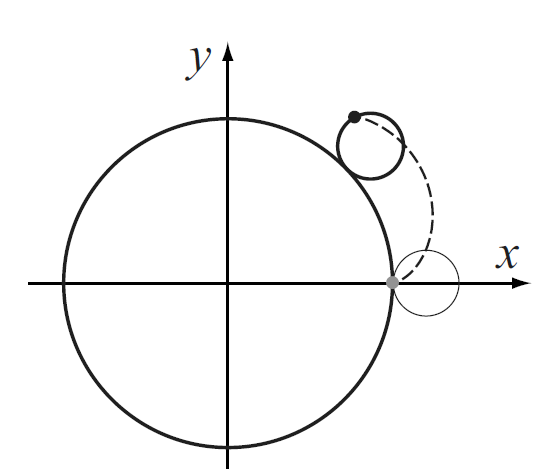
1. , bunda funksiyaning -10 ≤ *x* ≤ 10 va -10 ≤ *y* ≤ 10 sohalardagi 3D sirt grafigini tuzing. **(10 ball)**
2. Epitsikloid - bu boshqa qo'zg'almas doira bo'ylab aylanib yuruvchi aylanadagi nuqtaning traektoriyasi sifatida olingan egri chiziq (qisman rasmda ko'rsatilgan). Tsikloidning parametrik tenglamasi quyidagi formulalar bilan berilgan:

0 uchun sikloid grafigini tuzing. Grafiklarni shunday

formatlangki, ikkala o‘q ham –20 dan 20 gacha bo'lsin. O’qlarni belgilang va grafikga sarlavha bering. **(10 ball)**

1. N yil davomida umumiy F summasini to'plash uchun yillik foiz stavkasi r bo'lgan hisobvaraqqa o'tkazilishi kerak bo'lgan depozit hisobvarag'iga oylik P omonatini quyidagi formula yordamida hisoblash mumkin:

Yillik foiz stavkasi 4,35% bo'lsa, 5, 6, 7, 8, 9 va 10 yil davomida 100 000 dollarni to'plash uchun oylik omonat miqdorini hisoblang. Natijalarni ikkita ustunli jadvalda ko'rsating, bu erda birinchi ustun yillar soni, ikkinchi ustun oylik omonat hisoblanadi. **(10 ball)**

1. n ta musbat *x1, x2, …, xn* sonlar to'plamining o'rtacha geometrik GM qiymati quyidagicha aniqlanadi:

Sonlar to’plamining o’rtacha geometrik qiymatini hisoblaydigan maxsus funksiyani tuzing. Funksiya nomi va argumentlari uchun **GM=Geomean(x)** dan foydalaning, bu yerda kirish argumenti x sonlardan iborat vektor (istalgan uzunlikdagi), va GM – chiqish argumenti – sonlar to’plamining o’rtacha geometrik qiymati. Geometrik o’rtacha darajalarning o’rtacha qiymatini hisoblash uchun foydalidir. Quyidagi jadvalda AQSHda 1978 – 1987 yillardagi inflyatsiya darajalari keltirilgan (7.6% li inflyatsiya jadvalda 1.076 sifatida ko’rsatilgan). O'n yil davomida o'rtacha inflyatsiyani hisoblash uchun **Geomean** maxsus funksiyasidan foydalaning. **(15 ball)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Yil* | 1978 | 1979 | 1980 | 1981 | 1982 | 1983 | 1984 | 1985 | 1986 | 1987 |
| *Inflyatsiya darajasi* | 1.076 | 1.113 | 1.135 | 1.103 | 1.062 | 1.032 | 1.043 | 1.036 | 1.019 | 1.036 |

1. *ax2 + bx + c = 0* kvadrat tenglamaning haqiqiy ildizlarini aniqlaydigan dasturni script faylida yozing. Uning ishlashi davomida fayl foydalanuvchidan *a, b* va *c* konstantalarining qiymatlarini kiritishni taklif qilishi kerak. Tenglamaning ildizlarini hisoblash uchun dastur quyidagi formula yordamida diskriminant *D* ni hisoblab chiqadi:

Agar D > 0, bo'lsa, u holda dasturda "Tenglama ikkita ildizga ega" xabarlari paydo bo'ladi va keyingi qatorda ekranda ildizlar ko'rsatiladi.

Agar D = 0, bo'lsa, u holda dastur "Tenglama bitta ildizga ega" xabarlarini ko'rsatadi va ildiz keyingi qatorda ko'rsatiladi.

Agar D < 0, bo'lsa, u holda dastur "Tenglamaning ildizlari yo'q" xabarlarini ko'rsatadi.

Quyidagi uchta tenglamaning yechimini olish uchun ushbu skript faylini buyruqlar oynasida uch marta ishga tushiring: **(25 ball)**

a) 3x2 + 6x + 3 = 0 b) -3x2 + 4x – 6 = 0

1-savol

% Define the range of x values from -10 to 10

x = linspace(-10, 10, 100);

% Define the range of y values from -10 to 10

y = linspace(-10, 10, 100);

% Create a grid of x and y values

[X, Y] = meshgrid(x, y);

% Calculate R

R = sqrt(X.^2 + Y.^2) / 2;

% Calculate Z, handling the case where R is 0 to avoid division by zero

Z = sin(R) ./ R;

Z(R == 0) = 1; % Set Z=1 where R=0 (limit of sin(R)/R as R->0)

% Create a new figure window

figure;

% Use the surf function to create the 3D surface plot

surf(X, Y, Z);

% Add labels to the x, y, and z axes

xlabel('x');

ylabel('y');

zlabel('Z');

% Add a title to the plot

title('3D Surface Plot of Z = sin(R)/R, where R = sqrt(x^2 + y^2)/2');

% Add a colorbar to show the mapping of Z values to colors

colorbar;

% Optional: Add a grid for better visualization

grid on;

2-savol

% Define the range of the parameter t

t = linspace(0, 4\*pi, 500); % Use 500 points for a smooth curve

% Calculate the x and y coordinates using the parametric equations

x = 13\*cos(t) - 2\*cos(6.5\*t);

y = 13\*sin(t) - 2\*sin(6.5\*t);

% Create a new figure window

figure;

% Plot the epicycloid

plot(x, y);

% Set the x-axis limits

xlim([-20 20]);

% Set the y-axis limits

ylim([-20 20]);

% Add labels to the x and y axes

xlabel('x');

ylabel('y');

% Add a title to the plot

title('Epicycloid');

% Ensure the aspect ratio is equal so the shape is not distorted

axis equal;

% Add a grid for better visualization (optional)

grid on;

3-savol

% Given values

F = 100000; % Future value

r\_annual = 4.35; % Annual interest rate

r = r\_annual / 100;

years = 5:10;

% Display header

disp('-------------------------');

disp(' Years | Monthly Deposit ');

disp('-------------------------');

% Calculate and display monthly deposit for each year

for N = years

monthly\_rate = r / 12;

number\_of\_periods = 12 \* N;

P = F \* monthly\_rate / ((1 + monthly\_rate)^number\_of\_periods - 1);

fprintf(' %2d | $%8.2f \n', N, P);

end

% Display footer

disp('-------------------------');

4-savol

function GM = Geomean(x)

% Geomean: Calculates the geometric mean of a vector of positive numbers.

% GM = Geomean(x) returns the geometric mean of the elements in the vector x.

n = length(x);

product\_of\_elements = prod(x);

GM = product\_of\_elements^(1/n);

end

% US inflation rates from 1978 to 1987 (from the table)

inflation\_rates = [1.076, 1.113, 1.135, 1.103, 1.062, 1.032, 1.043, 1.036, 1.019, 1.036];

% Calculate the geometric mean of the inflation rates using the Geomean function

average\_inflation = Geomean(inflation\_rates);

% Display the result

fprintf('The average inflation rate in the US from 1978 to 1987 is: %.4f\n', average\_inflation);

5-savol

% Script to find the real roots of a quadratic equation ax^2 + bx + c = 0

% Prompt the user to enter the coefficients

a = input('Enter the coefficient a: ');

b = input('Enter the coefficient b: ');

c = input('Enter the coefficient c: ');

% Calculate the discriminant

D = b^2 - 4\*a\*c;

% Check the value of the discriminant to determine the nature of the roots

if D > 0

% Two distinct real roots

root1 = (-b + sqrt(D)) / (2\*a);

root2 = (-b - sqrt(D)) / (2\*a);

disp('Tenglama ikkita ildizga ega:');

fprintf('Ildiz 1: %.2f\n', root1);

fprintf('Ildiz 2: %.2f\n', root2);

elseif D == 0

% One real root (repeated)

root = -b / (2\*a);

disp('Tenglama bitta ildizga ega:');

fprintf('Ildiz: %.2f\n', root);

else

% No real roots (discriminant is negative)

disp('Tenglamaning haqiqiy ildizlari yo''q.');

end