

## 1. Введение

Maxima — это мощная система для символических и численных вычислений, которая предоставляет широкий спектр возможностей для работы с матрицами. С помощью Maxima можно создавать матрицы, выполнять базовые и сложные операции, такие как сложение, умножение, нахождение определителя, обратной матрицы и т.д. Этот справочник содержит основные формулы и команды, которые помогут эффективно использовать матрицы в Maxima.

---

## 2. Ввод и вывод матрицы

### 1. Создание матрицы:

Для создания матрицы используйте функцию *matrix*. Например, для создания матрицы A размера 3×3:

*A: matrix([1, 2, 3], [4, 5, 6], [7, 8, 9]);*

Это создаст матрицу:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

### 2. Вывод матрицы:

Чтобы вывести матрицу, введите её имя:

*A;*

---

## 3. Действия с матрицами

### 1. Сложение матриц:

Если имеются две матрицы одинакового размера A и B, их можно сложить:

*C: A + B;*

### 2. Вычитание матриц:

Вычитание выполняется аналогично сложению:

*D: A - B;*

### 3. Умножение матриц:

Для умножения матриц A и B используйте оператор “.”

*E: A . B;*

### 4. Умножение матрицы на число:

Например, чтобы умножить матрицу A на число 3:

*F: 3 \* A;*

#### 5. Транспонирование матрицы:

Для нахождения транспонированной матрицы используйте функцию `transpose`:

*At: transpose(A);*

#### 6. Нахождение определителя:

Для вычисления определителя матрицы A:

*detA: determinant(A);*

---

### 4. Удаление элементов матрицы

#### 1. Удаление строки или столбца:

Чтобы удалить строку *i* и столбец *j*, используйте функцию `submatrix`. Например, удаление второй строки и третьего столбца:

*submatrix(A, [2], [3]);*

---

### 5. Другие операции

#### 1. Обратная матрица:

Если матрица A обратима, её обратную матрицу можно найти с помощью функции `invert`:

*invA: invert(A);*

#### 2. След матрицы:

Для вычисления следа матрицы (суммы элементов главной диагонали):

*traceA: trace(A);*

#### 3. Собственные значения матрицы:

Для нахождения собственных значений матрицы A:

*eigenvalues(A);*

#### 4. Возведение в степень:

Возведение матрицы A в квадрат:

*A2: A ^2;*

---

### 6. Практические примеры

#### 1. Решение систем линейных уравнений:

Пусть дана система  $AX=B$ . Чтобы найти решение:

*solve\_by\_lu(A, B);*

#### 2. Пример вычисления определителя:

Для матрицы A:

*determinant(matrix([1, 2], [3, 4]));*

#### 3. Нахождение собственных значений и векторов:

Для нахождения собственных значений и векторов используйте:

*eigenvectors(A);*

---

## 7. Заключение

Maxima предоставляет простой и удобный интерфейс для работы с матрицами, что делает её незаменимым инструментом для изучения линейной алгебры и решения инженерных задач. Этот справочник поможет вам быстро освоить базовые команды и начать их применять в своей работе.

---