L. Odilon Petra I.
Tugas minggu 3

Link Github:

1. Inputkan nilai polinom dan K
2. Menghitung nilai kolom tabel routh-hurwitz
   Ketika semua baris pertama benilai 0 maka fungsi rhtable(i-1),:)==0 dijalankan. Lakukan perhitungan untuk baris pertama. Menghitung setiap element di dalam tabel.
3. Menghitung nilai poles yang tidak stabil.
4. Dengan mengganggap pole tidak stabil = 0
5. Dicek poles yang tidak stabil
6. Dihasilkan jika tidak ada poles maka sistem stabil.

```
%% Initialization
clear ; close all; clc
% Taking coefficients vector and organizing the first two rows
coeffVector = input('Inputkan koefisien dari sistem: \n i.e. [an an-1 an-2
... K] = ');
ceoffLength = length(coeffVector);
rhTableColumn = round(ceoffLength/2);
%  Initialize Routh-Hurwitz table with empty zero array
rhTable = zeros(ceoffLength,rhTableColumn);
%  Compute first row of the table
rhTable(1,:) = coeffVector(1,1:2:ceoffLength);
%  Check if length of coefficients vector is even or odd
if (rem(ceoffLength,2) ~= 0)
    % if odd, second row of table will be
    rhTable(2,1:rhTableColumn - 1) = coeffVector(1,2:2:ceoffLength);
else
    % if even, second row of table will be
    rhTable(2,:) = coeffVector(1,2:2:ceoffLength);
end
%% Calculate Routh-Hurwitz table's rows
%  Set epss as a small value
epss = 0.01;
%  Calculate other elements of the table
for i = 3:ceoffLength

    %  special case: row of all zeros
    if rhTable(i-1,:) == 0
        order = (ceoffLength - i);
        cnt1 = 0;
        cnt2 = 1;
        for j = 1:rhTableColumn - 1
            rhTable(i-1,j) = (order - cnt1) * rhTable(i-2,cnt2);
            cnt2 = cnt2 + 1;
            cnt1 = cnt1 + 2;
        end
    end

    for j = 1:rhTableColumn - 1
        %  first element of upper row
        firstElemUpperRow = rhTable(i-1,1);

        %  compute each element of the table
        rhTable(i,j) = ((rhTable(i-1,1) * rhTable(i-2,j+1)) - ....
            (rhTable(i-2,1) * rhTable(i-1,j+1))) / firstElemUpperRow;
    end


    %  special case: zero in the first column
    if rhTable(i,1) == 0
        rhTable(i,1) = epss;
    end
end
%%  Compute number of right hand side poles(unstable poles)
%   Initialize unstable poles with zero
unstablePoles = 0;
%   Check change in signs
for i = 1:ceoffLength - 1
    if sign(rhTable(i,1)) * sign(rhTable(i+1,1)) == -1
        unstablePoles = unstablePoles + 1;
    end
end
%   Mencetak Routh Table
fprintf('\n Routh-Hurwitz Table:\n')
rhTable
%   Print the stability result on screen
if unstablePoles == 0
    fprintf(' Sistem stabil! \n')
else
    fprintf(' Sistem tidak stabil! \n')
end
```