

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА
ВЕЛИКОГО
Институт компьютерных наук и кибербезопасности
Высшая школа программной инженерии

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине: «Вычислительная математика»
Вариант №27

Выполнила
студентка гр. в5130904/30022

Г.М.Феллер

Преподаватель

С.П.Воскобойников

« _____ » _____ 2025 г.

Санкт-Петербург
2025

Задание

Написать процедуру формирования матрицы A по заданному вектору B

$$pA = \begin{pmatrix} 1 & a_1 & a_1 & \dots & a_1 \\ 1 & 1 & a_2 & \dots & a_2 \\ \dots & \dots & \dots & \dots & \dots \\ 1 & 1 & 1 & \dots & a_{n-1} \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix}, B = (a_1 \ a_2 \ \dots \ a_{n-1})^T$$

Задавая $n = 5, a_1 = 4, a_2 = 3, a_3 = 2, a_4 = var = 1.5; 1.01; 1.001; 1.0001$ и вычисляя A^{-1} с помощью DECOMP и SOLVE, найти нормы матриц $R = AA^{-1} - E$ для всех вариантов a_4 .

Код программы

```
1  program main
2  use matrix_ops
3  implicit none
4  ! Определение размеров матрицы
5  integer, parameter :: n = 5, ndim = n
6  real :: A(ndim, n), A_inv(ndim, n), B(n-1), cond, work(n)
7  integer :: ipvt(n), i, j
8  real :: R(ndim, n), Identity(ndim, n), normR
9  real, dimension(4) :: var_values = [1.5, 1.01, 1.001, 1.0001]
10 integer :: k
11
12 ! Задание вектора B
13 B = [4.0, 3.0, 2.0, 0.0] ! Последний элемент заменяется на var позже
14
15 do k = 1, 4
16     B(4) = var_values(k) ! Устанавливаем текущее значение var
17     print *, "-----"
18     print *, "For var =", B(4)
19
20     ! Формирование матрицы A
21     do i = 1, n
22         do j = 1, n
23             if (j < i) then
24                 ! Элементы ниже главной диагонали равны 1
25                 A(i, j) = 1.0
26             else if (j > i) then
27                 ! Элементы выше главной диагонали равны B
28                 A(i, j) = B(j-1)
29             else
30                 A(i, j) = 1.0
31             end if
32         end do
33     end do
34
35     print *, "Matrix A:"
36     do i = 1, n
37         print *, (A(i, j), j=1, n)
38     end do
39
40     ! Копируем A в A_inv для получения A^(-1)
41     A_inv = A
42
43     ! Разложение A и нахождение A^(-1)
44     call decomp(ndim, n, A_inv, cond, ipvt, work)
```

```

45     do i = 1, n
46         work = 0.0
47         work(i) = 1.0
48         call solve(ndim, n, A_inv, work, ipvt)
49         ! Формируем столбцы обратной матрицы
50         A_inv(:, i) = work
51     end do
52
53     print *, "Inverse Matrix A_inv:"
54     do i = 1, n
55         print *, (A_inv(i, j), j=1, n)
56     end do
57
58     ! Вычисление  $R = AA^{(-1)} - E$ 
59     ! Умножаем A на  $A^{(-1)}$ 
60     R = matmul(A, A_inv)
61     ! Формирование единичной матрицы I
62     Identity = 0.0
63     do i = 1, n
64         Identity(i, i) = 1.0
65     end do
66     ! Вычисляем отклонение от единичной матрицы
67     R = R - Identity
68
69     print *, "Matrix R =  $AA^{(-1)} - I$ :"
70     do i = 1, n
71         print *, (R(i, j), j=1, n)
72     end do
73
74     ! Вычисление нормы матрицы R
75     normR = 0.0
76     do i = 1, n
77         do j = 1, n
78             ! Суммируем абсолютные значения элементов
79             normR = normR + ABS(R(i, j))
80         end do
81     end do
82
83     ! Вывод результатов
84     print *, "Norm of R:", normR
85 end do
86
87 end program main

```

Выполнение программы

```

-----
For var = 1.50000000
Matrix A:
  1.00000000    4.00000000    3.00000000    2.00000000    1.50000000
  1.00000000    1.00000000    3.00000000    2.00000000    1.50000000
  1.00000000    1.00000000    1.00000000    2.00000000    1.50000000
  1.00000000    1.00000000    1.00000000    1.00000000    1.50000000
  1.00000000    1.00000000    1.00000000    1.00000000    1.00000000
Inverse Matrix A_inv:
-0.333333373    0.499999821    1.49999988    2.99999976    -8.99999905
 0.333333343   -0.333333343   -0.00000000   -0.00000000   -0.00000000
-0.00000000    0.500000000   -0.50000000   -0.00000000   -0.00000000
-0.00000000   -0.00000000    1.00000000   -1.00000000   -0.00000000
-0.00000000   -0.00000000   -0.00000000    2.00000000   -2.00000000
Matrix R = AA^(-1) - I:
 0.00000000    0.666666448    1.99999988    3.99999976   -11.9999990
-2.98023224E-08 0.666666508    1.99999988    3.99999976   -11.9999990
-2.98023224E-08 0.666666508    2.00000000    3.99999976   -11.9999990
-2.98023224E-08 0.666666508    1.99999988    4.00000000   -11.9999990
-2.98023224E-08 0.666666508    1.99999988    3.99999976   -11.9999990
Norm of R: 93.3333282
-----

For var = 1.00999999
Matrix A:
  1.00000000    4.00000000    3.00000000    2.00000000    1.00999999
  1.00000000    1.00000000    3.00000000    2.00000000    1.00999999
  1.00000000    1.00000000    1.00000000    2.00000000    1.00999999
  1.00000000    1.00000000    1.00000000    1.00000000    1.00999999
  1.00000000    1.00000000    1.00000000    1.00000000    1.00000000
Inverse Matrix A_inv:
-0.333333373    0.499999821    1.49999988    297.000275   -303.000275
 0.333333343   -0.333333343   -0.00000000   -0.00000000   -0.00000000
-0.00000000    0.500000000   -0.50000000   -0.00000000   -0.00000000
-0.00000000   -0.00000000    1.00000000   -1.00000000   -0.00000000
-0.00000000   -0.00000000   -0.00000000   100.000099   -100.000099
Matrix R = AA^(-1) - I:
 0.00000000    0.666666448    1.99999988    396.000366   -404.000366
-2.98023224E-08 0.666666508    1.99999988    396.000366   -404.000366
-2.98023224E-08 0.666666508    2.00000000    396.000366   -404.000366
-2.98023224E-08 0.666666508    1.99999988    396.000366   -404.000366
-2.98023224E-08 0.666666508    1.99999988    396.000366   -404.000366
Norm of R: 4013.33740
-----

```

```

-----
For var = 1.00100005
Matrix A:
  1.00000000    4.00000000    3.00000000    2.00000000    1.00100005
  1.00000000    1.00000000    3.00000000    2.00000000    1.00100005
  1.00000000    1.00000000    1.00000000    2.00000000    1.00100005
  1.00000000    1.00000000    1.00000000    1.00000000    1.00100005
  1.00000000    1.00000000    1.00000000    1.00000000    1.00000000
Inverse Matrix A_inv:
-0.333333373    0.499999821    1.49999988    2996.85938    -3002.85938
 0.333333343   -0.333333343   -0.00000000   -0.00000000   -0.00000000
-0.00000000    0.500000000   -0.50000000   -0.00000000   -0.00000000
-0.00000000   -0.00000000    1.00000000   -1.00000000   -0.00000000
-0.00000000   -0.00000000   -0.00000000    999.953247   -999.953247
Matrix R = AA^(-1) - I:
  0.00000000    0.666666448    1.99999988    3995.81250   -4003.81250
-2.98023224E-08  0.666666508    1.99999988    3995.81250   -4003.81250
-2.98023224E-08  0.666666508    2.00000000    3995.81250   -4003.81250
-2.98023224E-08  0.666666508    1.99999988    3995.81250   -4003.81250
-2.98023224E-08  0.666666508    1.99999988    3995.81250   -4003.81250
Norm of R: 40011.4570

```

```

-----
For var = 1.00010002
Matrix A:
  1.00000000    4.00000000    3.00000000    2.00000000    1.00010002
  1.00000000    1.00000000    3.00000000    2.00000000    1.00010002
  1.00000000    1.00000000    1.00000000    2.00000000    1.00010002
  1.00000000    1.00000000    1.00000000    1.00000000    1.00010002
  1.00000000    1.00000000    1.00000000    1.00000000    1.00000000
Inverse Matrix A_inv:
-0.333333373    0.499999821    1.49999988    29992.0195   -29998.0195
 0.333333343   -0.333333343   -0.00000000   -0.00000000   -0.00000000
-0.00000000    0.500000000   -0.50000000   -0.00000000   -0.00000000
-0.00000000   -0.00000000    1.00000000   -1.00000000   -0.00000000
-0.00000000   -0.00000000   -0.00000000    9998.34082   -9998.34082
Matrix R = AA^(-1) - I:
  0.00000000    0.666666448    1.99999988    39989.3594   -39997.3594
-2.98023224E-08  0.666666508    1.99999988    39989.3594   -39997.3594
-2.98023224E-08  0.666666508    2.00000000    39989.3594   -39997.3594
-2.98023224E-08  0.666666508    1.99999988    39989.3594   -39997.3594
-2.98023224E-08  0.666666508    1.99999988    39989.3594   -39997.3594
Norm of R: 399946.938

```