1. funksiyaning -3 ≤ *x* ≤ 3 va -3 ≤ *y* ≤ 3 sohalardagi 3D sirt grafigini tuzing. **(10 ball)**
2. funksiyani -4 ≤ *x* ≤ 4 sohadagi grafigini chizing. Funksiya ikkita nuqtada vertikal asimptotaga ega ekanligini unutmang (*x1* = -1 va *x2* = 1 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 4 gacha. Y o’qi diapazonini -15 dan 15 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 4,35% 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. O'rtacha baho (GPA) ni 0 dan 4 gacha bo'lgan shakl bo'yicha hisoblaydigan maxsus funksiyani yozing, bunda *A* = 4, *A-* = 3.7, *B+* = 3.3, *B* = 3, *B-* = 2.7, *C+* = 2.3, *C* = 2, *C-* = 1.7, *D+* = 1.3, *D* = 1, va *E* = 0. Funksiya nomi va argumentlari uchun **av = GPA(g,h)** dan foydalaning. Kirish argumenti **g** – elementlari fanlardagi baholarning raqamli qiymatlari bo'lgan vektor. Kirish argumenti **h** – mos ravishda fanlar bo'yicha tegishli kredit soatlari bo'lgan vektor. Chiqish argumenti **av** esa hisoblangan GPA qiymati (*GPA = (Kredit soatlariga ko'paytirilgan ballar yig'indisi) / (kredit soatlari yig'indisi)*). Quyidagi natijalar bilan talabaning GPA sini hisoblash uchun funksiyadan foydalaning: **(15 ball)**:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Baholar* | *A-* | *B* | *B+* | *C* | *E* | *A* | *D+* | *A* |
| *Kredit soatlari* | 4 | 3 | 3 | 2 | 3 | 4 | 3 | 3 |

1. Pifagor teoremasi *a2 + b2 = c2* ekanligini ta’kidlaydi. Pifagor teoremasini qanoatlantiradigan 50 dan kichik yoki unga teng musbat butun sonlarning *a*, *b* va *c* uchliklarining barcha kombinatsiyalarini topadigan MATLAB dasturini yozing. Natijalarni uchta ustunli jadvalda ko'rsating, har bir satr bir uchlik sonlarga mos keladi. Jadvalning dastlabki uchta qatori: **(25 ball)**:

1-savol

% 1. Define the range of x and y values

x = linspace(-3, 3, 100); % Create 100 points between -3 and 3 for x

y = linspace(-3, 3, 100); % Create 100 points between -3 and 3 for y

% 2. Create a grid of x and y values using meshgrid

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

% 3. Calculate the z values using the given function

Z = (Y.^2 / 4) - 2 \* sin(1.5 \* X);

% 4. Create the 3D surface plot

figure; % Open a new figure window

surf(X, Y, Z); % Create the surface plot

% 5. Add labels and title for clarity

xlabel('x'); % Label the x-axis

ylabel('y'); % Label the y-axis

zlabel('z'); % Label the z-axis

title('3D Surface Plot of z = (y^2 / 4) - 2sin(1.5x)'); % Add a title

% 6. Optional: Add a colorbar to show the z values

colorbar;

% 7. Optional: Adjust the view angle for better visualization

view(30, 45); % Adjust the elevation and azimuth angles

2-savol

% 1. Define the x ranges for the three parts of the function

x1 = linspace(-4, -1.01, 100); % From -4 to just before -1

x2 = linspace(-0.99, 0.99, 100); % Between -1 and 1

x3 = linspace(1.01, 4, 100); % From just after 1 to 4

% 2. Calculate the corresponding y values for each part

y1 = x1 + 1./(x1.^2 - 1);

y2 = x2 + 1./(x2.^2 - 1);

y3 = x3 + 1./(x3.^2 - 1);

% 3. Create the plot

figure; % Open a new figure window

plot(x1, y1, 'b-', 'LineWidth', 1.5); % Plot the first part in blue

hold on; % Keep the current plot and add more to it

plot(x2, y2, 'r-', 'LineWidth', 1.5); % Plot the second part in red

plot(x3, y3, 'g-', 'LineWidth', 1.5); % Plot the third part in green

% 4. Set the y-axis limits

ylim([-15, 15]);

% 5. Add labels and title

xlabel('x');

ylabel('f(x)');

title('Graph of f(x) = x + 1/(x^2 - 1)');

% 6. Add a legend (optional)

legend('x < -1', '-1 < x < 1', 'x > 1');

% 7. Add vertical asymptote lines (optional)

plot([-1, -1], ylim, 'k--', 'LineWidth', 1); % Black dashed line at x = -1

plot([1, 1], ylim, 'k--', 'LineWidth', 1); % Black dashed line at x = 1

hold off; % Release the hold on the plot

3-savol

% 1. Define the given values

F = 100000; % Total amount to be accumulated

r = 0.0435; % Annual interest rate (4.35%)

N\_values = 5:9; % Number of years (5, 6, 7, 8, 9)

% 2. Calculate the monthly deposit amount for each number of years

P\_values = zeros(size(N\_values)); % Initialize an array to store P values

for i = 1:length(N\_values)

N = N\_values(i);

P = (F \* (r / 12)) / ((1 + r / 12)^(12 \* N) - 1);

P\_values(i) = P;

end

% 3. Create a table to display the results

results\_table = table(N\_values', P\_values', 'VariableNames', {'Years', 'MonthlyDeposit'});

% 4. Display the table

disp(results\_table);

4-savol

grades = {'A-', 'B', 'B+', 'C', 'E', 'A', 'D+', 'A'};

hours = [4, 3, 3, 2, 3, 4, 3, 3];

grade\_values = containers.Map({'A', 'A-', 'B+', 'B', 'B-', 'C+', 'C', 'C-', 'D+', 'D', 'E'}, ...

[4, 3.7, 3.3, 3, 2.7, 2.3, 2, 1.7, 1.3, 1, 0]);

total\_points = 0;

total\_hours = sum(hours);

for i = 1:length(grades)

total\_points = total\_points + grade\_values(grades{i}) \* hours(i);

end

gpa = total\_points / total\_hours;

disp(['GPA: ', num2str(gpa)]);

5-savol

% MATLAB program to find Pythagorean triples (a, b, c) where a, b, and c are

% positive integers less than or equal to 50.

% Definition of Pythagorean Theorem:

% The Pythagorean theorem states that in a right-angled triangle, the square

% of the length of the hypotenuse (the side opposite the right angle) is

% equal to the sum of the squares of the lengths of the other two sides

% (the legs). Mathematically, this is expressed as: a^2 + b^2 = c^2, where 'c'

% represents the length of the hypotenuse, and 'a' and 'b' represent the

% lengths of the other two sides.

% Definition of Pythagorean Triple:

% A Pythagorean triple is a set of three positive integers a, b, and c that

% satisfy the Pythagorean theorem, a^2 + b^2 = c^2.

% Constraints for the problem:

% - a, b, and c must be positive integers.

% - a, b, and c must be less than or equal to 50.

% Algorithm:

% 1. Initialize an empty matrix to store the Pythagorean triples.

% 2. Iterate through all possible integer values for 'a' from 1 to 50.

% 3. For each value of 'a', iterate through all possible integer values for 'b' from 'a' to 50.

% (Starting 'b' from 'a' helps avoid duplicate triples like (3, 4, 5) and (4, 3, 5))

% 4. For each pair of 'a' and 'b', iterate through all possible integer values for 'c' from max(a,b) + 1 to 50.

% (Starting 'c' from max(a,b) + 1 ensures that 'c' is the hypotenuse and avoids unnecessary checks)

% 5. Inside the innermost loop, check if the Pythagorean theorem holds: a^2 + b^2 == c^2.

% 6. If the condition in step 5 is true, then (a, b, c) is a Pythagorean triple. Store this triple in the matrix.

% 7. After iterating through all possible values, display the matrix of Pythagorean triples in a table format.

% Initialize an empty matrix to store the Pythagorean triples

pythagoreanTriples = [];

% Iterate through possible values of 'a'

for a = 1:50

% Iterate through possible values of 'b' (starting from 'a' to avoid duplicates)

for b = a:50

% Iterate through possible values of 'c' (starting from max(a,b) + 1)

for c = (max(a,b) + 1):50

% Check if the Pythagorean theorem holds

if a^2 + b^2 == c^2

% If it holds, add the triple to the matrix

pythagoreanTriples = [pythagoreanTriples; a, b, c];

end

end

end

end

% Display the results in a table format

disp('Pythagorean Triples (a, b, c) where a^2 + b^2 = c^2 and a, b, c <= 50:');

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

disp(' a | b | c |');

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

% Iterate through the found triples and display them

for i = 1:size(pythagoreanTriples, 1)

fprintf(' %-4d | %-5d | %-5d |\n', pythagoreanTriples(i, 1), pythagoreanTriples(i, 2), pythagoreanTriples(i, 3));

end

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