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 матиматика»
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

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

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

```
чprogram FormAndInvert
implicit none
integer, parameter :: N = 5
integer :: i, j, k, variant
real :: A(N,N), A_copy(N,N), AINV(N,N), R(N,N)
real :: B(N-1), bvec(N)
real :: cond, work(N)
integer :: ipvt(N)
real :: normR, rowSum
real :: a1, a2, a3, a4
real, dimension(4) :: a4_values

data a4_values / 1.5, 1.01, 1.001, 1.0001 /
a1 = 4.0
a2 = 3.0
a3 = 2.0

do variant = 1, 4

   a4 = a4_values(variant)
   B(1) = a1
   B(2) = a2
   B(3) = a3
   B(4) = a4

   A(1,1) = 1.0
   do j = 2, N
      A(1,j) = a1
   end do

   A(2,1) = 1.0
   A(2,2) = 1.0
   do j = 3, N
      A(2,j) = a2
   end do

   A(3,1) = 1.0
   A(3,2) = 1.0
   A(3,3) = 1.0
   do j = 4, N
      A(3,j) = a3
   end do

   A(4,1) = 1.0
```

```
45          A(4,2) = 1.0
46          A(4,3) = 1.0
47          A(4,4) = 1.0
48          A(4,5) = a4
49
50          do j = 1, N
51             A(5,j) = 1.0
52          end do
53
54          do i = 1, N
55             do j = 1, N
56                A_copy(i,j) = A(i,j)
57             end do
58          end do
59
60          ! Печать матрицы A
61          write(*,'(A, F10.6)') 'For var = ', a4
62          write(*,'(A)') 'Matrix A:'
63          do i = 1, N
64             write(*,'(5F12.4)') (A(i,j), j=1,N)
65          end do
66
67          call DECOMP(N, N, A_copy, cond, ipvt, work)
68
69          ! Вычисляем обратную матрицу AINV
70          do k = 1, N
71             do i = 1, N
72                if (i == k) then
73                   bvec(i) = 1.0
74                else
75                   bvec(i) = 0.0
76                end if
77             end do
78             call SOLVE(N, N, A_copy, bvec, ipvt)
79             do i = 1, N
80                AINV(i,k) = bvec(i)
81             end do
82          end do
83
84          ! Печать обратной матрицы AINV
85          write(*,'(A)') 'Inverse Matrix A_inv:'
86          do i = 1, N
87             write(*,'(5F12.4)') (AINV(i,j), j=1,N)
88          end do
89
90          ! Вычисляем R = A*A_inv - I
91          do i = 1, N
92             do j = 1, N
93                R(i,j) = 0.0
94                do k = 1, N
95                   R(i,j) = R(i,j) + A(i,k) * AINV(k,j)
96                end do
97                if (i == j) then
98                   R(i,j) = R(i,j) - 1.0
99                end if
100            end do
101         end do
102
103         ! Печать матрицы R
104         write(*,'(A)') 'Matrix R = A*A_inv - I:'
105         do i = 1, N
```

```
106           write(*,'(5F12.4)') (R(i,j), j=1,N)
107        end do
108
109        ! Норма матрицы R
110        normR = 0.0
111        do i = 1, N
112           rowSum = 0.0
113           do j = 1, N
114              rowSum = rowSum + abs(R(i,j))
115           end do
116           if (rowSum > normR) then
117              normR = rowSum
118           end if
119        end do
120
121        write(*,'(A, F12.6)') 'Norm of R: ', normR
122        write(*,'(A)') '--------------------------------------------'
123
124     end do
125
126  end program FormAndInvert
127
128  subroutine DECOMP(NDIM, N, A, COND, IPVT, WORK)
129     implicit none
130     integer, intent(in) :: NDIM, N
131     real, intent(inout) :: A(NDIM, N)
132     real, intent(out) :: COND
133     integer, intent(out) :: IPVT(N)
134     real, intent(inout) :: WORK(N)
135     real :: EK, T, ANORM, YNORM, ZNORM
136     integer :: NM1, I, J, K, KP1, KB, M
137
138     IPVT(N) = 1
139     if (N == 1) then
140        COND = 1.0
141        if (A(1,1) /= 0.0) return
142        COND = 1.0e+32
143        return
144     end if
145
146     NM1 = N - 1
147     ANORM = 0.0
148     do J = 1, N
149        T = 0.0
150        do I = 1, N
151           T = T + abs(A(I,J))
152        end do
153        if (T > ANORM) then
154           ANORM = T
155        end if
156     end do
157
158     do K = 1, NM1
159        KP1 = K + 1
160        M = K
161        do I = KP1, N
162           if (abs(A(I,K)) > abs(A(M,K))) then
163              M = I
164           end if
165        end do
166        IPVT(K) = M
```

```
      if (M /= K) then
         IPVT(N) = -IPVT(N)
      end if
      T = A(M,K)
      A(M,K) = A(K,K)
      A(K,K) = T
      if (T == 0.0) cycle
      do I = KP1, N
         A(I,K) = -A(I,K) / T
      end do
      do J = KP1, N
         T = A(M,J)
         A(M,J) = A(K,J)
         A(K,J) = T
         if (T == 0.0) cycle
         do I = KP1, N
            A(I,J) = A(I,J) + A(I,K) * T
         end do
      end do
   end do

   do K = 1, N
      T = 0.0
      if (K /= 1) then
         do I = 1, K-1
            T = T + A(I,K) * WORK(I)
         end do
      end if
      EK = 1.0
      if (T < 0.0) EK = -1.0
      if (A(K,K) == 0.0) then
         COND = 1.0e+32
         return
      end if
      WORK(K) = -(EK + T) / A(K,K)
   end do

   do KB = 1, NM1
      K = N - KB
      T = WORK(K)
      do I = K+1, N
         T = T + A(I,K) * WORK(I)
      end do
      WORK(K) = T
      if (IPVT(K) /= K) then
         T = WORK(IPVT(K))
         WORK(IPVT(K)) = WORK(K)
         WORK(K) = T
      end if
   end do

   YNORM = 0.0
   do I = 1, N
      YNORM = YNORM + abs(WORK(I))
   end do
   call SOLVE(NDIM, N, A, WORK, IPVT)
   ZNORM = 0.0
   do I = 1, N
      ZNORM = ZNORM + abs(WORK(I))
   end do
   COND = ANORM * ZNORM / YNORM
```

```
    if (COND < 1.0) then
       COND = 1.0
    end if
    return
  end subroutine DECOMP

  subroutine SOLVE(NDIM, N, A, B, IPVT)
    implicit none
    integer, intent(in) :: NDIM, N
    integer, intent(in) :: IPVT(N)
    real, intent(inout) :: A(NDIM, N)
    real, intent(inout) :: B(N)
    integer :: KB, NM1, KP1, I, K, M
    real :: T

    if (N == 1) then
       B(1) = B(1) / A(1,1)
       return
    end if

    NM1 = N - 1
    do K = 1, NM1
       KP1 = K + 1
       M = IPVT(K)
       T = B(M)
       B(M) = B(K)
       B(K) = T
       do I = KP1, N
          B(I) = B(I) + A(I,K) * T
       end do
    end do

    do KB = 1, NM1
       K = N - KB + 1
       B(K) = B(K) / A(K,K)
       T = -B(K)
       do I = 1, K-1
          B(I) = B(I) + A(I,K) * T
       end do
    end do
    B(1) = B(1) / A(1,1)
    return
  end subroutine SOLVE
```

Выполнение программы

```
For var =   1.500000
Matrix A:
      1.0000      4.0000      4.0000      4.0000      4.0000
      1.0000      1.0000      3.0000      3.0000      3.0000
      1.0000      1.0000      1.0000      2.0000      2.0000
      1.0000      1.0000      1.0000      1.0000      1.5000
      1.0000      1.0000      1.0000      1.0000      1.0000
Inverse Matrix A_inv:
     -0.3333      0.0000      0.0000      0.0000      1.3333
      0.3333     -0.5000     -0.0000     -0.0000      0.1667
     -0.0000      0.5000     -1.0000     -0.0000      0.5000
     -0.0000     -0.0000      1.0000     -2.0000      1.0000
     -0.0000     -0.0000     -0.0000      2.0000     -2.0000
Matrix R = A*A_inv - I:
      0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
Norm of R:        0.000000
```

```
For var =   1.010000
Matrix A:
      1.0000      4.0000      4.0000      4.0000      4.0000
      1.0000      1.0000      3.0000      3.0000      3.0000
      1.0000      1.0000      1.0000      2.0000      2.0000
      1.0000      1.0000      1.0000      1.0000      1.0100
      1.0000      1.0000      1.0000      1.0000      1.0000
Inverse Matrix A_inv:
     -0.3333      0.0000      0.0000      0.0000      1.3333
      0.3333     -0.5000     -0.0000     -0.0000      0.1667
     -0.0000      0.5000     -1.0000     -0.0000      0.5000
     -0.0000     -0.0000      1.0000   -100.0001     99.0001
     -0.0000     -0.0000     -0.0000    100.0001   -100.0001
Matrix R = A*A_inv - I:
      0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
Norm of R:        0.000000
```

```
For var =   1.001000
Matrix A:
      1.0000      4.0000      4.0000      4.0000      4.0000
      1.0000      1.0000      3.0000      3.0000      3.0000
      1.0000      1.0000      1.0000      2.0000      2.0000
      1.0000      1.0000      1.0000      1.0000      1.0010
      1.0000      1.0000      1.0000      1.0000      1.0000
Inverse Matrix A_inv:
     -0.3333      0.0000      0.0000      0.0000      1.3333
      0.3333     -0.5000     -0.0000     -0.0000      0.1667
     -0.0000      0.5000     -1.0000     -0.0000      0.5000
     -0.0000     -0.0000      1.0000  -1000.0725    999.0725
     -0.0000     -0.0000     -0.0000   1000.0725  -1000.0725
Matrix R = A*A_inv - I:
      0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0001     -0.0001
     -0.0000      0.0000      0.0000      0.0000      0.0000
Norm of R:        0.000244
```

```
For var =   1.000100
Matrix A:
      1.0000      4.0000      4.0000      4.0000      4.0000
      1.0000      1.0000      3.0000      3.0000      3.0000
      1.0000      1.0000      1.0000      2.0000      2.0000
      1.0000      1.0000      1.0000      1.0000      1.0001
      1.0000      1.0000      1.0000      1.0000      1.0000
Inverse Matrix A_inv:
     -0.3333      0.0000      0.0000      0.0000      1.3333
      0.3333     -0.5000     -0.0000     -0.0000      0.1667
     -0.0000      0.5000     -1.0000     -0.0000      0.5000
     -0.0000     -0.0000      1.0000  -9986.4385   9985.4385
     -0.0000     -0.0000     -0.0000   9986.4385  -9986.4385
Matrix R = A*A_inv - I:
      0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000      0.0000      0.0000
     -0.0000      0.0000      0.0000     -0.0010      0.0010
     -0.0000      0.0000      0.0000      0.0000      0.0000
Norm of R:        0.001953
```