Лабораторная работа 5. Геометрия и линейная алгебра.

Цель: знакомство с функциями пакетов алгебры и геометрии.

Основные команды **геометрии** находятся в пакетах geometry и geom3d. Небольшая часть команд в пакете Student[Precalculus].

# Линейная алгебра.

Небольшая часть команд **линейной алгебры** находится в стандартной библиотеке (Vector, Matrix и некоторые другие).

Основная часть команд линейной алгебры находятся в специальных пакетах:

* Пакет linalg был единственным пакетом линейной алгебры в старых версиях Maple, в современных версиях считается устаревшим.
* Пакет LinearAlgebra – основной пакет линейной алгебры для работы с матрицами и векторами, удобен для матричных вычислений.
* Пакет VectorCalculus – команды для векторного анализа.
* Подпакеты пакета Student: Student[LinearAlgebra], Student[VectorCalculus].

## Задание векторов:

1. С помощью угловых скобок:

<a1,a2,…,an> – вектор-столбец.

<a1|a2|…|an>– вектор-строка.

1. С помощью команды Vector(больше возможностей, см. help)

Vector([a1,a2,…,an]) – вектор-столбец.

Vector[row]([a1,a2,…,an])– вектор-строка.

## Задание матриц:

1. С помощью угловых скобок:

<<a11,…,an1>|<a12,…,an2>|….|<a1m,…,anm>> – матрица размера nxm, заданная по столбцам.

<<a11|…|a1m>,<a21|….|a2m>,…,<an1|…|anm>>–матрица размера nxm, заданная по строкам.

1. С помощью команды-конструктора Matrix(больше возможностей, см. help)

Matrix([[a11,a12,…,a1m],[a21,a22,…,a2m],…,[an1,an2,…,anm]])

Матрицы и векторы размером более 10 не выводятся на экран, просмотр элементов осуществляется в браузере (двойным нажатием на область вывода матрицы).

## Команды для задания матриц определенного вида:

ZeroMatrix, ZeroVector – нулевая матрица и нулевой вектор.

IdentityMatrix – единичная матрица.

DiagonalMatrix – диагональная матрица.

RandomMatrix – случайная матрица.

и другие.

## Доступ к элементам матрицы и вектора

осуществляется указанием номеров строки и столбца в квадратных скобках:

Например, если заданы вектор V и матрица M:

V[1] – первый элемент вектора

V[1..2] – элементы с первого по второй

V[-1] – последний элемент.

M[2,3] – элемент матрицы [2,3] (2 строка, 3 столбец)

M[1..2,2..3] – элементы матрицы: строки с 1 по 2, столбцы со 2 по 3.

## Размерность векторов и матриц

Dimension(A) – размерность матрицы или вектора

RowDimension(A) – число строк матрицы

ColumnDimension(A) – число столбцов матрицы

## Операции со столбцами и строками матрицы

* DeleteRow(A, L) – удаление строк матрицы A.
* DeleteColumn(A, L) – удаление столбцов матрицы A.
* Row(A, L) – извлечение строк матрицы **A**
* Column(A, L) – извлечение столбцов матрицы **A**

L – номера строк (столбцов), могут быть в виде интервала или списка.

* RowOperation(A, [ri,rj]) – перестановка строк **ri** и **rj** матрицы **A**
* ColumnOperation(A, [ci,cj])– перестановка столбцов **ci** и **cj** матрицы **A**
* RowOperation(A, [ri,rj],expr) – изменение строки **ri**: **ri:=ri+rj\*expr**, где **expr**– число или выражение (сложение строк).
* ColumnOperation(A, [ci,cj],expr) – изменение столбца **ci**: **ci:=ci+cj\*expr**, где **expr** – число или выражение (сложение столбцов).
* RowOperation(A, r,expr) – умножение строки **r** на выражение **expr**: **r:=r\*expr**
* ColumnOperation(A, c,expr) – умножение столбца **c** на выражение **expr**: **c:=c\*expr**
* Minor(A, i, j, out) – вычисление минора M(i,j) к элементу A[i,j] матрицы **A, out** задает тип результатов **output=matrix** или/и **output=determinant**(определитель).

## Действия с матрицами и векторами

A+B– сложение матриц или векторов **A** и **B.**

A\*c – умножение элементов матрицы (вектора) **A** на скаляр(число) **c.**

A.B– **матричное (некоммутативное) умножение** матриц и векторов.

A^n–возведение матрицы **A** в степень **n.**

A^(-1) – вычисление обратной матрицы или команда MatrixInverse(A).

A^(%T) – транспонированная матрица или команда Transpose(A).

Determinant(A)– вычисление определителя матрицы **A.**

Rank(A) – ранг матрицы.

Trace(A) – след матрицы (сумма диагональных элементов).

## Решение систем линейных уравнений

LinearSolve(A,B) – решение уравнения AX=B, где B – матрица или вектор правой части, X – матрица или вектор неизвестных.

NullSpace(A)– поиск базиса ядра матрицы, т.е. векторов {X: AX=0}, эквивалентно решению однородной системы уравнений.

GenerateEquations(A, v, B)– генерирование системы линейных уравнений Av=B, где A – матрица коэффициентов размера mxn, v –список неизвестных длины n, B – вектор правой части. GenerateEquations( A, [x,y,z], b )

GenerateMatrix(eqns, vars)– генерирование матрицы коэффициентов из списка (множества) уравнений **eqns** и списка (множества) неизвестных **vars.**GenerateMatrix( [eq1,e2,eq3], [x,y,z]).

# Геометрия

Функции пакета geometry можно разбить на несколько условных групп:

* Создание новых геометрических объектов: точки (point), прямые (line), отрезки (segment), направленные отрезки (dsegment), окружности (circle), эллипсы (ellipse), треугольники (triangle) и т.д. Кроме самостоятельных объектов, к этой категории можно отнести геометрические фигуры, являющиеся, по существу, одними из вышеперечисленных (например, отрезками или окружностями), но определяемые по отношению к другим. К таким объектам можно отнести высоты (altitude), медианы (median) и биссектрисы (bisector) треугольника, окружность, вписанную в треугольник (incircle) и описанную около треугольника (circumcircle), прямую Эйлера (EulerLine), касательные к окружности (TangentLine), точки Жергонна (GergonnePoint), Нагеля (NagelPoint) и многое другое.

Все функции имеют имя создаваемого объекта в качестве обязательного первого параметра. Также один объект можно задать несколькими способами. Например, задать треугольник можно координатами его вершин; уравнениями сторон; длинами сторон. В последнем случае треугольник задается безотносительно его расположения на координатной плоскости.

* Функции, возвращающие различные характеристики геометрических объектов: координаты (coordinates ), площадь (area), угол между прямыми (FindAngle ), тип объекта (form), способ задания треугольника (method), уравнение (Equation ) и т.д. Наиболее полную информацию о том или ином геометрическом объекте можно получить, используя функцию detail.
* Функции, которые осуществляют проверку инцидентности и взаимного расположения геометрических объектов. С помощью функций этой группы можно проверить принадлежность трех точек одной прямой (AreCollinear), четырех точек одной окружности (AreConcyclic), ориентацию треугольника (IsRightTriangle) и т.д.
* Функции геометрических преобразований. Сюда входят параллельный перенос (translation), вращение (rotation), центральная и осевая симметрии (reflection), гомотетия (dilatation), инверсия (inversion) и некоторые другие преобразования. К этой же группе можно отнести и ортогональное проектирование (projection).
* Команда draw, позволяющая изображать геометрические объекты. Единственным обязательным параметром команды является список изображаемых объектов. Кроме того имеется много опциональных параметров. Задаваемые ими опции могут действовать как глобально (на все элементы списка вывода), так и локально. В последнем случае соответствующие опции указываются в круглых скобках непосредственно после элементов списка изображаемых объектов.

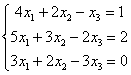
Функции пакета geom3d имеют аналогичные группы, можно посмотреть в справке.

# Задание 1. Линейная алгебра.

* Задать вектор столбец  с помощью команды Vector
* Задать матрицу  используя угловые скобки.
* Используя угловые скобки соединить их, получить матрицу: 
* Задать случайную матрицу (специальная команда) *А* размера 5х5.
* Заменить в ней центральные элементы матрицей *М*:



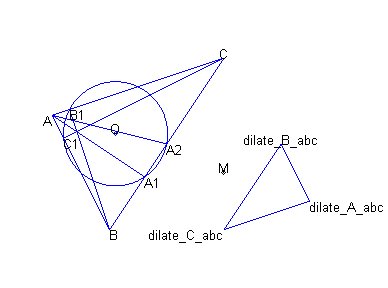
* Для данной матрицы *A* найдите , , .
* Найти для матрицы *А* определитель, ранг, след, транспонированную матрицу.
* Задать систему (задать три уравнения как список)



* Сгенерировать с нее матрицу.
* Решить систему в матричном виде AX=B
* Решить систему методом Гаусса: командой RowOperation привести матрицу к треугольному виду, потом обратным ходом найти неизвестные.
* Решить систему методом обратной матрицы, т.е. найти 

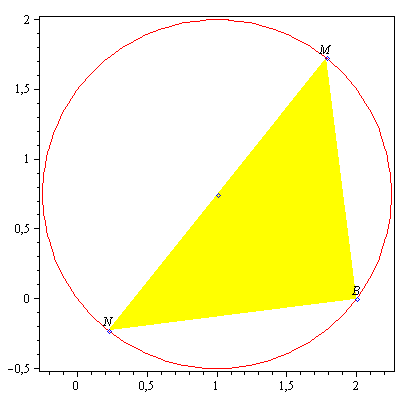
# Задание 2. Геометрия.

* Задать три точки с именами *А, В, С* (координаты любые).
* Задать прямую AB и найдем ее уравнение.
* Задать треугольник ABC и найти его площадь.
* Задать высоты ha, hb и hc треугольника ABC и обозначить их основания через A1, B1 и C1 соответственно.
* Провести окружность **c** через основания высот и обозначим ее центр через O.
* Провести медиану ma из вершины A и убедится, что ее основание лежит на одной окружности с основаниями высот.
* Построить треугольник abc, гомотетичный треугольнику ABC с центром гомотетии в точке M(5;2) и коэффициентом -1/2, и вывести на экран информацию об этом треугольнике.
* Изобразить на одном чертеже все построенные объекты. Дожно получиться:



# Задание 3. Геометрия.

* Задать окружность **c1**, проходящую через три заданные точки **А(0,0)**, **В(2,0)** и **С(1,2)**. Найти координаты центра этой окружности.
* Далее найти радиус окружности.
* Найти уравнение окружности.
* С помощью функции detail получить детальное описание окружности.
* Провести прямую через центр окружности и точку (1,1).
* Найти точки пересечения этой прямой и окружности – **N**, **M**
* Найти координаты точек – **N**, **M**
* Проверить лежит ли точка **М** на окружности (IsOnCircle).
* Построить треугольник через точки N,M,B.
* Нарисовать:



# Задание 4. Геометрия в пространстве.

* Задать вершины тетраэдра:

А(1,2,-3), B(5,-2,-1), C(-2,-3,-5), D(4,2,3)

* Найти длину ребра АВ
* Написать уравнение плоскости, содержащей грань АВС
* Найти площадь грани АВС
* Написать уравнение прямой АС
* Найти проекцию вершины D на грань АВС.
* Написать уравнение высоты, опущенной из вершины D
* Найти угол между гранями АСВ и АСD
* Найти угол между ребрами DС иDА
* Найти объем тетраэдра АВСD
* Найти точку пересечения медиан треугольника АВС
* Найти угол между ребром АD и гранью АВС
* Нарисовать тетраэдр и высоту и точку (точку плохо видно, не изменяется ее цвет, может зависит от версии Maple).

