Notes:

- The first problem is based on the same model as assignment I, so you can use the model from that assignment.
- For the first problem, you need to submit both a simulink model, matlab '.m' files as well as a document file where you put your answers.
- The second problem can be submitted just as an '.m' file

Problem 1
Submarine problem:

To control the vertical position of a submarine, dedicated pumps would expel or inhale water from its buoyancy tanks in order to vary the amount of buoyancy force applied.

The forces acting on the submarine are
1. the weight (assumed constant)
2. the viscous drag (assumed linear for small motion $V = b\,y'$)
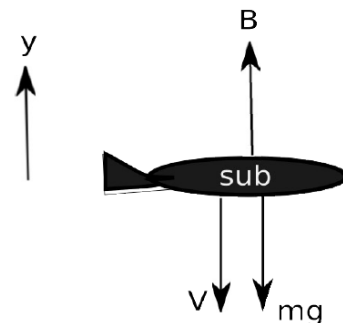3. the buoyancy force ($B = \rho_w * V_{disp} * g$) where

V_{disp} is the volume of water displaced by the submarine (V_{disp} = volume of submarine – volume of water inside the tanks).

Let

$I = \rho_w V_{disp} - m$ be the input to the system, and y is the output.

Given:
– Density of water $\rho_w = 1000$kg $/m_3$
– Mass of submarine $m = 100$kg
– Damping gain $b = 10$N.s$/m$

Part I:
1) Build a simulink model of the problem and using LINMOD find the transfer function of the submarine.
2) Plot the root locus. Is a proportional gain enough to stabilize the submarine?
3) Using SGRID function find for which gain we have a damping ratio of ζ=0.9. Perform simulation of a unit step input with the found gain and plot y vs t.

Part II:

1) The actual control variable on a real submarine is the voltage (V) applied on the pumps responsible for expelling and inhaling water. The transfer function of the pump/tank system is the following:

$$\frac{I}{V} = \frac{0.05s}{0.003s^2 + 0.0031s + 0.0026}$$

    1) Using the same method like before (root locus) can a unity-feedback control with proportional gain stabilize the submarine?
    2) For which gain values is the system marginally stable?
    3) The steady-state behavior for a proportional control is not satisfactory, how do you suggest improving it?
    4) Design a PID control using Ziegler-Nichols method and plot the system response.
    5) Draw the plot bode plot of the Open Loop system (with gains) and find the phase and gainmargins on Matlab.

Problem 2:
Write a matlab m-file(s) that perform partial fraction expansion (you cannot use the 'residue' function). To simplify the problem, assume that the transfer function that will be used as an input, when expanded, will have denominators and numerators consisting of first order terms only, there are no double roots, there are no complex roots, and the order of the denominator is higher than the order of

the numerator (e.g. $\dfrac{3s^2 + 4}{s^3 + 3s^2 - 4s + 1}$).

Your matlab file should have variables *num, den* which act as the inputs that store the coefficients of the numerator and denominator, and fraction in expanded form.