Практическое занятие №3

Работа с матрицами

Внимание! Вариант задания соответствует порядковому номеру в списке и будет закреплен за вами на протяжении всего семестра.

Первое знакомство с матрицами состоялось в ходе выполнения практического занятия №1.

Вот некоторые основные команды:
1) Создать вектор-строку V
-->V=[1 2 3 4 5]
2) Создать вектор-столбец R
-->R=[1; 2; 3; 4; 5]
3) Создать матрицу L
-->L=[1 2 3;4 5 6;7 8 9]
4) Обратится к элементу массива L(2,2)
5) Выделить из матрицы L второй столбец
-->L=(:,2)
6) Выделить из матрицы L подматрицу M
—> M=L(1:2,2:3)
7) Удалить из матрицы L второй столбец
—> L(:,2)=[]
8) Представить матрицу M в виде вектора-столбца
—>v=M(:)
9) Выделить из вектора v элементы со второго по четвертый
—>b=v(2:4)

В SciLab предусмотрены следующие действия над матрицами:
- сложение и вычитание (операции сложения и вычитания определены для матриц одной размерности или векторов одного типа, т.е. суммировать (вычитать) можно либо векторы-столбцы, либо векторы-строки одинаковой длины);
- тренспонирование (если в некоторой матрице заменить строки соответствующими столбцами, то получится транспонированная матрица);
- матричное умножение (операция умножения вектора на вектор определена только для векторов одинакового раз- мера, причем один из них должен быть вектором-столбцом, а второй вектором-строкой. Матричное умножение выполняется по правилу «строка на столбец» и допустимо, если количество строк во второй матрице совпадает с количеством столбцов в первой);
- умножение матрицы на число;
- возведение в степень, т.е. умножение матрицы саму на себя n раз (целочисленный показатель степени может быть как положительным, так и отрицательным. В первом случае выполняется алгоритм умножения матрицы на себя указанное число раз, во втором умножается на себя матрица, обратная к данной.);
- левое и правое деление;
- поэлементное: умножение матриц, возведение в степень, левое деление, правое деление;

Задача 1. Изучить простые действия с матрицами:
1) Завести две матрицы Q и R согласно вашему варианту.
2) Транспонировать матрицы так, что Q_T=A и R_T=B
3) Вычислить $(A+R)^2$-2Q(0.5B-Q)

4) Вычислить $Q_T(2R-R_T)$
5) Решить матричные уравнения Q•X=R и X•Q=R
6) Применить к массиву Q функцию sin(Q)

Пример выполнения:
1)
```
--> Q=[1 2 0;-1 3 1;4 -2 5]
 Q  =

   1.   2.   0.
  -1.   3.   1.
   4.  -2.   5.


--> R=[-1 0 1;2 1 1;3 -1 -1]
 R  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.
```

2)
```
--> A=Q'
 A  =

   1.  -1.   4.
   2.   3.  -2.
   0.   1.   5.


--> B=R'
 B  =

  -1.   2.   3.
   0.   1.  -1.
   1.   1.  -1.
```

3)
```
--> (A+R)^2+2*Q*(0.5*B-Q)
 ans  =

   12.  -16.   18.
   15.   4.   -11.
  -39.   24.  -6.
```

4)
```
--> Q'*(2*R-R')
 ans  =

   15.  -15.  -8.
   0.     5.    9.
   29.  -14.  -2.
```

5) Типовое матричное уравнение состоит как правило из нескольких матриц и неизвестной матрицы X, которую предстоит найти. То есть, решением матричного уравнения является также матрица.
Решение таких уравнений происходит аналогично решению, например, уравнения 2*x*+3=1 из которого нужно найти неизвестную величину *x*. Матричные уравнения устроены практически так же, только вместо чисел – матрицы (и конечно, числа тоже есть, помним,

что матрицу можно умножить на число). Плюс особенности, характерные для действий с матрицами.
Проделаем решение такого уравнения без использования SciLab:

$$\begin{pmatrix} 1 & 2 & 0 \\ -1 & 3 & 1 \\ 4 & -2 & 5 \end{pmatrix} X = \begin{pmatrix} -1 & 0 & 1 \\ 2 & 1 & 1 \\ 3 & -1 & -1 \end{pmatrix}$$

В общем случае, выражение может быть более сложным, и требуется приведение его к общему виду QX=R. В данном случае выбрано простое уравнение.
Для разрешения уравнения относительно X умножим обе его части на Q^{-1} слева (здесь и далее предполагаем, что обратная матрица существует):
$Q^{-1}QX=Q^{-1}B$
!!! Внимание! Произведение матриц не перестановочно, поэтому критически важно, с какой стороны проводить умножение.
По свойству матричных операций $Q^{-1}Q=E$ – единичная матрица, поэтому:
$EX=Q^{-1}R$
По правилам единичную матрицу можно убрать:
$X=Q^{-1}R$
При этом нам не известна матрица Q^{-1}.
Теперь проделаем те же действия с уравнением XQ=R:
$XQQ^{-1}=RQ^{-1}$
$XE=RQ^{-1}$
$X=RQ^{-1}$. Готово.
Матрица Q^{-1} нам все еще не известна.
Следующим шагом является вычисление обратной матрицы Q^{-1} и умножение $Q^{-1}R$ или RQ^{-1}, в результате чего получим X.

Проделайте вычисления на бумаге, получите X, представьте ход вычислений в отчете. После этого проверьте ответ с помощью SciLab, как показано ниже.

```
--> //Решение матричного уравнения QX=R:

--> X=Q\R
 X  =

  -0.8857143  -0.3428571   0.1428571
  -0.0571429   0.1714286   0.4285714
   1.2857143   0.1428571  -0.1428571


--> //Решение матричного уравнения XQ=R:

--> X=R/Q
 X  =

  -0.7714286   0.5714286   0.0857143
   0.9428571  -0.1428571   0.2285714
   1.4857143  -1.2857143   0.0571429


--> //Проверка

--> X*Q-R
 ans  =

   0.          0.   0.
   0.          0.   0.
   4.441D-16   0.   0.
```

6)

```
--> sin(Q)
 ans  =

   0.841471    0.9092974   0.
  -0.841471    0.14112     0.841471
  -0.7568025  -0.9092974  -0.9589243
```

Вариант	Задание
1	$Q=\begin{pmatrix}1 & 2 & 3\\ 3 & 3 & 1\\ 4 & -1 & 1\end{pmatrix},\quad R=\begin{pmatrix}-1 & 1 & 1\\ 2 & 3 & 2\\ 3 & 2 & -1\end{pmatrix}$
2	$Q=\begin{pmatrix}-1 & -2 & 0\\ 1 & 3 & -1\\ 4 & 2 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & -1\\ -2 & 5 & 1\\ 3 & 2 & 1\end{pmatrix}$
3	$Q=\begin{pmatrix}4 & 2 & 0\\ -1 & 4 & -1\\ 4 & -2 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & 3\\ 3 & 1 & 1\\ 3 & -3 & -1\end{pmatrix}$
4	$Q=\begin{pmatrix}1 & 5 & 0\\ -1 & 3 & 1\\ 4 & -5 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & 0\\ 2 & 3 & 1\\ -1 & -1 & -1\end{pmatrix}$
5	$Q=\begin{pmatrix}1 & 2 & 4\\ -1 & 1 & 1\\ 4 & -2 & 1\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & 1\\ 2 & 0 & 1\\ 3 & -1 & 1\end{pmatrix}$
6	$Q=\begin{pmatrix}1 & 2 & 3\\ -1 & 2 & 1\\ 1 & -2 & 3\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & 5\\ 2 & 0 & 1\\ -1 & -1 & -1\end{pmatrix}$
7	$Q=\begin{pmatrix}1 & -2 & 0\\ -4 & -3 & -1\\ 4 & -2 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 3 & 1\\ -2 & 2 & 1\\ 3 & 1 & -1\end{pmatrix}$
8	$Q=\begin{pmatrix}1 & 2 & 0\\ 1 & 3 & 1\\ 4 & 2 & 5\end{pmatrix},\quad R=\begin{pmatrix}1 & 0 & 1\\ 2 & 1 & 1\\ 3 & 1 & 1\end{pmatrix}$
9	$Q=\begin{pmatrix}1 & -2 & 0\\ 1 & -3 & 1\\ 4 & -2 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & -1\\ 4 & -1 & 1\\ 3 & -1 & -1\end{pmatrix}$
10	$Q=\begin{pmatrix}1 & 4 & 0\\ -3 & -2 & 5\\ 4 & 4 & 5\end{pmatrix},\quad R=\begin{pmatrix}-1 & 0 & 1\\ 0 & 1 & 0\\ 3 & 0 & -1\end{pmatrix}$

11	$Q=\begin{pmatrix}3 & 2 & 0\\ 1 & -3 & -1\\ -4 & -2 & -5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 0 & 1\\ -2 & 1 & 1\\ -3 & -1 & -2\end{pmatrix}$
12	$Q=\begin{pmatrix}1 & 2 & 0\\ 0 & 3 & 1\\ 4 & -1 & 1\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 2 & 0\\ 0 & 1 & 1\\ 3 & -1 & 1\end{pmatrix}$
13	$Q=\begin{pmatrix}-1 & -2 & 0\\ -1 & -3 & -1\\ 4 & -2 & 5\end{pmatrix}, \quad R=\begin{pmatrix}4 & 0 & 1\\ 2 & 1 & 1\\ 3 & 1 & 4\end{pmatrix}$
14	$Q=\begin{pmatrix}-1 & 2 & 0\\ -1 & 3 & 1\\ -4 & -2 & -5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 0 & -1\\ 2 & 1 & -1\\ 3 & 1 & -1\end{pmatrix}$
15	$Q=\begin{pmatrix}-1 & -2 & -3\\ 1 & 3 & -1\\ 4 & 2 & -5\end{pmatrix}, \quad R=\begin{pmatrix}1 & 0 & 1\\ 2 & 2 & 1\\ 3 & -1 & 3\end{pmatrix}$
16	$Q=\begin{pmatrix}1 & -2 & 0\\ -1 & -3 & -1\\ 4 & -2 & 5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 3 & 1\\ 2 & 3 & 1\\ 3 & 3 & -1\end{pmatrix}$
17	$Q=\begin{pmatrix}1 & 2 & 2\\ -1 & 3 & 1\\ -2 & 4 & 5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 0 & 1\\ -4 & 1 & -4\\ -3 & 1 & 4\end{pmatrix}$
18	$Q=\begin{pmatrix}1 & 0 & 2\\ -1 & 0 & 1\\ 4 & 0 & 5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 0 & 1\\ 0 & 1 & 1\\ 3 & -1 & -1\end{pmatrix}$
19	$Q=\begin{pmatrix}1 & 2 & 4\\ -2 & 6 & 1\\ 4 & -4 & 5\end{pmatrix}, \quad R=\begin{pmatrix}-1 & 0 & -1\\ -2 & 1 & 1\\ -3 & -1 & -1\end{pmatrix}$
20	$Q=\begin{pmatrix}2 & 2 & 2\\ -3 & 3 & 3\\ 4 & -2 & 5\end{pmatrix}, \quad R=\begin{pmatrix}3 & 0 & 3\\ 2 & 3 & 1\\ -1 & -1 & -1\end{pmatrix}$

Задача 2. Изучить специальные матричные функции

1) функция matrix(A,[,n,m]) – преобразует матрицу A в матрицу другого размера;
2) функция ones(m,n) – создает матрицу единиц из m строк и n столбцов;
3) функция zeros(m,n) – создает нулевую матрицу из m строк и n столбцов;
4) функция eye(m,n) – формирует единичную матрицу из m строк и n столбцов;
5) функция rand(n1,n2,…nn[,fl]) – формирует многомерную матрицу случайных чисел;
6) функция sparse([i1 j1;i2 j2;…;in jn],[n1,n2,…,nn])—формирует разреженную матрицу и функция full(M) — вывод разреженной матрицы M в виде таблицы;;

7) diag(V[,k]) — возвращает квадратную матрицу с элементами V на главной или на k-й диагонали;
8) cat(n, A, B, [C, …]) — объединяет матрицы A и B или все входящие матрицы, при n=1 по строкам, при n=2 по столбцам; то же что [A; B] или [A, B];
9) tril(A[,k]) — формирует из матрицы A нижнюю треугольную матрицу, начиная с главной или с k-й диагонали;
10) triu(A[,k]) — формирует из матрицы A верхнюю треугольную матрицу, начиная с главной или с k-й диагонали;
11) gsort(X) — выполняет упорядочивание массива X; если X — матрица, сортировка выполняется по столбцам;

Попробовать применение всех перечисленных функций (с матрицей из задания 1, где это возможно или со своими собственными параметрами матриц отличными от представленного примера). Результат отразить в отчете.

Пример:
1)

```
--> R=[-1 0 1;2 1 1;3 -1 -1]
 R  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.


--> matrix(R,9,1)
 ans  =

  -1.
   2.
   3.
   0.
   1.
  -1.
   1.
   1.
  -1.
```

2) -3)

```
--> ones(3,4)
 ans  =

   1.   1.   1.   1.
   1.   1.   1.   1.
   1.   1.   1.   1.


--> zeros(2,5)
 ans  =

   0.   0.   0.   0.   0.
   0.   0.   0.   0.   0.


--> zeros(3,4)
 ans  =

   0.   0.   0.   0.
   0.   0.   0.   0.
   0.   0.   0.   0.
```

4)

```
--> eye(2,2)
 ans  =

   1.   0.
   0.   1.


--> eye(3,7)
 ans  =

   1.   0.   0.   0.   0.   0.   0.
   0.   1.   0.   0.   0.   0.   0.
   0.   0.   1.   0.   0.   0.   0.
```

5)

```
--> rand(3,5)
 ans  =

   0.2113249   0.3303271   0.8497452   0.068374    0.7263507
   0.7560439   0.6653811   0.685731    0.5608486   0.1985144
   0.0002211   0.6283918   0.8782165   0.6623569   0.5442573


--> rand(3,5)
 ans  =

   0.2320748   0.8833888   0.9329616   0.3616361   0.4826472
   0.2312237   0.6525135   0.2146008   0.2922267   0.3321719
   0.2164633   0.3076091   0.312642    0.5664249   0.5935095
```

6)

```
--> A=sparse([1 3;3 2;3 5],[4,5,6])
 A  =

(  3,  5) sparse matrix

(  1,  3)         4.
(  3,  2)         5.
(  3,  5)         6.


--> full(A)
 ans  =

   0.   0.   4.   0.   0.
   0.   0.   0.   0.   0.
   0.   5.   0.   0.   6.
```

7)

```
--> R=[-1 0 1;2 1 1;3 -1 -1]
 R  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.


--> //Найдем главную диагональ

--> diag(R)
 ans  =

  -1.
   1.
  -1.
```

8)

```
--> R=[-1 0 1;2 1 1;3 -1 -1]
 R  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.


-->  Q=[1 2 0;-1 3 1;4 -2 5]
 Q  =

   1.   2.   0.
  -1.   3.   1.
   4.  -2.   5.


--> //Объединим матрицы из задания 1

--> cat(2,R,Q)
 ans  =

  -1.   0.   1.   1.   2.   0.
   2.   1.   1.  -1.   3.   1.
   3.  -1.  -1.   4.  -2.   5.


--> cat(1,R,Q)
 ans  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.
   1.   2.   0.
  -1.   3.   1.
   4.  -2.   5.
```

9)

```
-->  Q=[1 2 0;-1 3 1;4 -2 5]
 Q  =

   1.   2.   0.
  -1.   3.   1.
   4.  -2.   5.


--> //Нижняя треугольная матрица, начиная с главной диагонали

--> tril(Q)
 ans  =

   1.   0.   0.
  -1.   3.   0.
   4.  -2.   5.


--> //Нижняя треугольная матрица, начиная с диагонали на одну ниже главной

--> tril(Q,-1)
 ans  =

   0.   0.   0.
  -1.   0.   0.
   4.  -2.   0.
```

10)

```
--> triu(Q)
 ans  =

   1.   2.   0.
   0.   3.   1.
   0.   0.   5.


--> triu(Q,2)
 ans  =

   0.   0.   0.
   0.   0.   0.
   0.   0.   0.
```

11)

```
--> R=[-1 0 1;2 1 1;3 -1 -1]
 R  =

  -1.   0.   1.
   2.   1.   1.
   3.  -1.  -1.


--> gsort(R)
 ans  =

   3.   1.  -1.
   2.   1.  -1.
   1.   0.  -1.
```

```
--> A=rand(1,4)
 A  =

   0.4094825   0.8784126   0.113836   0.1998338


--> //Сортировка по убыванию

--> gsort(A)
 ans  =

   0.8784126   0.4094825   0.1998338   0.113836


--> //Сортировка по возрастанию

--> -gsort(-A)
 ans  =

   0.113836   0.1998338   0.4094825   0.8784126
```

Задание 3. Работа с функциями вычисления различных числовых характеристик матриц. Все функции применить к своим массивам по вариантам. Для некоторых функций примеры не приведены, выполнить самостоятельно.

1) size(V[,fl])—определяет размер массива V; если V — двумерный массив, то size(V,1) или size(V,’r’) определяют число строк матрицы V, а size(V,2) или size(V,’c’) — число столбцов;

```
--> size(R)
 ans  =

   3.   3.


--> size(R,1)
 ans  =

   3.
```

2) length(X) — определяет количество элементов массива X; если X — вектор, его длину; если X — матрица, вычисляет общее число ее элементов;

3) sum(X[,fl]) — вычисляет сумму элементов массива X, имеет необязательный параметр fl. Если параметр fl отсутствует, то функция sum(X) возвращает скалярное значение, равное сумме элементов массива. Если fl=’r’ или fl=1, что то же самое, то функция вернет строку, равную поэлементной сумме столбцов матрицы X. Если fl=’c’ или fl=2, то результатом работы функции будет вектор-столбец, каждый элемент которого равен сумме элементов строк матрицы X. Частный случай применения функции sum — это вычисление скалярного произведения векторов;

```
--> A=sum(R) //Сумма всех элементов
 A  =

   5.


--> sum(Q,1) //Сумма по строкам
 ans  =

   4.   3.   6.


--> sum(Q,2) //Сумма по столбцам
 ans  =

   3.
   3.
   7.
```

4) prod(X[,fl])—вычисляет произведение элементов массива X, работает аналогично функции sum;

5) max(M[,fl]) — вычисляет наибольший элемент в массиве M, имеет необязательный параметр fl. Если параметр fl отсутствует, то функция max(M) возвращает максимальный элемент массива M; если fl=’r’, то функция вернет строку максимальных элементов столбцов матрицы M; если fl=’c’, то результатом работы функции будет вектор-столбец, каждый элемент которого равен максимальному элементу соответствующих строк матрицы M. Функция [x, nom]=max(M[,fl]) вернет значение максимального элемента x и его номер в массиве nom;

min(M[,fl]) — вычисляет наименьший элемент в массиве M, работает аналогично функции max;

6) mean(M[,fl]) — вычисляет среднее значение массива M; если M двумерный массив, то mean(M,1) или mean(M,’r’) определяют среднее значение строк матрицы M, а mean(M,2) или mean(M,’c’) — среднее значение столбцов;

median(M[,fl]) — вычисляет медиану (значение, которое делит массив на две части), работает аналогично функции mean;

7) det(M) — вычисляет определитель квадратной матрицы M;

8) rank(M[,tol]) — вычисление ранга (максимальное число линейно независимых строк) матрицы M с точностью tol.

Задание 4. Изучение функций, реализующих численные алгоритмы решения задач линейной алгебры.

1) spec(M) — вычисляет собственные значения и собственные векторы квадратной матрицы M.

2) inv(A) — вычисляет матрицу, обратную к A.

Функция не работает с вырожденными матрицами.

Выполнить вычисление обратной матрицы на бумаге. Представить решение в отчете. Затем проверить результат с помощью SciLab. Удивиться экономии времени и сил.

3) Ознакомиться с примерами дополнительных функций в учебнике [1] на с. 47-52. Повторить представленные примеры. Результат отразить в отчете.