1. funksiyaning -2 ≤ *x* ≤ 2 va -2 ≤ *y* ≤ 2 sohalardagi 3D sirt grafigini tuzing. **(10 ball)**
2. funksiyani -4 ≤ *x* ≤ 9 sohadagi grafigini chizing. Funksiya ikkita nuqtada vertikal asimptotaga ega ekanligini unutmang (*x1* = -2 va *x2* = 5 da). Funksiya grafigini, x o'zgaruvchisining diapazonini uch qismga bo'lish orqali chizing: birinchisi -4 dan chap asimptotagacha, ikkinchi qismi ushbu ikki asimptota o'rtasida va uchinchisi o'ng asimptotadan 9 gacha. Y o'qi diapazonini -20 dan 20 gacha o'rnating. O’qlarni belgilang va grafikga sarlavha bering. **(10 ball)**
3. 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 4,85% bo'lsa, 10, 11, 12, 13, 14 va 15 yil davomida 200 000 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 aksiyaning o'rtacha daromadini hisoblash uchun foydalidir. Quyidagi jadvalda so’ngi o’n yil ichida IBM aksiyalarining daromadi ko’rsatilgan (16% daromad 1,16 ni bildiradi). Quyidagi aksiyalarning o’rtacha daromadligini hisoblash uchun **Geomean** funksiyasidan foydalaning. **(15 ball)**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Yil* | 1997 | 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 |
| *Daromadi* | 1.38 | 1.76 | 1.17 | 0.79 | 1.42 | 0.64 | 1.2 | 1.06 | 0.83 | 1.18 |

1. Foydalanuvchiga ixtiyoriy uzunlikdagi butun sonlardan iborat vektorni kiritishni so‘raydigan dastur tuzing. Keyin dastur elementlarning sonini, musbat elementlarning sonini va 3 ga bo'linadigan manfiy elementlarning sonini sanaydi. Dastur kiritilgan vektor va natijalarni jumla ko'rinishida, ya'ni: "Vektorda XX element mavjud. XX elementlar musbat va XX elementlar manfiy va 3 ga bo'linadi." deb ko'rsatishi kerak, bu yerda XX – elementlarning mos keladigan sonini bildiradi. Dasturni ishga tushiring va dastur foydalanuvchidan vektor kiritishni so'raganda, ***randi([-20 20],1,16)*** kiriting. Bu -20 dan 20 gacha bo'lgan tasodifiy butun sonli 16 elementli vektorni yaratadi. **(25 ball)**

1-savol

% Define the range of x values from -2 to 2

x = linspace(-2, 2, 50);

% Define the range of y values from -2 to 2

y = linspace(-2, 2, 50);

% Create a grid of x and y values

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

% Calculate the corresponding z values using the function z = 0.5 \* |x| + 0.5 \* |y|

Z = 0.5 \* abs(X) + 0.5 \* abs(Y);

% 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 = 0.5|x| + 0.5|y|');

% 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 function

f = @(x) (x.^2 + 3\*x - 5) ./ (x.^2 - 3\*x - 10);

% Define the x ranges, avoiding the asymptotes at -2 and 5

x1 = linspace(-4, -2.1, 100);

x2 = linspace(-1.9, 4.9, 100);

x3 = linspace(5.1, 9, 100);

% Calculate the y values for each x range

y1 = f(x1);

y2 = f(x2);

y3 = f(x3);

% Create a new figure

figure;

% Plot the function in three parts

plot(x1, y1, 'b');

hold on;

plot(x2, y2, 'r');

plot(x3, y3, 'g');

% Set the x-axis range

xlim([-4 9]);

% Set the y-axis range

ylim([-20 20]);

% Add labels to the axes

xlabel('x');

ylabel('f(x)');

% Add a title to the plot

title('Graph of f(x) = (x^2 + 3x - 5) / (x^2 - 3x - 10)');

% Add a legend

legend('Part 1 (-4 to -2)', 'Part 2 (-2 to 5)', 'Part 3 (5 to 9)');

% Optional: Add vertical lines to mark the asymptotes

plot([-2 -2], ylim, 'k--');

plot([5 5], ylim, 'k--');

hold off;

3-savol

% Given values

F = 200000; % Future value

r\_annual = 4.85; % Annual interest rate

r = r\_annual / 100;

years = 10:15;

% 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

% IBM stock returns for the last ten years (from the table)

returns = [1.38, 1.76, 1.17, 0.79, 1.42, 0.64, 1.20, 1.06, 0.83, 1.18];

% Calculate the geometric mean of the returns using the Geomean function

average\_return = Geomean(returns);

% Display the result

fprintf('The average return of IBM stock over the last ten years is: %.4f\n', average\_return);

5-savol

% Program to count elements in a user-defined vector

% Prompt the user to enter a vector of integers

user\_vector = input('Enter a vector of integers: ');

% Calculate the total number of elements

total\_elements = length(user\_vector);

% Initialize counters for positive and negative divisible by 3 elements

positive\_count = 0;

negative\_divisible\_by\_3\_count = 0;

% Iterate through the vector to count elements

for element = user\_vector

if element > 0

positive\_count = positive\_count + 1;

elseif element < 0 && mod(element, 3) == 0

negative\_divisible\_by\_3\_count = negative\_divisible\_by\_3\_count + 1;

end

end

% Display the results in the specified sentence format

fprintf('Vektorda %d ta element mavjud. %d ta elementlar musbat va %d ta elementlar manfiy va 3 ga bo''linadi.\n', ...

total\_elements, positive\_count, negative\_divisible\_by\_3\_count);

% --- Instructions for running the program as requested ---

% 1. Save this code as a .m file (e.g., count\_elements.m).

% 2. Open MATLAB.

% 3. Navigate to the directory where you saved the file.

% 4. Type `count\_elements` in the MATLAB command window and press Enter.

% 5. When the program prompts "Enter a vector of integers: ", type the following and press Enter:

% `randi([-20 20], 1, 16)`

% This will generate a 1x16 vector of random integers between -20 and 20.

% 6. The program will then display the results based on the generated vector.