**МАИ**

**Лабораторная работа №10**

Функции

Вариант №14

Факультет робототехнических и интеллектуальных систем

Кафедра «Системы приводов летательных аппаратов»

**Выполнил:**

Студент группы М7О-114БВ-24

Фельдман Лев Борисович

**Проверил:**Доцент Кафедры 702 Козлова Н.М.

Ассистент Кафедры 702 Милославский Я.Г.

Москва 2025

**Цель: Закрепление навыков по работе с функциями.**

**Задание 3. Создать функцию, вычисляющую расстояние от точки логарифмической спирали до полюса по формуле r = a \* e ^ (b \* pi). Входными параметрами функции являются угол pi и коэффициенты a и b, выходным параметром расстояние r.**

function r = distance\_spiral(theta, a, b)

% distance\_spiral – функция вычисляющая расстояние от точки до полюса.

% Формула: r = a \* exp(b \* theta)

%

% Входные параметры:

% theta - угол (числовой скаляр)

% a - коэффициент a (числовой скаляр)

% b - коэффициент b (числовой скаляр)

%

% Выходной параметр:

% r - расстояние

% Проверка числа входных параметров

if nargin ~= 3

error('Ошибка: Функция требует три входных параметра: угол, a и b.');

end

% Проверка типов входных параметров (должны быть числовыми скалярами)

if ~isscalar(theta) || ~isnumeric(theta) || ...

~isscalar(a) || ~isnumeric(a) || ...

~isscalar(b) || ~isnumeric(b)

error('Ошибка: Все входные параметры должны быть числовыми скалярами.');

end

% Вычисление расстояния по формуле

r = a \* exp(b \* theta);

end

% Пример вызова:

r = distance\_spiral(pi/4, 2, 0.5);

**>>** **r =**

**2.9619**

**Задание 4. Разработать функцию, при выполнении которой происходит построение конуса второго порядка по формуле (x^2/a^2) + (y^2/b^2) + (z^2/c^2) = 0. Функция принимает два входных параметра. Первый параметр содержит значения величин a, b и c, а также высоты конуса һ, второй дополнительный параметр значения углов вращения в градусах вокруг соответствующих осей. Значение дополнительного параметра по умолчанию равно [45, 0, 0].**

function plot\_cone(params, varargin)

% plot\_cone - строит график конуса второго порядка с возможностью поворота.

%

% Синтаксис:

% plot\_cone(params)

% plot\_cone(params, rotAngles)

%

% Входные параметры:

% params - числовой вектор [a, b, c, h], где a, b, c - коэффициенты,

% h - высота конуса.

%

% Дополнительный параметр:

% rotAngles - вектор [alpha, beta, gamma] углов вращения (в градусах)

% вокруг осей X, Y и Z соответственно. По умолчанию: [45, 0, 0].

%

% Уравнение конуса: (x^2/a^2) + (y^2/b^2) - (z^2/c^2) = 0

% Проверка первого параметра

if ~isnumeric(params) || numel(params) ~= 4

error('Ошибка: Первый параметр должен быть числовым вектором [a, b, c, h].');

end

% Определение углов вращения

if nargin == 2

rotAngles = varargin{1};

if ~isnumeric(rotAngles) || numel(rotAngles) ~= 3

error('Ошибка: Второй параметр должен быть числовым вектором с 3 элементами (углы вращения).');

end

else

rotAngles = [45, 0, 0];

end

% Извлечение параметров конуса

a = params(1); b = params(2); c = params(3); h = params(4);

% Параметризация конуса.

% Для конуса с вершиной в начале координат можно задать:

% x = a\*(z/c)\*cos(theta), y = b\*(z/c)\*sin(theta), z = z,

% где z изменяется от 0 до h, а theta от 0 до 2\*pi.

theta = linspace(0, 2\*pi, 50);

z = linspace(0, h, 50);

[Theta, Z] = meshgrid(theta, z);

X = a \* (Z / c) .\* cos(Theta);

Y = b \* (Z / c) .\* sin(Theta);

% Применение поворота.

% Преобразуем координаты в виде набора точек для применения матричного умножения.

pts = [X(:)'; Y(:)'; Z(:)'];

% Построение матриц вращения.

alpha = deg2rad(rotAngles(1));

beta = deg2rad(rotAngles(2));

gamma = deg2rad(rotAngles(3));

Rx = [1 0 0; 0 cos(alpha) -sin(alpha); 0 sin(alpha) cos(alpha)];

Ry = [cos(beta) 0 sin(beta); 0 1 0; -sin(beta) 0 cos(beta)];

Rz = [cos(gamma) -sin(gamma) 0; sin(gamma) cos(gamma) 0; 0 0 1];

% Комбинированная матрица вращения (порядок: Rx -> Ry -> Rz)

R = Rz \* Ry \* Rx;

pts\_rot = R \* pts;

% Преобразуем координаты обратно в матричный формат

X\_rot = reshape(pts\_rot(1,:), size(X));

Y\_rot = reshape(pts\_rot(2,:), size(Y));

Z\_rot = reshape(pts\_rot(3,:), size(Z));

% Построение поверхности

figure;

surf(X\_rot, Y\_rot, Z\_rot);

xlabel('X'); ylabel('Y'); zlabel('Z');

title('Построение конуса второго порядка');

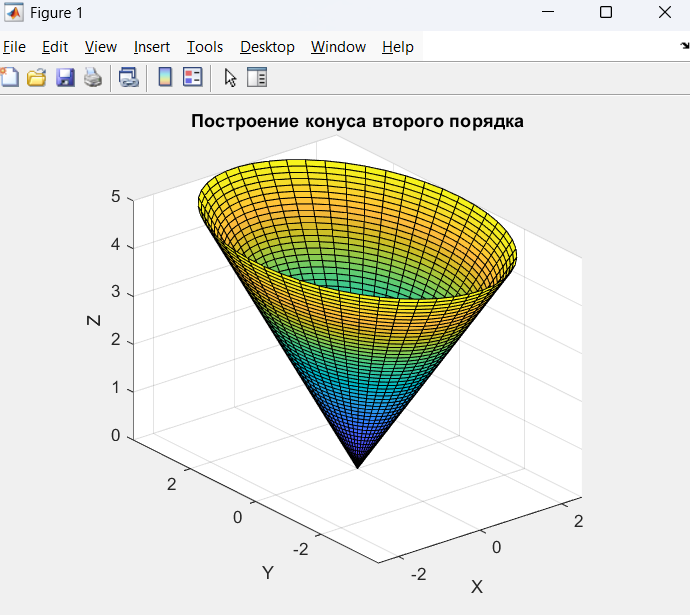
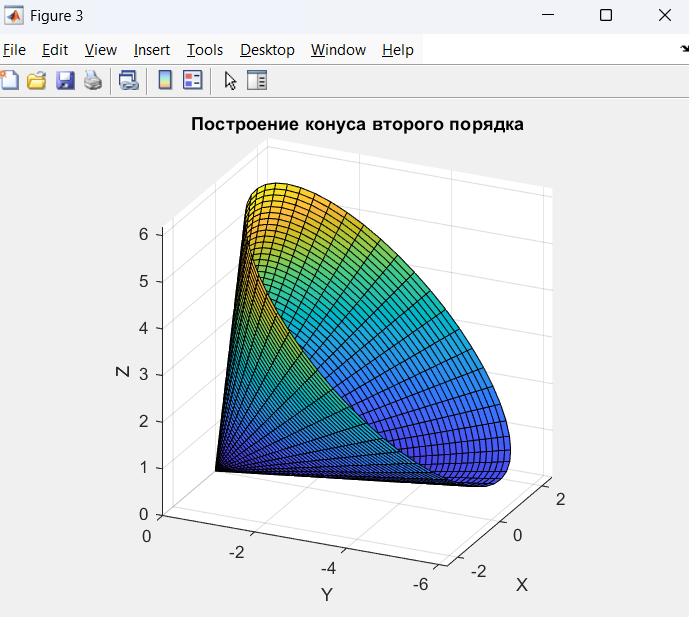
axis equal;

grid on;

end

plot\_cone([2, 3, 4, 5])

plot\_cone([2, 3, 4, 5], [30, 15, 0]);

>>   
>> 

Задание 5.

5. Создать функцию, которая выполняет построение конуса второго по-рядка. Входные параметры в функцию определяют внешний вид графи-ческого окна (текст в строке заголовка, размеры и т.д.).

function plot\_custom\_cone(varargin)

% plot\_custom\_cone - строит конус второго порядка с произвольными параметрами.

%

% Использование:

% plot\_custom\_cone('a',2, 'b',3, 'c',4, 'h',5, 'Color', 'red', 'Title', 'Мой конус')

%

% Допустимые параметры:

% a - коэффициент a (по умолчанию 1)

% b - коэффициент b (по умолчанию 1)

% c - коэффициент c (по умолчанию 1)

% h - высота конуса (по умолчанию 1)

% Color - цвет конуса (по умолчанию 'blue')

% Title - заголовок окна (по умолчанию 'Построение конуса')

% Position - позиция окна [left, bottom, width, height] (по умолчанию [100, 100, 800, 600])

%

% Если параметр не указан, используется значение по умолчанию.

% Значения по умолчанию

params.a = 1;

params.b = 1;

params.c = 1;

params.h = 1;

params.Color = 'blue';

params.Title = 'Построение конуса';

params.Position = [100, 100, 800, 600];

% Проверка, что число аргументов чётное (пара имя-значение)

if mod(length(varargin),2) ~= 0

error('Ошибка: Параметры должны передаваться парами "имя-значение".');

end

% Обработка входных параметров

for i = 1:2:length(varargin)

fieldName = varargin{i};

fieldValue = varargin{i+1};

if ischar(fieldName)

% Если поле существует в структуре параметров, обновляем его

if isfield(params, fieldName)

params.(fieldName) = fieldValue;

else

warning('Неизвестный параметр: %s. Он будет проигнорирован.', fieldName);

end

else

error('Ошибка: Имена параметров должны быть строками.');

end

end

% Создание графического окна

fig = figure('Name', params.Title, 'Position', params.Position);

% Построение конуса (параметризация)

theta = linspace(0, 2\*pi, 50);

z = linspace(0, params.h, 50);

[Theta, Z] = meshgrid(theta, z);

% Параметризация конуса:

% x = a\*(z/c)\*cos(theta), y = b\*(z/c)\*sin(theta)

X = params.a \* (Z / params.c) .\* cos(Theta);

Y = params.b \* (Z / params.c) .\* sin(Theta);

% Построение поверхности конуса с заданным цветом

surf(X, Y, Z, 'FaceColor', params.Color);

xlabel('X'); ylabel('Y'); zlabel('Z');

title(params.Title);

axis equal;

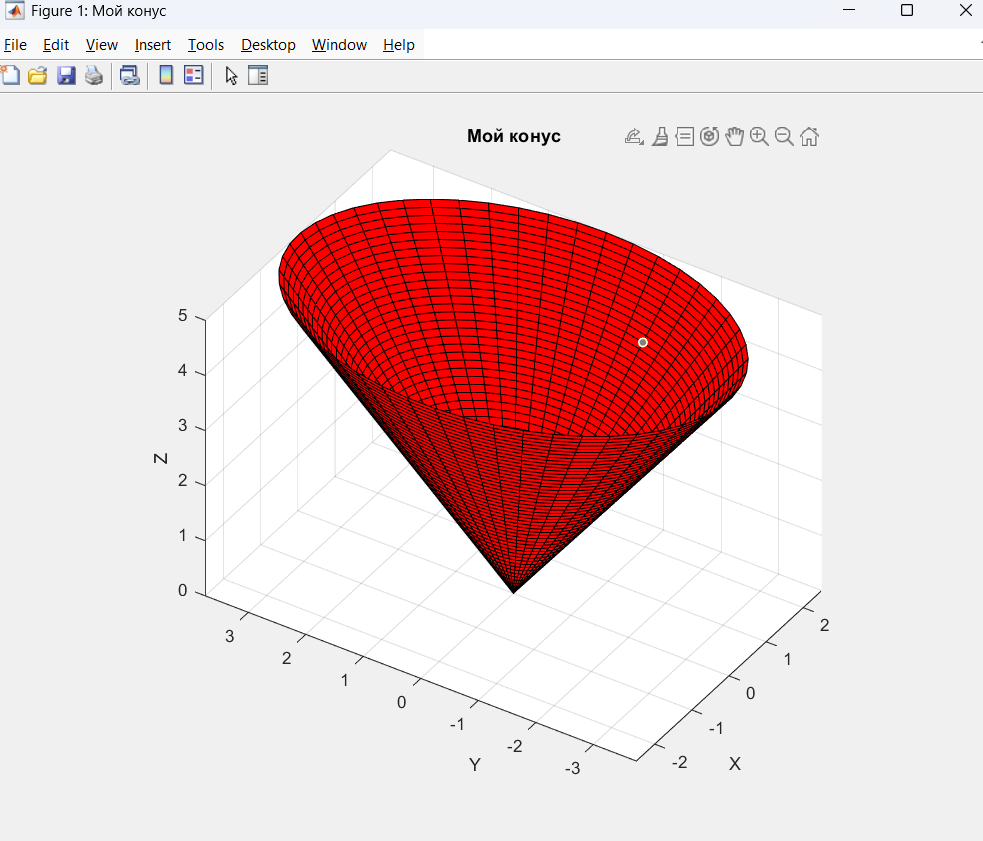
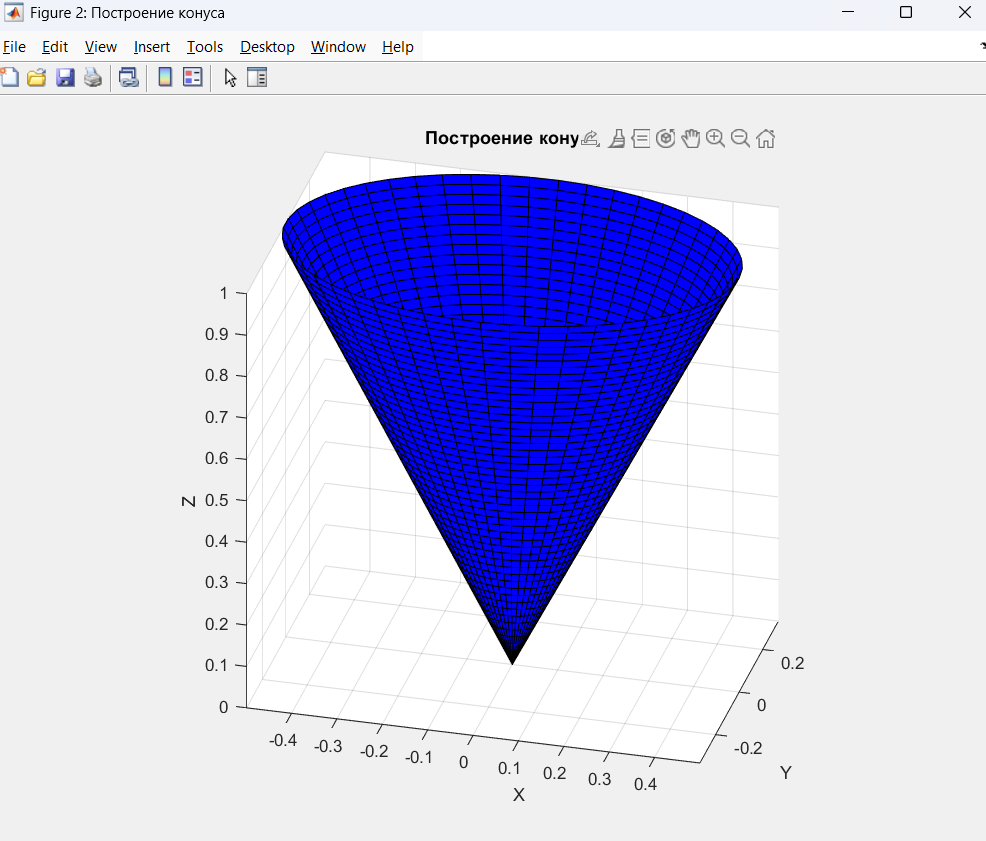
grid on;

end

plot\_custom\_cone('Title', 'Мой конус', 'Position', [200 200 600 500], 'Color', 'b')

plot\_custom\_cone('a', 3, 'b', 2, 'c', 6, 'Color', 'blue', 'Rotation', [0,0,0]);

>>

**** ****