**Вариативная самостоятельная работа задание №3.2.**

**Справочник по формулам Scilab, используемых при работе с матрицами.**

1. Ввод и вывод матрицы

Матрицы можно задавать следующим образом:

***name = Xn:dX:Xk***,

где name - имя переменной, хранящей матрицу, Xn - значение первого элемента массива, dX - шаг, с помощью которого формируется следующий элемент массива, Xk - последний элемент массива.

Также матрицы можно вводить поэлементно:

***name = [x1 x2 ... xn]*** или ***name = [x1, x2, …, xn]***

Элементы вектора-столбца вводятся через точку с запятой:

***name = [x1; x2; …; xn]***

Ввод элементов двумерной матрицы:

***name = [x11, x21, …, x1n; x21, x22, …, x2n; …; xm1, xm2, …, xmn]***

Обратиться к элементу матрицы можно указав после имени матрицы в круглых скобках через запятую номер строки и номер столбца:

***name = (i, j)***

Также можно составлять матрицы из других матриц:

***name = [V V V] или name = [V;V]***

1. Действия с матрицами

Для матриц и векторов в Scilab предусмотрены следующие операции:

+ — сложение;

- — вычитание;

’ — транспонирование ;

\* — матричное умножение ;

\* — умножение на число;

ˆ — возведение в степень ;

\ — левое деление ;

/ — правое деление ;

.\* — поэлементное умножение матриц;

.ˆ — поэлементное возведение в степень;

.\ — поэлементное левое деление;

./ — поэлементное правое деление

1. Функции для работы с матрицами:

***matrix(A, [,n,m])*** - преобразует матрицу А в матрицу другого размера;

***ones(m,n)*** — создает матрицу единиц из m строк и n столбцов;

***zeros(m,n)*** — создает нулевую матрицу из m строк и n столбцов;

***eye(m,n)*** — формирует единичную матрицу из m строк и n столбцов;

***rand(n1,n2,...nn[,fl])*** — формирует многомерную матрицу случайных чисел;

***sparse([i1 j1;i2 j2;...;in jn],[n1,n2,...,nn])*** — формирует разреженную матрицу;

***full(M)*** — вывод разреженной матрицы М в виде таблицы;

***hypermat(D[,V])*** — создание многомерной матрицы с размерностью, заданной вектором D и значениями элементов, хранящихся в векторе V (использование параметра V необязательно);

***diag(V[,k])*** — возвращает квадратную матрицу с элементами V на главной или на k-й диагонали;

***diag(A[,k])***, где A — ранее определенная матрица, в качестве результата выдает вектор-столбец, содержащий элементы главной или k-ой диагонали матрицы А;

***cat(n, A, B, [C, ...])*** — объединяет матрицы А и В или все входящие матрицы, при n=1 по строкам, при n=2 по столбцам; то же что [A; B] или [A, B];

***tril(A[,k])*** — формирует из матрицы А нижнюю треугольную матрицу, начиная с главной или с k-й диагонали;

***triu(A[,k])*** — формирует из матрицы А верхнюю треугольную матрицу, начиная с главной или с k-й диагонали;

***sort(X)*** — выполняет упорядочивание массива X; если X — матрица, сортировка выполняется по столбцам;

***size(V[,fl])*** — определяет размер массива;

***sum(X[,fl])*** — вычисляет сумму элементов массива X, имеет необязательный параметр fl;

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

***max(M[,fl])*** — вычисляет наибольший элемент в массиве M, имеет необязательный параметр fl;

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

***det(M)*** — вычисляет определитель квадратной матрицы М;

***rank(M[,tol])*** — вычисление ранга матрицы M с точностью tol;

1. Удаление элементов матрицы

Программа удаления m-го элемента из массива x(n):

disp(x);

n=length(x);

//Ввод номера удаляемого элемента.

m=input(’m=’);

//Сдвиг всех элементов, начиная с m-го на один влево.

for i=m:n-1

x(i)=x(i+1);

end;

//Удаление n-го элемента из массива.

x(:,n)=[];

//Уменьшение n на 1.

n=n-1;

//Вывод преобразованного массива.

disp(x);