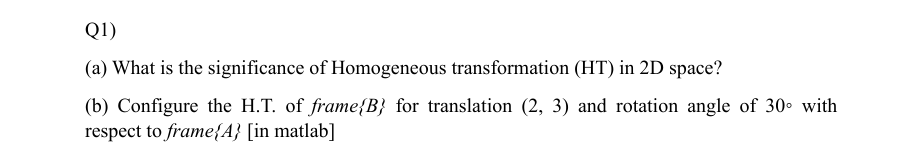
**22AIE214 - INTRODUCTION TO ROBOTICS**

**LABSHEET 3**

Name : Anuvind M P

Roll no : AM.EN.U4AIE22010



**Answer :**

* Integrate translation, rotation, and scaling into one unified 3x3 matrix.
* Utilize homogeneous coordinates [𝑥,𝑦,1] to facilitate transformations through matrix multiplication.
* Allow straightforward sequential application of multiple transformations by multiplying matrices.
* Crucial in computer graphics, robotics, and computer vision for tasks such as object manipulation and image alignment.



**CODE :**

translation = transl2(2, 3);

rotation = trot2(30, 'deg');

HT = translation \* rotation;

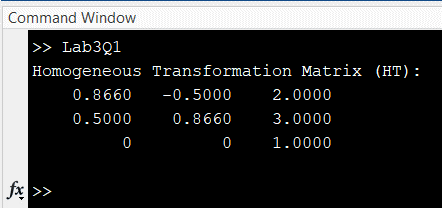
disp('Homogeneous Transformation Matrix (HT):');

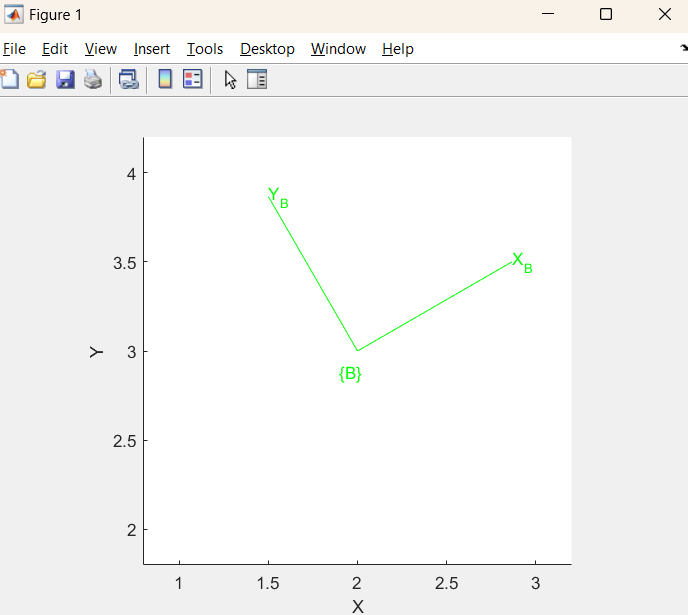
disp(HT);

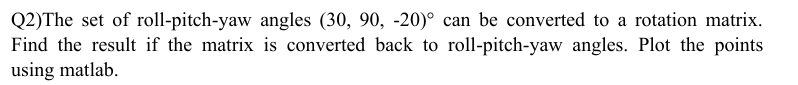
figure;

trplot2(HT, 'frame', 'B', 'color', 'g');

**OUTPUT :**



****

****

**Answer :**

**CODE :**

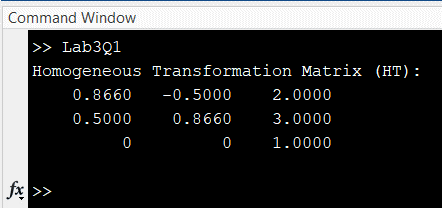
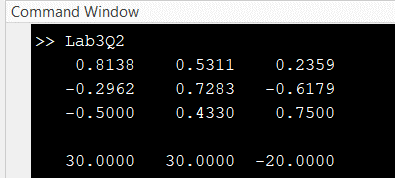
R=rpy2r(30,30,-20,'deg');

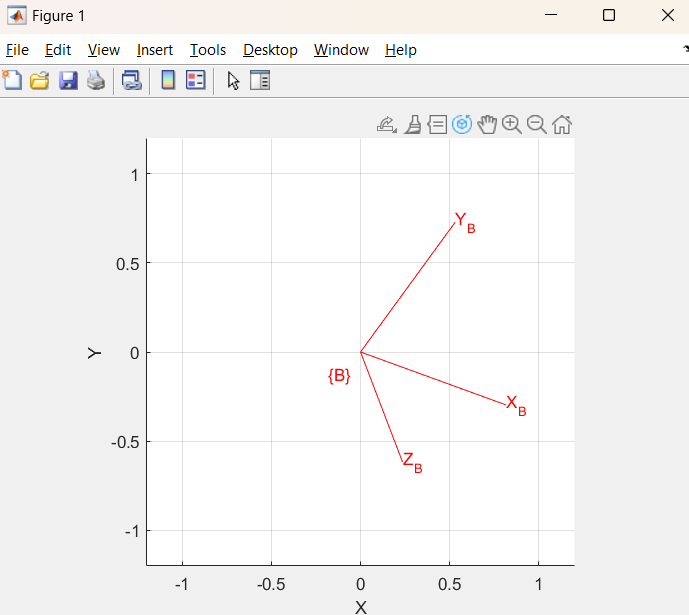
disp(R);

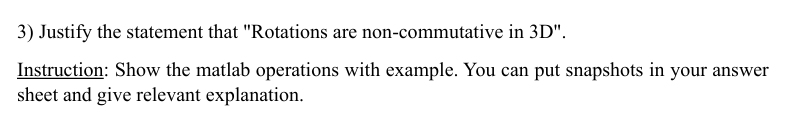
trplot(R, 'frame', 'B', 'color', 'r');

disp(tr2rpy(R,'deg'));

**OUTPUT :**



****



**Answer :**

**CODE :**

Rx = rotx(45, 'deg');

Ry = roty(90, 'deg');

R1 = Ry \* Rx;

R2 = Rx \* Ry;

disp('Rotation matrix R1 (Ry \* Rx):');

disp(R1);

disp('Rotation matrix R2 (Rx \* Ry):');

disp(R2);

isEqual = isequal(round(R1, 10), round(R2, 10));

disp('Are the rotation matrices equal?');

disp(isEqual);

figure;

subplot(1,2,1);

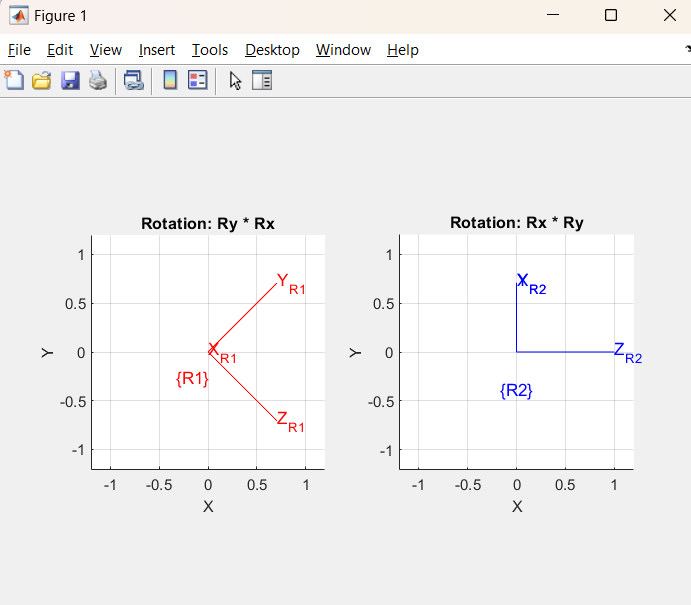
trplot(R1, 'frame', 'R1', 'color', 'r');

title('Rotation: Ry \* Rx');

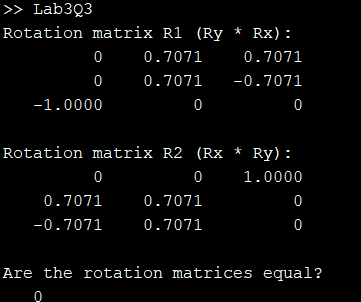
subplot(1,2,2);

trplot(R2, 'frame', 'R2', 'color', 'b');

title('Rotation: Rx \* Ry');

****

**OUTPUT :**



**Explanation :**

The two plots will display different orientations of the coordinate frames, visually illustrating the non-commutative nature of rotations. When the order of rotations is altered, the resulting matrices and their respective coordinate frames differ.