

СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА

P11-85-321

П.Г. Акишин

**SQROOT** - ПРОГРАММА РЕШЕНИЯ  
БОЛЬШИХ ЛИНЕЙНЫХ СИСТЕМ УРАВНЕНИЙ  
С СИММЕТРИЧНЫМИ МАТРИЦАМИ

1985

Многие задачи математической физики приводят к необходимости решения линейных систем уравнений с симметричными матрицами. Существующие методы решения можно разделить на два класса: прямые и итерационные. Большой набор прямых методов решения симметричных линейных систем уравнений для случая, когда матрица помещается в оперативной памяти ЭВМ, приведен в /1-4/. Алгоритм решения произвольной невырожденной линейной системы уравнений с использованием внешних устройств приведен в /5/. В данной работе предлагается прямой метод решения симметричных линейных систем уравнений с использованием периферийных устройств, причем элементы матрицы  $\{b_{ij}\}$  формируются с помощью подпрограммы-функции от двух целочисленных переменных  $i, j$ .

При разработке алгоритма использовался метод квадратного корня /4/.

#### Описание программы SQROOT

Пусть необходимо решить систему линейных уравнений

$$B\bar{x} = \bar{c}, \quad (I)$$

где  $B$  - симметричная квадратная матрица размера  $[NDIM \times NDIM]$ ,  $\bar{c}, \bar{x}$  - вектора размерности  $NDIM$ .

Для работы с подпрограммой SQROOT необходимы два рабочих файла TAPE1 - на устройстве внешней памяти с номером 1, и TAPE2 - на устройстве с номером 2. Обращаться к подпрограмме необходимо оператором

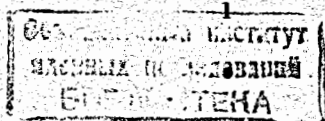
```
CALL SQROOT(A,A1,III,NN,NDIM,C,LB,JOB,FUNCT)
```

Параметры подпрограммы SQROOT:

$A, A1$  - рабочие массивы длиной  $LB$  каждый, используются как обменные буферы.

$NDIM$  - размерность решаемой системы уравнений.

$III$  - целочисленный массив длины, не меньший  $NDIM$ .



- NN - целочисленный рабочий массив, необходимый для записи элементов матрицы на TAPE2.  
 C - массив длиной, не меньшей NDIM, он содержит правую часть (I), и в него же помещается решение.  
 LB - длина буферов A, A1.  
 FUNCT - идентификатор подпрограммы-функции от двух целочисленных переменных. FUNCT(I,J) равно  $b_{IJ}$  элементу матрицы B из (I). FUNCT в вызывающей программе необходимо описать, как  
 EXTERNAL FUNCT

Подпрограмма-функция FUNCT(I,J) формируется пользователем.

JOB - управляющий код, принимает три значения 1,2,3. В случае, когда обращение к подпрограмме SQRROOT проводится с JOB=1, производится запись элементов исходной матрицы B на устройство с номером 2. После этого проводится факторизация матрицы, запись факторизованной матрицы на устройство с номером 1 и решение системы (I) с правой частью, содержащейся в массиве C. Решение помещается в массиве C, переменной JOB присваивается значение 2. Если факторизация матрицы становится невозможной - процесс обрывается, переменной JOB присваивается значение 3 и происходит возврат в место вызова подпрограммы SQRROOT. В случае, если обращение к SQRROOT происходит с JOB=2 (предполагается, что к этому моменту уже было обращение с JOB=1, и матрица уже факторизована), подпрограмма сразу идет на решение системы уравнений с правой частью из массива C.

Опишем кратко, как производится запись элементов матрицы на файл TAPE2. Матрица B из (I) записывается в файл TAPE2 следующим образом. Пусть первый столбец состоит из одного элемента  $b_{11}$ , второй -  $(b_{12}, b_{22})^T$ , третий -  $(b_{13}, b_{23}, b_{33})^T$ , ..., K-й столбец -  $(b_{1k}, b_{2k}, \dots, b_{kk})^T$  и т.д. Буферный массив A длиной LB последовательно заполняется такими усеченными столбцами максимально плотно. После этого производится запись массива A на TAPE2. При этом NN(1) присваивается значение 0, NN(2) равно суммарному количеству записанных к данному моменту элементов. После этого процедура повторяется. Массив A заполняется следующими столбцами, производится запись, NN(3) присваивается общее число записанных элементов и т.д., пока не будет записана вся матрица. Исходя из этого выбирается максимальная необходимая длина массива NN. Для просто-

ты длину массива NN можно задавать равной порядку решаемой системы уравнений.

Алгоритм решения системы (I) состоит из двух частей. В первой части проводится факторизация матрицы, т.е. находятся такие левая треугольная матрица S и диагональная матрица D, что

$$B = SDS^*$$

причем элементы матрицы D имеют значения, равные только 1 или -1. Во второй части производится умножение вектора  $\bar{c}$  из (I) последовательно на матрицы  $S^{-1}$ , D и  $(S^*)^{-1}$ , что эквивалентно решению исходной системы уравнений.

В таблице приведены счетные времена, затрачиваемые на решение системы линейных уравнений для симметричных матриц различной размерности на ЭВМ CDC-6500. Длина буфера LB бралась равной 10 000.

Таблица I.

Размерность систем уравнений	Время на факторизацию в С	Время на определение решения по уже факторизованной матрице в С
200	28.462	2.594
400	215.021	9.768
600	717.297	22.204

Автор признателен Л.А.Корнейчуку за ряд ценных замечаний по изложению материала.

Приложение I.

SUBROUTINE SQROOT(A,A1,III,NN,NDIM,C,LB,JOB,FUNCT)

```

C*
C*
C*   ОБЪЕДИНЕННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ
C*   ЛАБОРАТОРИЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И АВТОМАТИЗАЦИИ
C*   ДУБНА, 1985
C*
C*   SQROOT - ПРОГРАММА РЕШЕНИЯ ЛИНЕЙНЫХ СИСТЕМ УРАВНЕНИЙ С
C*   СИММЕТРИЧНЫМИ МАТРИЦАМИ С ИСПОЛЬЗОВАНИЕМ МЕТОДА КВАДРАТНОГО
C*   КОРНЯ.
C*
C*   NDIM - РАЗМЕРНОСТЬ СИСТЕМЫ.
C*
C*   A, A1 - РАБОЧИЕ МАССИВЫ ДЛИНОЙ LB КАЖДАЯ.
C*
C*   LB - ДЛИНА БУФЕРОВ A, A1.
C*
C*   III - ЦЕЛОЧИСЛЕННЫЙ РАБОЧИЙ МАССИВ.
C*
C*   NN - ЦЕЛОЧИСЛЕННЫЙ РАБОЧИЙ МАССИВ.
C*
C*   C - МАССИВ ДЛИНОЙ NDIM, СОДЕРЖИТ ПРАВУЮ ЧАСТЬ СИСТЕМ,
C*   И В НЕГО ПОМЕЩАЕТСЯ РЕШЕНИЕ СИСТЕМЫ.
C*
C*   FUNCT - ФУНКЦИЯ ВЫЧИСЛЕНИЯ ЭЛЕМЕНТОВ МАТРИЦЫ.
C*   В ВЫЗЫВАЮЩЕЙ ПРОГРАММЕ НЕОБХОДИМО СПИСАТЬ КАК EXTERNAL.
C*
C*   JOB - КОД ОПЕРАЦИИ 1, 2, 3
C*
C*   JOB = 1 ФАКТОРИЗАЦИЯ МАТРИЦЫ, Т. Е. ПРЕДСТАВЛЕНИЕ
C*   МАТРИЦЫ В ВИДЕ ПРОИЗВЕДЕНИЯ ЛЕВОЙ ТРЕУГОЛЬНОЙ
C*   МАТРИЦЫ S, ДИАГОНАЛЬНОЙ МАТРИЦЫ D И МАТРИЦЫ ST,
C*   ТРАНСПОНИРОВАННОЙ К МАТРИЦЕ S. МАТРИЦА D ИМЕЕТ НА
C*   ДИАГОНАЛИ ЛИБО 1, ЛИБО -1. ПОСЛЕ ЭТОГО ПРОИЗВОДИТСЯ
C*   РЕШЕНИЕ СИСТЕМЫ УРАВНЕНИЙ С ПРАВОЙ ЧАСТЬЮ ИЗ МАССИВА C.
C*   РЕШЕНИЕ ТАКЖЕ ПОМЕЩАЕТСЯ В МАССИВ C.
C*   ЕСЛИ СИСТЕМА ВЫРОЖДЕНА, JOB ПРИСВАИВАЕТСЯ ЗНАЧЕНИЕ 3.
C*   ПРОИЗВОДИТСЯ ВОЗВРАТ В ВЫЗЫВАЮЩУЮ ПРОГРАММУ.
C*
C*   JOB = 2 ФАКТОРИЗАЦИИ МАТРИЦЫ НЕ ПРОИЗВОДИТСЯ (ПРЕДПОЛАГАЕТСЯ,
C*   ЧТО МАТРИЦА УЖЕ ФАКТОРИЗОВАНА). ПРОГРАММА СРАЗУ ИДЕТ НА
C*   РЕШЕНИЕ СИСТЕМЫ УРАВНЕНИЙ. ПРИ ЭТОМ НЕОБХОДИМО
C*   СОХРАНЯТЬ МАССИВЫ III И NN.
C*
C*
C*   DIMENSION A(1),A1(1),NN(2),III(1),C(1)
C*   IF (JOB.EQ.2) GO TO 2000
C*   IF (JOB.EQ.3) RETURN
C*   NN(1)=0
C*   NK=1
C*   LL=0
C*   DO 1 K=1,NDIM
C*   LL=LL+K
C*   IF (LL.LE.LB) GO TO 1
C*   NK=NK+1
C*   NN(NK)=K-1
C*   LL=K
C*   CONTINUE
C*   NK=NK+1
C*   NN(NK)=NDIM
C*   REWIND 2

```

```

N0=NK-1
DO 2 K=1,N0
N1=NN(K)
N2=NN(K+1)
NNN=0
N1=N1+1
DO 3 L=N1,N2
DO 4 M=1,L
NNN=NNN+1
A(NNN)=FUNCT(L,M)
4 CONTINUE
3 CONTINUE
2 WRITE(2) (A(J),J=1,LB)
CONTINUE
K=K+1
NN0=0
N1=NN(1)
N2=NN(2)-1
REWIND 2
REWIND 1
5 READ(2) (A(J),J=1,LB)
GO TO 11
CONTINUE
6 READ(2) (A(J),J=1,LB)
LLLL=0
KL=0
REWIND 1
7 KL=KL+1
NN1=NN(KL)
NN2=NN(KL+1)
NN0=N2+1
NN=N1
IF (NN1.EQ.NN0) GO TO 11
READ(1) (A1(J),J=1,LB)
NNN=-LLLL
NNN=NNN+LLLL
LLLL=LLLL
IF (LLLL.EQ.NN2) GO TO 6
LLLL=LLLL+1
LLLL=-N1
DO 8 K=N1,N2
LLLL=LLLL+K
LLLL2=LLLL+LLLL
IF (LLLL.EQ.1) GO TO 9
DO 10 K1=1,LLLL
R1=A1(NNN+K1)*A(LLLL+K1)
10 IF (III(K1).EQ.0) R1=-R1
A(LLLL2)=A(LLLL2)+R1
9 A(LLLL2)=A(LLLL2)/A1(NNN+LLLL)
IF (III(LLLL).EQ.1) A(LLLL2)=-A(LLLL2)
8 CONTINUE
GO TO 7
11 CONTINUE
LLLL2=0
LLLL=N1
12 LLLL=LLLL
LLLL=LLLL+1

```

```

LLL3=LLL2+LLLL
IF(LLL1.EQ.0)GO TO 14
DO 13 K=1,LLL1
R1=A(LLL2+K)**2
IF(II(K).EQ.1)R1=-R1
A(LLL3)=A(LLL3)-R1
13 CONTINUE
14 CONTINUE
IF(A(LLL3))I 16,17
15 A(LLL3)=SQRT(-A(LLL3))
II(LLL)=1
GO TO 18
17 A(LLL3)=SQRT(A(LLL3))
II(LLL)=0
18 CONTINUE
IF(LLL.GT.N2)GO TO 22
LL3=LLL3
DO 19 K1=LLL,N2
LL3=LL3+K1
LL=LL3-LLL-1
IF(LLL1.EQ.0)GO TO 21
DO 20 K2=1,LLL1
R1=A(LLL2+K2)*A(LL1+K2)
IF(III(K2).EQ.0)R1=-R1
20 /A(LLL3)=A(LLL3)+R1
21 CONTINUE
IF(II(LLL).EQ.1)A(LLL3)=-A(LLL3)
A(LLL3)=A(LLL3)/A(LLL3)
19 CONTINUE
LLL2=LLL3
GO TO 12
22 CONTINUE
WRITE(1) (A(J),J=1,LB)
IF(NN00.EQ.NDIM)GO TO 23
K444=K444+1

N1=NN(K444)
N2=NN(K444+1)-1
NN0=N1
NN00=N2+1
GO TO 5
23 CONTINUE
GO TO 2000
16 CONTINUE
JOB=3
RETURN
2000 CONTINUE
K444=0
LLL=0
NN00=0
REWIND 1
24 CONTINUE
IF(NN00.EQ.NDIM)GO TO 28
K444=K444+1
NN00=NN(K444+1)
N444=0
25 READ(1)(A(J),J=1,LB)
LLL=LLL
LLL=LLL+1

```

```

IF(LLL1.EQ.0)GO TO 27
DO 26 K=1,LLL1
NNN=NNN+1
26 C(LLL)=C(LLL)-C(K)*A(NNN)
27 NNN=NNN+1
C(LLL)=C(LLL)/A(NNN)
IF(LLL.EQ.NN00)GO TO 24
GO TO 25
28 CONTINUE
DO 29K=1,NDIM
IF(II(K).EQ.1)C(K)=-C(K)
29 CONTINUE
N1=NN(K444)
N2=NN(K444+1)
NNN=(N2-N1)*(N2+N1+1)/2
GO TO 31
30 K444=K444-1
IF(K444.EQ.0)GO TO 33
N1=NN(K444)
N2=NN(K444+1)
NNN=(N2-N1)*(N2+N1+1)/2
BACKSPACE 1
BACKSPACE 1

READ(1)(A(J),J=1,LB)
31 CONTINUE
IF(LLL.EQ.N1)GO TO 30
C(LLL)=C(LLL)/A(NNN)
X1=C(LLL)
LLL=LLL-1
NNN=NNN-1
IF(NNN.EQ.0)GO TO 30
DO 32 K=1,LLL
C(LLL+1-K)=C(LLL+1-K)-X1*A(NNN)
NNN=NNN-1
32 CONTINUE
GO TO 31
33 CONTINUE
RETURN
END

```

Приложение 2

```

SUBROUTINE GAUSS(LB,NDIM,NQ,A,A1,IJJ,C,X,JOB)
C*
C*
C* ОБЪЕДИНЕННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ
C* ЛАБОРАТОРИЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И АВТОМАТИЗАЦИИ
C* ДУБНА, 1984
C*
C*
C* GAUSS - ПРОГРАММА РЕШЕНИЯ ЛИНЕЙНЫХ СИСТЕМ
C* УРАВНЕНИЙ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ГАУССА С ВЫБОРОМ
C* ГЛАВНОГО ЭЛЕМЕНТА ПО СТОЛБЦУ.
C* В * X = C
C* МАТРИЦА В ХРАНИТСЯ В ФАЙЛЕ TARE1 .
C*
C* NDIM - РАЗМЕРНОСТЬ СИСТЕМЫ.
C*
C* A , A1 - РАБОЧИЕ МАССИВЫ ДЛИНОЙ LB КАЖДЫЙ.
C*
C* LB - ДЛИНЫ РАБОЧИХ БУФЕРОВ A , A1 .
C*
C* IJJ - РАБОЧИЙ МАССИВ ДЛИНОЙ NDIM , В НЕМ
C* ХРАНИТСЯ ВЕКТОР ПЕРЕСТАНОВКИ.
C*
C* C - МАССИВ ДЛИНОЙ NDIM , СОДЕРЖИТ ПРАВУЮ ЧАСТЬ
C* СИСТЕМЫ.
C* X - МАССИВ ДЛИНОЙ NDIM , В НЕГО ПОМЕЩАЕТСЯ
C* РЕШЕНИЕ СИСТЕМЫ.
C*
C* JOB - КОД ОПЕРАЦИИ 1, 2 .
C*
C* JOB =1 ФАКТОРИЗАЦИЯ МАТРИЦЫ В , Т.Е.
C* ПРЕДСТАВЛЯЕТСЯ В ВИДЕ ПРОИЗВЕДЕНИЯ
C* ОБРАТНОЙ К ЛЕВОЙ ТРЕУГОЛЬНОЙ МАТРИЦЫ S НА
C* ПРАВУЮ ТРЕУГОЛЬНУЮ МАТРИЦУ R .
C* ФАКТОРИЗОВАННАЯ МАТРИЦА ЗАПИСЫВАЕТСЯ НА ФАЙЛ
C* TARE2 . ПРОВОДИТСЯ РЕШЕНИЕ СИСТЕМЫ УРАВНЕНИЙ
C* С ПРАВой ЧАСТЬЮ C .
C* ЕСЛИ СИСТЕМА НЕУЫРОЖДЕННАЯ , JOB ПРИСВАЕВАЕТСЯ
C* ЗНАЧЕНИЕ 2.
C* ЕСЛИ СИСТЕМА ВЫРОЖДЕННАЯ , JOB ПРИСВАЕВАЕТСЯ
C* ЗНАЧЕНИЕ 3, ДАЕТСЯ ПРЕДУПРЕДИТЕЛЬНАЯ ДИАГНОСТИКА.
C*
C* JOB =2 РЕШЕНИЕ ПО УЖЕ ФАКТОРИЗОВАННОЙ
C* МАТРИЦЕ. ИСПОЛЬЗУЕТСЯ МАССИВ IJJ , ПОЛУЧЕННЫЙ
C* В ХОДЕ ФАКТОРИЗАЦИИ. ПРАВая ЧАСТЬ C НЕ
C* СОХРАНЯЕТСЯ. ПЕРВОНАЧАЛЬНАЯ МАТРИЦА И ФАКТОРИЗОВАННЫЕ
C* МАТРИЦЫ СОХРАНЯЮТСЯ НА ФАЙЛАХ TARE1 И TARE2 .
C*
C* ПОДРОБНОЕ ОПИСАНИЕ ПРОГРАММЫ GAUSS СОДЕРЖИТСЯ
C* В РАБОТЕ : АКИВИЧ П. Г., ДИТИ, Р11-84-493, ДУБНА, 1984.
C*
C*
C* DIMENSION A(LB),A1(LB),IJJ(NDIM),C(NDIM),X(NDIM)
C*
C* JOB =2 ПРОГРАММА СРАЗУ ПРИСТУПАЕТ К РЕШЕНИЮ
C* СИСТЕМЫ УРАВНЕНИЙ.
C*
C* IF(JOB.EQ.2)GO TO 17

```

```

C*
C*
C* ФАКТОРИЗАЦИЯ МАТРИЦЫ.
C*
C* ОПРЕДЕЛЕНИЕ МАШИНОГО НУЛЯ.
C*
C*
C* EPS=1.
C* EPS1=EPS*1.5
C* IF(EPS1.EQ.0.)GO TO 2
C* EPS=EPS1
C* GO TO 1
C* CONTINUE
C*
C* 2
C* REWIND 1
C* NQ=NDIM
C* KQ=NQ
C* KTQ=NQ
C* IF(KQ.GT.NQ)KQ=NQ
C* IF(KTQ.GT.NQ)KTQ=NQ
C*
C*
C* ПЕРВОНАЧАЛЬНАЯ НУМЕРАЦИЯ СТРОК.
C*
C*
C* DO 3 K=1,NQ
C* IJJ(K)=K
C* NUMB=0
C* LLLQ=0
C*
C*
C* ЧТЕНИЕ ПЕРВЫХ СТОЛБЦОВ МАТРИЦЫ.
C*
C*
C* READ(1) (A(LK),LK=1,LB)
C* REWIND 2
C* GO TO 9
C* CONTINUE
C*
C*
C* ПРОВЕДЕНИЕ С ЭЛЕМЕНТАМИ ОЧЕРЕДНЫХ
C* СТОЛБЦОВ ДЕЙСТВИЙ, КОТОРЫЕ
C* ВЫПОЛНЯЛИСЬ СЭ СТРОКАМИ
C* МАТРИЦЫ К ДАННОМУ МОМЕНТУ.
C*
C*
C* IF(NUMB.EQ.NQ)GO TO 15
C* IF(NUMB+KQ.GT.NQ)KTQ=NQ-NUMB
C* READ(1) (A(LK),LK=1,LB)
C* REWIND 2
C* KKQ=0
C* DO 5 K1=1,LLLQ
C* READ(2) (A1(LK),LK=1,LB)
C* NDIF=-NQ
C* DO 6 K2=1,KQ
C* NDIF=NDIF+KQ
C* KKQ=KKQ+1
C* KK1=KKQ+1
C* KK5=IJJ(KK1)
C* NNN1=-NQ
C* DO 7 K3=1,KTQ
C* NNN1=NNN1+KQ
C* R1=A(NNN1+KK5)
C* DO 8 K4=KK1,NQ
C* I1=IJJ(K4)
C* A(NNN1+I1)=R1*A1(NDIF+I1)+A(NNN1+I1)
C* CONTINUE
C* CONTINUE
C* CONTINUE
C* CONTINUE

```

```

C*
C* ОБРАБОТКА ОЧЕРЕДНЫХ СТОЛБЦОВ МАТРИЦЫ.
C*
      NDIF=-N0
      DO 10 K1=1,KT0
      NUMB=NUMB+1
      NDIF=NDIF+ND
      KK1=NUMB+1
      IF(KK1.GT.N0)GO TO 10
C*
C* ПОИСК ГЛАВНОГО ЭЛЕМЕНТА В СТОЛБЦЕ.
C*
      IMAX=NUMB
      RMAX=ABS(A(NDIF+IIJ(NUMB)))
      DO 11 K2=KK1,N0
      IF(ABS(A(NDIF+IIJ(K2))).LE.RMAX)GO TO 11
      IMAX=K2
      RMAX=ABS(A(NDIF+IIJ(IMAX)))
11      CONTINUE
C*
C* ПЕРЕСТАНОВКА СТРОК МАТРИЦЫ.
C*
      K7=IIJ(IMAX)
      I1=IIJ(NUMB)
      IIJ(NUMB)=K7
      IIJ(IMAX)=I1
C*
C* ПРОВЕРКА МАТРИЦЫ НА ВЫРОЖДЕННОСТЬ.
C*
      IF(ABS(A(NDIF+K7)).LT.EPS)GO TO 16
C*
C* ВЫЧИСЛЕНИЕ ЭЛЕМЕНТОВ ТРЕУГОЛЬНЫХ МАТРИЦ,
C* ВХОДЯЩИХ В РАССМАТРИВАЕМЫЕ СТОЛБЦЫ.
C*
      RMAX=-1./A(NDIF+K7)
      DO 12 K2=KK1,N0
      K5=IIJ(K2)+NDIF
12      A(K5)=A(K5)*RMAX
      IF(K1.EQ.KT0)GO TO 10
      KK2=K1+1
      NNN1=NDIF
      DO 13 K3=KK2,KT0
      NNN1=NNN1+ND
      R1=A(NNN1+K7)
      DO 14 K4=KK1,N0
      I1=IIJ(K4)
14      A(NNN1+I1)=A(NNN1+I1)+R1*A(NDIF+I1)
13      CONTINUE
10      CONTINUE
C*
C* ЗАПИСЬ ЭЛЕМЕНТОВ ТРЕУГОЛЬНЫХ МАТРИЦ,
C* ВХОДЯЩИХ В РАССМАТРИВАЕМЫЕ СТОЛБЦЫ.
C*
      WRITE(2) (A(LK),LK=1,LB)
      LLL0=LLL0+1
      GO TO 4
15      CONTINUE
      GO TO 17
16      CONTINUE
C*

```

```

C* МАТРИЦА ВЫРОЖДЕННАЯ.
C*
200  FORMAT(10X,19HDETERMINANT IS ZERO)
      PRINT 200
      JOB=3
      RETURN
17      CONTINUE
C*
C* РЕШЕНИЕ СИСТЕМЫ УРАВНЕНИЙ С ФАКТОРИЗОВАННОЙ
C* МАТРИЦЕЙ.
C*
      REWIND 2
      NUMB=0
      KT0=NQ
      IF(KT0.GT.N0)KT0=N0
      NNN1=N0*(KT0-1)
18      CONTINUE
      IF(NUMB+NQ.GT.N0)KT0=N0-NUMB
      READ(2) (A(LK),LK=1,LB)
      NDIF=-N0
      DO 20 K1=1,KT0
C*
C* УМНОЖЕНИЕ ПРАВОЙ ЧАСТИ НА МАТРИЦУ, ОБРАТНУЮ
C* К ЛЕВОЙ ТРЕУГОЛЬНОЙ.
C*
      NDIF=NDIF+N0
      NUMB=NUMB+1
      KK1=NUMB+1
      IF(KK1.GT.N0)GO TO 20
      R1=C(IIJ(NUMB))
      DO 19 K2=KK1,N0
      I1=IIJ(K2)
19      C(I1)=C(I1)+R1*A(NDIF+I1)
20      CONTINUE
      IF(NUMB.NE.N0)GO TO 18
C*
C* ОБРАЩЕНИЕ ПРАВОЙ ТРЕУГОЛЬНОЙ МАТРИЦЫ.
C*
21      I1=IIJ(NUMB)
      X(NUMB)=C(I1)/A(NDIF+I1)
      KK1=NUMB-1
      R1=X(NUMB)
      IF(KK1.EQ.0)GO TO 23
      DO 22 K1=1,KK1
      I1=IIJ(K1)
22      C(I1)=C(I1)-A(NDIF+I1)*R1
23      CONTINUE
      NUMB=NUMB-1
      NDIF=NDIF-N0
      IF(NDIF.GE.0)GO TO 21
      IF(NUMB.EQ.0)GO TO 24
      BACKSPACE 2
      BACKSPACE 2
      READ(2) (A(LK),LK=1,LB)
      NDIF=NNN1
      GO TO 21
24      CONTINUE
      JOB=2
      RETURN
      END

```

Литература

1. Фаддеев Р.К., Фаддеева В.Н. Вычислительные методы линейной алгебры. Физматгиз, 1960.
2. Уилкинсон Д.Х. Алгебраическая проблема собственных значений. "Наука", М., 1970.
3. Уилкинсон Д.Х., Райншт. Справочник алгоритмов на языке Алгол. Линейная алгебра. "Машиностроение", М., 1976.
4. Калиткин Н.Н. Численные методы. "Наука", М., 1978.
5. Акишин П.Г. ОИЯИ, П11-84-493, Дубна, 1984.

Рукопись поступила в издательский отдел  
6 мая 1985 года

Акишин П.Г.

P11-85-321

SQROOT - программа решения больших линейных систем уравнений с симметричными матрицами

В работе дано описание программы решения больших линейных систем уравнений с симметричными матрицами. В основу положен метод квадратного корня. Приводится текст программы на фортране. Приведены тестовые расчеты.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1985

Перевод О.С. Виноградовой

Akishin P.G.

P11-85-321

SQROOT - Program for Solving Large Linear Systems with Symmetric Matrices

Programs for solving large linear systems with symmetric matrices is described. The method of square root is used. The results of test calculation are discussed.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna 1985