение

а на правой границе (i=Im)

Уравнение (3) отличается от (1) тем, что в нем содержатся сразу три неизвестных значения: , и , которые нельзя выразить только из одного уравнения.

Однако если записать совокупность сеточных уравнений для каждого узла (включая граничные), то мы получим замкнутую СЛАУ – систему линейных алгебраических уравнений, решением которой будут искомые значения (i = 1…Im) на «старшем» временном слое.

Применение неявнойсхемы свелось к тому, чтобы:

а) записать СЛАУ (т.е. из выражений (1), (2), (3) сформировать матрицу коэффициентов);

б) решить полученную СЛАУ.

Сформируем СЛАУ:

Для формирования матрицы СЛАУ сеточное уравнение представим в виде

Легко видеть, что согласно (1) для внутренних узлов (i = 2…Im-1)

Для левой границы из уравнения (2) имеем (\*)

Для правой границы из уравнения (3) можно получить

Поскольку в МКР сетка имеет сквозную упорядоченную нумерацию узлов (в том же порядке будут располагаться и уравнения СЛАУ), то все ненулевые элементы матрицы коэффициентов [K] (см. выражение (4)), будут расположены на трех соседних диагоналях вдоль главной̆ диагонали.

Решим СЛАУ методом векторной прогонки:

Для дальнейшего изложения перепишем уравнение (4) в виде

Будем искать решение в виде

Понизим индекс на единицу

После подстановки данных выражений в формулу (5) и перегруппировки получим

Сравнивая (6) и (7):

Это выражение (8) представляет собой рекуррентные отношения.

Т. к. на левой границе коэффициент (см. (\*)), получим

На правой границе коэффициент (см. (\*\*)), поэтому из получим

Итак, окончательно алгоритм 3-точечной прогонки в компьютерной программе реализуется в виде двух последовательных циклов:

1. Прямой ход прогонки (в сторону возрастания n)

1.1 Находим по (9)

1.2 В цикле n от 2 до Im-1 находим по (8)

1. Обратный ход прогонки (в сторону убывания n)

2.1 Находим по (10)

2.2 В цикле по n от Im-1 до 1 находим по (∆)

Блок-схема вычислительной программы

**Начало**

Расчет шагов по времени и

координате

Ввод исходных данных Bi, Fomax, Im и др.

Объявление массивов и переменных

Начальные условия

𝜃1 = 1,

𝑖

𝑖 = 1 … 𝐼𝑚

Начало цикла по времени n = 1

Вычисление векторов прогоночных коэфициентов

**А**, **В** и **С** по формулам (7), (8) и (9)



Вычисление вектора D по формуле

Обращение к процедуре прогонки:

вычисление 𝜃𝑛+1, 𝑖 = 1 … 𝐼𝑚

𝑖

Новое значение времени

*Fo = Fo +* *Fo*

Сохранение значений температуры, требуемых

для построения графиков

Fo  Fomax

Да

Нет

𝜃𝑛 = 𝜃𝑛+1, 𝑖 = 1 … 𝐼𝑚 ;

𝑖 𝑖

*n = n +* 1

**Конец**

# Тестовый расчет

# Для значений X=0, Fo=6, Bi=0.45 и Im=81, урезанная программа для тестирования (в ней осуществляется лишь расчет θ при заданных значениях X, Fo, Bi, Im) выдает значение θ=0.102; для X=0, Fo=3, Bi=0.5, Im=81: θ=0.297; для X=0, Fo=4, Bi=1.6, Im=81: θ=0.019; данные значения θ соответствуют значениям, полученным с помощью урезанной программы для нахождения θ методом Фурье (см. Приложение 2 и 6). Отсюда делаем вывод, что программа работает исправно.

# Результат решения задачи

# На рисунках 1 и 2 изображены зависимости температуры от времени при Im = 101 для трех разных сечений пластины (х=0; х=δ/4; х=δ/2) при Bi=0.0955 и Bi=68.2. На рисунках видно, что с увеличением значения параметра Bi уменьшается время, необходимое для достижения теплового равновесия. График зависимости температуры от времени для x=δ/2 испытывает резкий спад, а для х=δ/4 и х=0 графики асимптотически приближаемся к температуре, соответствующей состоянию теплового равновесия.

# На рисунках 3, 4 и 5 изображены зависимости температуры от координаты при Im = 101 для трех разных временных точек (t=0.1⋅Fomax; t=0.5⋅Fomax; t=0.9⋅Fomax) при Bi=0.0955, Bi=1.091 и Bi=68.2. На рисунках видно, что с увеличением значения параметра Bi разница температуры между центром пластины и ее краем возрастает, и при приближении к точке времени Fomax температура в точках пластины приближается к значениям температуры в момент теплового равновесия.

# A picture containing chart Description automatically generatedHistogram Description automatically generated

Рисунок 1 Рисунок 2

Зависимость температуры от Зависимость температуры от

времени при Bi=0.095 времени при Bi=68.2

Im=101 Im=101

**Chart, line chart

Description automatically generated**Chart, line chart

Description automatically generated

Рисунок 3 Рисунок 4

Зависимость температуры от Зависимость температуры от

координаты при Bi=0.095 координаты при Bi=1.09

Fomax = 55752 с Fomax = 7750 с

Im=101 Im=101

Chart, line chart

Description automatically generated

# Рисунок 5

Зависимость температуры от

координаты при Bi=68.2

Fomax = 2941 с

Im=101

# Выводы

# В ходе решения поставленной задачи был изучен метод конечных разностей. Решение задачи данным методом было реализовано на языке программирования Fortran (см. Приложение 1), результаты программы представлены в виде графиков с использованием пакета Matlab.

# Для выявления влияния величины шага по времени на точность решения были проведены расчеты программой для нахождения аналитического решения методом Фурье (см. Приложение 3) при Bi=0.095, Fo=5 и X=0 (см. Приложение 4), а затем расчеты с помощью основной программы (см. Приложение 1) при Im=21, Bi=1.09, Fo=1.5, X=0 для ∆Fo=dF, ∆Fo=5dF, ∆Fo=20dF, ∆Fo=100dF, где dF – максимально допустимый шаг для явной схемы на той же сетке (см. Приложение 5). Получились следующие результаты (таблица 1):

# Таблица 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ∆Fo | Fo | θBi=1.09 | ΔθBi=1.09 | δBi=1.09, % |
| dF | 1.49999999999999999999999999999997708 | 0.3455782 | -0.0000155 | -0.0044883 |
| 5dF | 1.50000000000000000000000000000000404 | 0.3463809 | 0.0007871 | 0.2277771 |
| 20dF | 1.49999999999999999999999999999999904 | 0.3493703 | 0.0037765 | 1.0927731 |
| 100dF | 1.50000000000000000000000000000000019 | 0.3647810 | 0.0191872 | 5.5519619 |
| Фурье | 1.5 | 0.3455937 | - | - |

# Из таблицы видно, что с увеличением значения dF растет и относительная погрешность.

# Для сравнения явного и неявного методов были выявлены влияния величины шагов по координате и времени на точность результатов, для чего была использована программа для нахождения аналитического решения методом Фурье (см. Приложение 3). Были выбраны значения координаты X=0 и времени Fo=1(см. Приложение 4). Явный метод для различных Im выдал следующие результаты (таблица 2):

# Таблица 2

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Im | Fo | θBi=0.095 | θBi=1.09 | θBi=68.2 | ΔθBi=0.095 | ΔθBi=1.09 | ΔθBi=68.2 | δBi=0.095, % | δBi=1.09, % | δBi=68.2, % |
| 11 | 1.00000000000000000000000000000000058 | 0.915731 | 0.470984 | 0.090433 | 0.010282 | 0.041391 | 0.025434 | -1.1103 | -8.078 | -21.951 |
| 21 | 0.999999999999999999999999999999986326 | 0.920733 | 0.491324 | 0.102635 | 0.005279 | 0.021051 | 0.013233 | -0.5701 | -4.1087 | -11.421 |
| 41 | 1.00000000000000000000000000000005970 | 0.923214 | 0.501677 | 0.109115 | 0.002799 | 0.010698 | 0.006752 | -0.30227 | -2.0879 | -5.8272 |
| 81 | 1.00000000000000000000000000000019356 | 0.924449 | 0.506904 | 0.112458 | 0.0015634 | 0.005461 | 0.003409 | -0.16883 | -1.06773 | -2.9429 |
| Фурье | 1.0 | 0.92601 | 0.51237 | 0.11586 | - | - | - | - | - | - |

# Погрешность менее 6% для всех трех Bi достигается при Im=41, что является приемлемой точностью. Если посмотреть на рисунки 1 и 2, то можно понять, с чем связана то, что погрешности для Bi=0.095 получились самыми маленькими – на этих рисунках фиолетовыми чертами отмечены целые числа Fo. Для Bi=0.095 изменения значений температур для всех трех сечений пластины менее 50 К, а для Bi=68.2 от 300 до 350. Т. е. с увеличением параметра Bi растет и скорость изменения температуры, с чем и может быть связана самая маленькая погрешность для Bi=0.095.

# Теперь же рассмотрим результаты неявного метода (таблица 3):

# Таблица 3

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Im | Fo | θBi=0.095 | θBi=1.09 | θBi=68.2 | ΔθBi=0.095 | ΔθBi=1.09 | ΔθBi=68.2 | δBi=0.095, % | δBi=1.09, % | δBi=68.2, % |
| 11 | 1.00000000000000000000000000000000058 | 0.925764 | 0.513042 | 0.117835 | -0.0002482 | 0.00066638 | 0.0019671 | 0.0268 | 0.13006 | 1.6977 |
| 21 | 0.999999999999999999999999999999986326 | 0.925702 | 0.512386 | 0.116361 | -0.0003099 | 0.0000111 | 0.00049404 | 0.0335 | 0.00216 | 0.4264 |
| 41 | 1.00000000000000000000000000000005970 | 0.925687 | 0.512222 | 0.115993 | -0.0003253 | -0.0001528 | 0.0001251 | 0.03514 | 0.02983 | 0.10801 |
| 81 | 1.00000000000000000000000000000019356 | 0.925683 | 0.512182 | 0.115901 | -0.0003292 | -0.0001938 | 0.00003288 | 0.03555 | 0.0378 | 0.0283 |
| Фурье | 1.0 | 0.92601 | 0.51237 | 0.11586 | - | - | - | - | - | - |

# Во всех случаях погрешность менее 2%, что является хорошей точностью. Выявить какую-либо зависимость погрешности от значений Bi или Im невозможно, т. к., например, для Bi=0.095 погрешность с увеличением Im растет, а для Bi=68.2 падает, хотя для Bi=1.09 сначала резко падает, затем увеличивается. Так же нельзя сказать, для какого Bi погрешность всегда наименьшая, потому что при Im=11 это Bi=0.095, а при Im=41 это уже Bi=68.2. Однако можно сказать, что минимальная погрешность достигается при Bi=68.2 и Im=81, а максимальная при Bi=68.2 и Im=11. Отсюда вывод: неявный метод оказывается более точным, но также тяжело предсказать как параметры Im и Bi повлияют на значение погрешности.

# Приложение

# Основная программа

module methods

contains

character(len=20) function str(k) !function for translating integer to string

integer, intent(in) :: k

write (str, \*) k

str = adjustl(str)

end function

subroutine sweep(*Im*,*A*,*B*,*C*,*D*,*F*)

implicit none

integer, intent(in):: Im

real\*16, dimension(1:Im), intent(in):: A, B, C, D

real\*16, dimension(1:Im), intent(out):: F

real\*16, dimension(1:Im):: alpha, beta

real\*16 :: k0

integer::i

!Прямой ход

alpha(1) = -A(1) / B(1)

beta(1) = -D(1) / B(1)

do i = 2, (Im-1)

k0 = (B(i) + C(i)\*alpha(i-1))

alpha(i) = -A(i) / k0

beta(i) = -(D(i) + C(i)\*beta(i-1)) / k0

end do

!Обратный ход

F(Im) = -(D(Im) + C(Im)\*beta(Im-1)) / (B(Im) + C(Im)\*alpha(Im-1))

do i = (Im-1), 1, -1

F(i) = alpha(i)\*F(i+1) + beta(i)

end do

end subroutine

subroutine finit\_differences(*p*, *Te*, *T0*, *delta*, *alpha*, *lambda*, *cp*, *po*, *check*,*t\_res*)

implicit none

real\*16 Te, T0, delta, lambda, cp, po, Bi, Fo, Fo\_max, st, delt\_X, delt\_Fo, one,delt\_Fo\_2,t\_res

real\*4 Bi\_wr

integer :: i, j, n, alpha, p, p\_Fo, Fo\_points, popul, check, t\_1

real\*16, allocatable, dimension (:,:) ::tau\_x

real\*16, allocatable, dimension (:) :: points, AA,CC,BB,DD, O\_n,T

integer, allocatable, dimension (:) :: points\_dot

Bi=(alpha\*delta)/(2\*lambda)

write(\*,\*)Bi

if (Bi<=0.5) then !Fo\_max based on Bi

write(13,'(F5.3,$)') Bi

write(\*,'(F5.3)') Bi

elseif (Bi>=10) then

write(13,'(F4.1,$)') Bi

write(\*,'(F4.1)') Bi

else

write(13,'(F4.2,$)') Bi

write(\*,'(F4.2)') Bi

end if

if (Bi<1.25) then

Fo\_max=3.11\*Bi\*\*(-0.81)

elseif (Bi>20) then

Fo\_max=1.1

else

Fo\_max=2.76\*Bi\*\*(-0.31)

end if

write(\*,'(A9,$)')'Fo\_max= '

write(\*,'(F12.6)')(Fo\_max\*cp\*po\*(delta\*\*2))/(4\*lambda)

allocate(points(6))

allocate(points\_dot(6))

points(1)=0

points(2)=0.5 !delta/4

points(3)=1 !delta/2

points(4)=Fo\_max\*0.1 !(Fo\_max\*0.1\*cp\*po\*(delta\*\*2))/(4\*lambda)

points(5)=Fo\_max\*0.5 !(Fo\_max\*0.5\*cp\*po\*(delta\*\*2))/(4\*lambda)

points(6)=Fo\_max\*0.9 !(Fo\_max\*0.9\*cp\*po\*(delta\*\*2))/(4\*lambda)

one=1

delt\_X=one/(p-1)

delt\_Fo\_2=(delt\_X\*\*2)/2

delt\_Fo=delt\_Fo\_2

if (check==1) delt\_Fo=delt\_Fo\_2\*1

if (check==2) delt\_Fo=delt\_Fo\_2\*5

if (check==3) delt\_Fo=delt\_Fo\_2\*20

if (check==4) delt\_Fo=delt\_Fo\_2\*100

p\_Fo=int(Fo\_max/delt\_Fo)

write(13,\*) p, p\_Fo, alpha

allocate(O\_n(p+1))

allocate(T(p+1))

allocate(tau\_x(p\_Fo+1,2))

allocate(AA(p),BB(p),CC(p),DD(p))

do i=2,p-1

CC(i)=(1.0)/(delt\_X\*\*2)

AA(i)=CC(i)

BB(i) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2))

end do

CC(1)=0

AA(1)=(2.0)/(delt\_X\*\*2)

BB(1) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2))

CC(p)=(2.0)/(delt\_X\*\*2)

AA(p)=0

BB(p) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2)) - ((2.0\*Bi)/(delt\_X))

st=0

popul=0

do i=1,p

popul=popul+1

O\_n(i)=1

T(i)=T0

tau\_x(i,2)=(st\*delta)/2.0

st=st+delt\_X

do j=1,3

if ((points(j)>st-delt\_x) .and. (points(j)<st+delt\_x)) then

points\_dot(j)=popul

end if

end do

end do

tau\_x(1,1)=0

if (check==0) then

open(10, file='bin/res/Im='//trim(str(p))//'/alpha'//trim(str(alpha))//'\_temp\_t.txt')

open(101, file='bin/res/Im='//trim(str(p))//'/alpha'//trim(str(alpha))//'\_temp\_x.txt')

open(11, file='bin/res/Im='//trim(str(p))//'/alpha'//trim(str(alpha))//'\_time.txt')

open(21, file='bin/res/Im='//trim(str(p))//'/alpha'//trim(str(alpha))//'\_coord.txt')

else

open(10, file='bin/res/Im='//trim(str(p))//'\_def/alpha'//trim(str(alpha))//'\_dFo'//trim(str(check))//'\_temp\_t.txt')

open(101, file='bin/res/Im='//trim(str(p))//'\_def/alpha'//trim(str(alpha))//'\_dFo'//trim(str(check))//'\_temp\_x.txt')

open(11, file='bin/res/Im='//trim(str(p))//'\_def/alpha'//trim(str(alpha))//'\_dFo'//trim(str(check))//'\_time.txt')

open(21, file='bin/res/Im='//trim(str(p))//'\_def/alpha'//trim(str(alpha))//'\_dFo'//trim(str(check))//'\_coord.txt')

end if

Fo=delt\_Fo

n=0

tau\_x(1,1)=0

do n=1,p\_Fo+1 !time

do i=4,6

if ((points(i)>Fo-delt\_fo) .and. (points(i)<Fo+delt\_fo)) then

points\_dot(i)=n

end if

end do

do i=1,p

DD(i)=O\_n(i)/delt\_Fo

end do

call sweep(p,AA,BB,CC,DD,O\_n)

do i=1,p

T(i)=O\_n(i)\*(T0-Te)+Te

end do

tau\_x(n+1,1)=(Fo\*cp\*po\*(delta\*\*2))/(4\*lambda) !real time

write(10,\*) T(1), T(points\_dot(2)), T(points\_dot(3))

write(10,\*)

do i=4,6

if ((points(i)>Fo-delt\_fo/2) .and. (points(i)<Fo+delt\_fo/2)) then

do j=1,p

write(101,'(f18.10,$)') T(j)

end do

write(101,\*)

end if

end do

if ((abs(Fo-1.0)<=(delt\_Fo/2.0)) .and. (check==0)) then

t\_1=n

write(\*,'(A10,$)')'For iter '

write(\*,'(i5,$)')t\_1

write(\*,'(A11,$)')' and time '

write(\*,\*) Fo, ': T(0,Fo=1)=', O\_n(1), (O\_n(1)-t\_res), ((O\_n(1)-t\_res)/t\_res)\*100

elseif ((abs(Fo-1.5)<=(delt\_Fo/2.0)) .and. (check/=0)) then

t\_1=n

write(\*,'(A10,$)')'For iter '

write(\*,'(i5,$)')t\_1

write(\*,'(A11,$)')' and time '

write(\*,\*) Fo, ': T(0,Fo=1.5)=', O\_n(1), (O\_n(1)-0.345593785639), ((O\_n(1)-0.345593785639)/0.345593785639)\*100

end if

Fo=Fo+delt\_Fo

if (Fo>=Fo\_max) exit

end do

do i=1,p\_Fo

write(11,\*) tau\_x(i,1)

end do

do i=1,p

write(21,\*) tau\_x(i,2)

end do

write(20,\*) 1, points\_dot(2), points\_dot(3), points\_dot(4), points\_dot(5), points\_dot(6)

deallocate(points)

deallocate(points\_dot)

close(10)

close(11)

close(101)

deallocate(O\_n)

deallocate(T)

deallocate(tau\_x)

deallocate(AA,BB,CC,DD)

end subroutine

end module

program main

use methods

implicit none

real\*16 Te, T0, delta, lambda, cp, po, E, t\_res,T\_res\_n(3)

integer, allocatable, dimension (:) :: alpha\_n, p\_n, tau\_points

integer :: n, i, alpha, p, j, check, Fo\_points

Te=130.0+273.15

T0=500.0+273.15

delta=0.6

lambda=110.0

cp=380.0

po=8600.0

allocate(p\_n(5))! grid size

allocate(alpha\_n(3))

allocate(tau\_points(100))

alpha\_n(1)=35

alpha\_n(2)=400

alpha\_n(3)=25000

p\_n(1)=11

p\_n(2)=21

p\_n(3)=41

p\_n(4)=81

p\_n(5)=101

T\_res\_n(1)= 0.926012747427353802229742345141718148

T\_res\_n(2)= 0.512375769314193418196748998307030519

T\_res\_n(3)= 0.115867936825182547473269331561302031

write(\*,\*)'start program'

write(\*,\*) 'Main + Research 2'

check=0

do i=1,5

p=p\_n(i)

write(\*,\*) '\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_'

write(\*,'(a4,$)') ' Im='

write(\*,'(i3)') p

write(\*,\*)

open(13, file='bin/res/Im='//trim(str(p))//'/bi\_iFORcoord\_iFORtime\_alpha.txt')

open(20, file='bin/res/Im='//trim(str(p))//'/points.txt')

do j=1,3

alpha=alpha\_n(j)

t\_res=T\_res\_n(j)

write(\*,'(A4,$)') ' Bi='

call finit\_differences(p, Te, T0, delta, alpha, lambda, cp, po,check,t\_res)

write(\*,\*)

end do

close(20)

close(13)

write(\*,\*) 'this Im done'

end do

write(\*,\*)

write(\*,\*) 'Research 1'

alpha=alpha\_n(2)

do check=1,4

p=21

write(\*,\*) '\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_'

write(\*,'(a4,$)') ' Im='

write(\*,'(i3)') p

write(\*,\*)

write(\*,'(a12,$)') ' num of dFo:'

write(\*,'(i3)') check

write(\*,\*)

open(13, file='bin/res/Im='//trim(str(p))//'\_def/bi\_iFORcoord\_iFORtime\_alpha.txt')

open(20, file='bin/res/Im='//trim(str(p))//'\_def/points.txt')

write(\*,'(A4,$)') ' Bi='

call finit\_differences(p, Te, T0, delta, alpha, lambda, cp, po,check,t\_res)

write(\*,\*)

close(20)

close(13)

end do

write(\*,\*) 'this res done'

open(29, file='bin/res/fo\_points.txt')

do Fo\_points=1,100

tau\_points(Fo\_points)=(Fo\_points\*cp\*po\*(delta\*\*2))/(4\*lambda)

write(29,\*) Fo\_points, tau\_points(Fo\_points), 400

end do

close(29)

write(\*,\*) 'end program'

end program

# Урезанная программа для тестирования

module methods\_test

contains

character(len=20) function str(k) !function for translating integer to string

integer, intent(in) :: k

write (str, \*) k

str = adjustl(str)

end function

subroutine sweep(*Im*,*A*,*B*,*C*,*D*,*F*)

implicit none

integer, intent(in):: Im

real\*16, dimension(1:Im), intent(in):: A, B, C, D

real\*16, dimension(1:Im), intent(out):: F

real\*16, dimension(1:Im):: alpha, beta

real\*16 :: k0

integer::i

!Прямой ход

alpha(1) = -A(1) / B(1)

beta(1) = -D(1) / B(1)

do i = 2, (Im-1)

k0 = (B(i) + C(i)\*alpha(i-1))

alpha(i) = -A(i) / k0

beta(i) = -(D(i) + C(i)\*beta(i-1)) / k0

end do

!Обратный ход

F(Im) = -(D(Im) + C(Im)\*beta(Im-1)) / (B(Im) + C(Im)\*alpha(Im-1))

do i = (Im-1), 1, -1

F(i) = alpha(i)\*F(i+1) + beta(i)

end do

end subroutine

subroutine finit\_differences(*X*, *Fo\_final*, *Bi*, *p*)

implicit none

real\*16 Bi, Fo, Fo\_final, delt\_X, delt\_Fo, one, X

integer :: i, j, n,p, x\_i, Fo\_i

real\*16, allocatable, dimension (:) :: AA,CC,BB,DD, O\_n

one=1

delt\_X=one/(p-1)

delt\_Fo=(delt\_X\*\*2)/2

allocate(O\_n(p+1))

do i=1,p

O\_n(i)=1

end do

allocate(AA(p),BB(p),CC(p),DD(p))

do i=2,p-1

CC(i)=(1.0)/(delt\_X\*\*2)

AA(i)=CC(i)

BB(i) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2))

end do

CC(1)=0

AA(1)=(2.0)/(delt\_X\*\*2)

BB(1) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2))

CC(p)=(2.0)/(delt\_X\*\*2)

AA(p)=0

BB(p) = - ((1.0)/(delt\_Fo)) - ((2.0)/(delt\_X\*\*2)) - ((2.0\*Bi)/(delt\_X))

Fo=delt\_Fo

Fo\_i=0

do n=1,100000 !time

do i=1,p

DD(i)=O\_n(i)/delt\_Fo

end do

call sweep(p,AA,BB,CC,DD,O\_n)

Fo=Fo+delt\_Fo

if (Fo>Fo\_final) exit

end do

if (X==0) then

x\_i=1

elseif (X==1) then

x\_i=p

end if

write(\*,'(A13,$)')'Your theta = '

write(\*,\*) O\_n(x\_i)

end subroutine

end module

program main

use methods\_test

implicit none

real\*16 X, Fo, Bi

integer p

X=0

p=81

Fo=4.0

Bi=1.6

call finit\_differences(X, Fo, Bi, p)

end program

# Программа для нахождения значений аналитического решения

module functions\_temp\_find

contains

real\*16 Function f(Mu, Bi) result(res)

real\*16 :: Mu, Bi

res=(cos(Mu))/(sin(Mu))-(Mu/Bi)

end function

real\*16 Function A\_f(Mu) result(res)

real\*16 :: Mu

res=(2\*sin(Mu))/(Mu+sin(Mu)\*cos(Mu))

end function

real\*16 Function O\_f(Mu, A, Fo, X) result(res)

real\*16 :: Mu, A, Fo, X

res=A\*exp( (-1)\*Fo\*(Mu\*\*2))\*cos(Mu\*X)

end function

end module

module methods\_temp\_find

contains

real\*16 Function bis(n, E, Bi) result(res)

use functions\_temp\_find

real\*16 :: E, a\_Mu, b\_Mu, c\_Mu, f1, f2, check, pi=3.1415926535897932, Bi, delt, n\_r

integer :: n, i

n\_r=real(n, 16)

a\_Mu=pi\*(n\_r-1)

b\_Mu=pi\*(n\_r-0.5)

f1=f(b\_Mu, Bi)

do i=1,1000

c\_Mu=(a\_Mu+b\_Mu)/2.0

f2=f(c\_Mu, Bi)

check=f1\*f2

if (check>0) then

b\_Mu=c\_Mu

f1=f2

elseif (check<0) then

a\_Mu=c\_Mu

elseif (check==0) then

a\_Mu=c\_Mu

b\_Mu=c\_Mu

end if

delt=abs(a\_Mu-b\_Mu)

if (delt<E) exit

end do

res=c\_Mu

end function

subroutine Fourier(*X*, *Fo*, *Bi*, *E*)

use functions\_temp\_find

implicit none

real\*16 O, Bi, Mu, A, Fo, X, E, O\_n, check,te,t0

integer :: i, n

n=0

O=0

O\_n=0

check=1

do i=1,1000

n=n+1

Mu=bis(n, E, Bi)

A=A\_f(Mu)

O\_n=O\_f(Mu, A, Fo, X)

O=O+O\_n

check=abs(O)

if (check<E) exit

end do

Te=130.0+273.15

T0=500.0+273.15

write(\*,\*)'T=', O

end subroutine

end module

program main

use functions\_temp\_find

use methods\_temp\_find

implicit none

real\*16 X, Fo, Bi, E

write(\*,\*)'Fo=1'

X=0

Fo=1.0

Bi=0.095

call Fourier(X, Fo, Bi, E)

Bi=1.09

call Fourier(X, Fo, Bi, E)

Bi=68.2

call Fourier(X, Fo, Bi, E)

write(\*,\*)'Fo=1.5'

X=0

Fo=1.5

Bi=0.095

call Fourier(X, Fo, Bi, E)

Bi=1.09

call Fourier(X, Fo, Bi, E)

Bi=68.2

call Fourier(X, Fo, Bi, E)

end program

# Вывод программы для нахождения значений аналитического решения

Fo=1

T= 0.926012747427353802229742345141718148

T= 0.512375769314193418196748998307030519

T= 0.115867936825182547473269331561302031

Fo=1.5

T= 0.884352382654567618325153560153664267

T= 0.345593785639343570197473485001033289

T= 3.49578492264228096670207494924388298E-0002

# Вывод основной программы

start program

Main + Research 2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 11

Bi= 9.54545492475683038884943181818181774E-0002

0.095

Fo\_max= 55751.558487

For iter 200 and time 1.00000000000000000000000000000000058 : T(0,Fo=1)= 0.925764528538175588010240507104157348 -2.48225902132029177259492895842652453E-0004 -2.68058837139948106386826522750102553E-0002

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 200 and time 1.00000000000000000000000000000000058 : T(0,Fo=1)= 0.513042156576282358060017501707569670 6.66384576923227200642501707569669933E-0004 0.130057784411410596683672823990412390

Bi= 68.1818208911202170632102272727272671

68.2

Fo\_max= 2941.200297

For iter 200 and time 1.00000000000000000000000000000000058 : T(0,Fo=1)= 0.117835061746644106369247362719973707 1.96712662558468864463798771997370717E-0003 1.69773166625384717087153730566316071

this Im done

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 21

Bi= 9.54545492475683038884943181818181774E-0002

0.095

Fo\_max= 55751.558487

For iter 800 and time 0.999999999999999999999999999999986326 : T(0,Fo=1)= 0.925702801711941502528604929568917351 -3.09952728366114658895070431082648825E-0004 -3.34717558564788372507222487395446335E-0002

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 800 and time 0.999999999999999999999999999999986326 : T(0,Fo=1)= 0.512386872355251442036927530643766552 1.11003558923111775525306437665520409E-0005 2.16644823954030786036960997816018307E-0003

Bi= 68.1818208911202170632102272727272671

68.2

Fo\_max= 2941.200297

For iter 800 and time 0.999999999999999999999999999999986326 : T(0,Fo=1)= 0.116361976279990501941767736609164783 4.94041158931084217158361609164782530E-0004 0.426382983708916068801547007715967603

this Im done

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 41

Bi= 9.54545492475683038884943181818181774E-0002

0.095

Fo\_max= 55751.558487

For iter 3200 and time 1.00000000000000000000000000000005970 : T(0,Fo=1)= 0.925687367184326494056602649740878269 -3.25387255981123130897350259121731268E-0004 -3.51385285376323783525935588538278387E-0002

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 3200 and time 1.00000000000000000000000000000005970 : T(0,Fo=1)= 0.512222911206710381084042315902740137 -1.52860792648749775332684097259862730E-0004 -2.98337277057942096620151319605437336E-0002

Bi= 68.1818208911202170632102272727272671

68.2

Fo\_max= 2941.200297

For iter 3200 and time 1.00000000000000000000000000000005970 : T(0,Fo=1)= 0.115993080218246844452239515575667228 1.25145097187426727630140575667228215E-0004 0.108006669020790336006096714122020877

this Im done

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 81

Bi= 9.54545492475683038884943181818181774E-0002

0.095

Fo\_max= 55751.558487

For iter 12800 and time 1.00000000000000000000000000000019356 : T(0,Fo=1)= 0.925683508401558838586680861283510178 -3.29246038748778600819138716489822209E-0004 -3.55552379996945678782021425741006162E-0002

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 12800 and time 1.00000000000000000000000000000019356 : T(0,Fo=1)= 0.512181912214759217249954010117763898 -1.93859784599913609420989882236101698E-0004 -3.78354706046007353517693282851912496E-0002

Bi= 68.1818208911202170632102272727272671

68.2

Fo\_max= 2941.200297

For iter 12800 and time 1.00000000000000000000000000000019356 : T(0,Fo=1)= 0.115900817647546991555443152946039561 3.28825264875738308337779460395611273E-0005 2.83793151687893610395402328808430012E-0002

this Im done

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im=101

Bi= 9.54545492475683038884943181818181774E-0002

0.095

Fo\_max= 55751.558487

For iter 20000 and time 0.999999999999999999999999999999981896 : T(0,Fo=1)= 0.925683045343734516207292436041747671 -3.29709096573100980207563958252328688E-0004 -3.56052435554606174155036971197211723E-0002

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 20000 and time 0.999999999999999999999999999999981896 : T(0,Fo=1)= 0.512176992102052091003938828626132748 -1.98779897307039855436171373867252270E-0004 -3.87957253582412724362528932988909051E-0002

Bi= 68.1818208911202170632102272727272671

68.2

Fo\_max= 2941.200297

For iter 20000 and time 0.999999999999999999999999999999981896 : T(0,Fo=1)= 0.115889745105927663187857673125801976 2.18099848682454632482981258019760512E-0005 1.88231410574964318528772139013155884E-0002

this Im done

Research 1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 21

num of dFo: 1

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 1200 and time 1.49999999999999999999999999999997708 : T(0,Fo=1.5)= 0.345578268856964442860744602311120040 -1.55114221951030376928976888799599623E-0005 -4.48833951310507078273614548210110387E-0003

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 21

num of dFo: 2

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 240 and time 1.50000000000000000000000000000000404 : T(0,Fo=1.5)= 0.346380964016712253581520972519537350 7.87183737552707683083472519537349574E-0004 0.227777171486375120959006400463019391

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 21

num of dFo: 3

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 60 and time 1.49999999999999999999999999999999904 : T(0,Fo=1.5)= 0.349370336476047863430545261224651378 3.77655619688831753210776122465137829E-0003 1.09277319569748530812644182483526204

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Im= 21

num of dFo: 4

Bi= 1.09090913425792347301136363636363645

1.09

Fo\_max= 7749.675765

For iter 12 and time 1.50000000000000000000000000000000019 : T(0,Fo=1.5)= 0.364781015423486140680957569339846389 1.91872351443265947825200693398463887E-0002 5.55196193890635504071138160740339321

this res done

end program

# Вывод урезанной программы для метода Фурье

Input X: 0

Input Fo: 6

Input Bi: 0.45

O= 0.102553513049214762797759192916365133

Input X: 0

Input Fo: 3

Input Bi: 0.5

O= 0.297449345355111229166080552273732902

Input X: 0

Input Fo: 4

Input Bi: 1.6

O= 1.98441311715528375238824378099299295E-0002