22AIE214 - INTRODUCTION TO ROBOTICS

LABSHEET 3

Name : Anuvind M P

Roll no : AM.EN.U4AIE22010

Q1)

(a) What is the significance of Homogeneous transformation (HT) in 2D space?

(b) Configure the H.T. of *frame{B}* for translation (2, 3) and rotation angle of 30◦ with respect to *frame{A}* [in matlab]

Answer :

a.

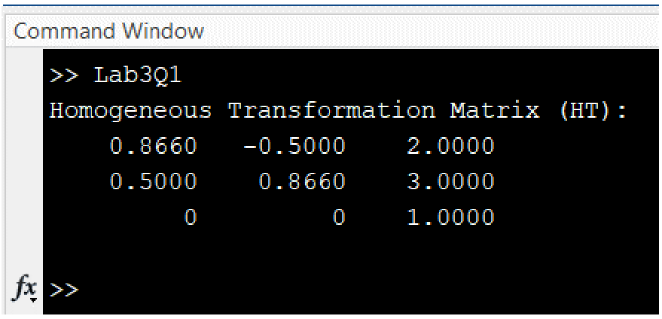
- Integrate translation, rotation, and scaling into one unified 3x3 matrix.
- Utilize homogeneous coordinates $[x,y,1]$ to facilitate transformations through matrix multiplication.
- Allow straightforward sequential application of multiple transformations by multiplying matrices.
- Crucial in computer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

b.

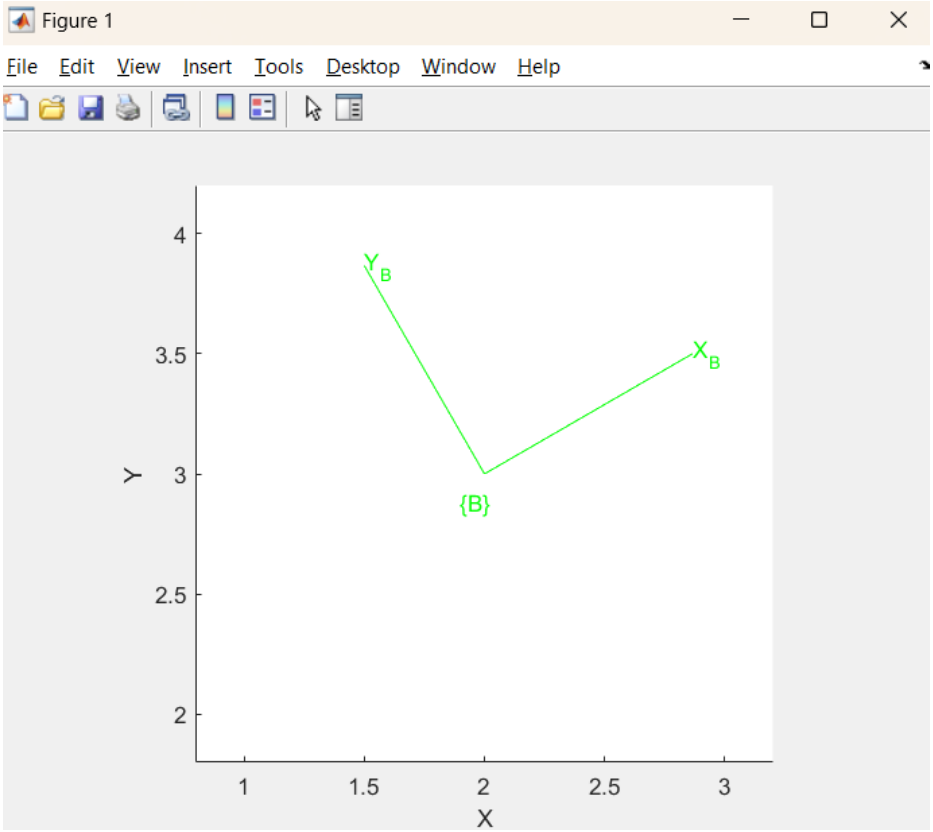
CODE :

```
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 :



```
>> Lab3Q1
Homogeneous Transformation Matrix (HT):
    0.8660   -0.5000    2.0000
    0.5000    0.8660    3.0000
         0         0    1.0000

>>
```



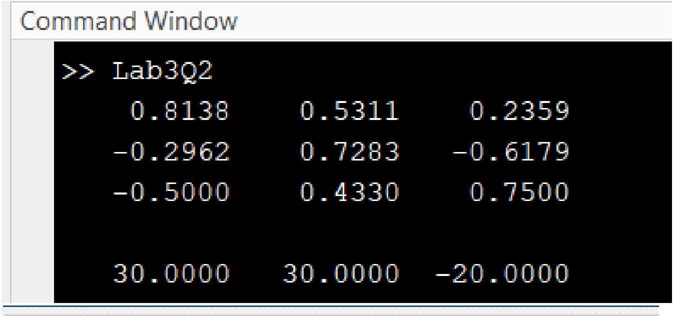
Q2)The set of roll-pitch-yaw angles (30, 90, -20)° can be converted to a rotation matrix. Find the result if the matrix is converted back to roll-pitch-yaw angles. Plot the points using matlab.

Answer :

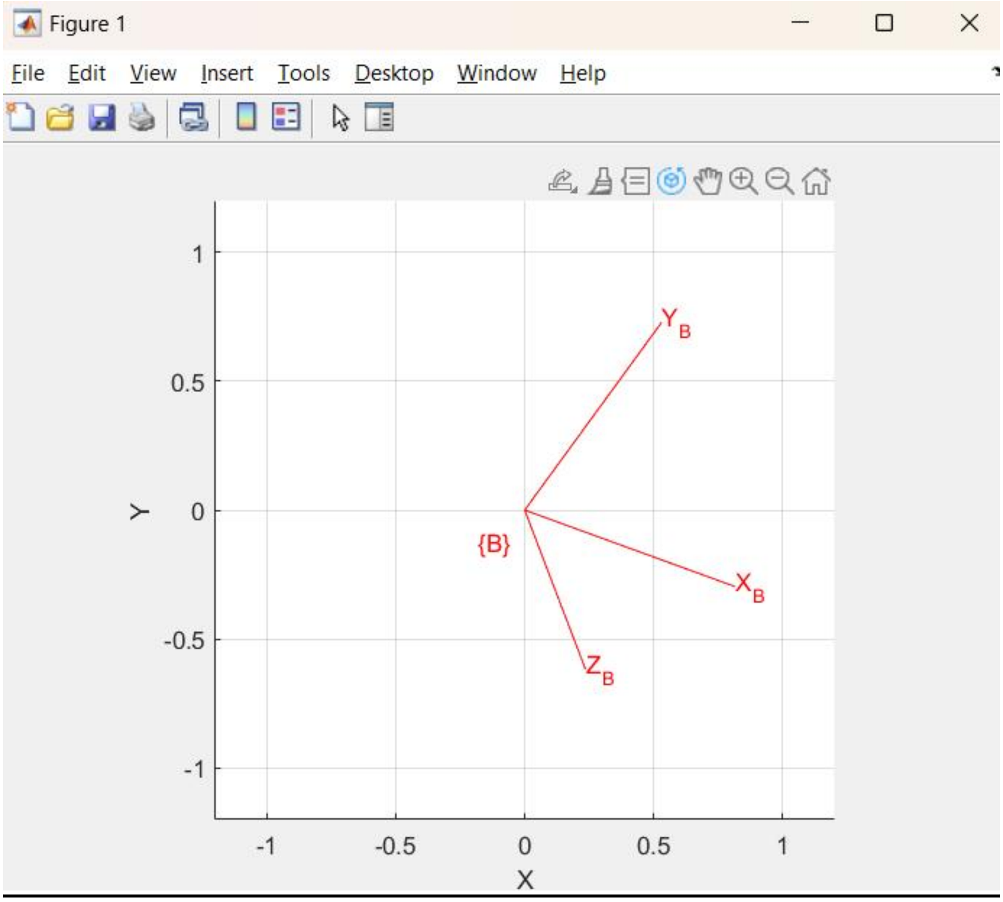
CODE :

```
R=rpy2r(30,30,-20,'deg');
disp(R);
trplot(R, 'frame', 'B', 'color', 'r');
disp(tr2rpy(R,'deg'));
```

OUTPUT :



```
>> Lab3Q2
    0.8138    0.5311    0.2359
   -0.2962    0.7283   -0.6179
   -0.5000    0.4330    0.7500

   30.0000   30.0000  -20.0000
```



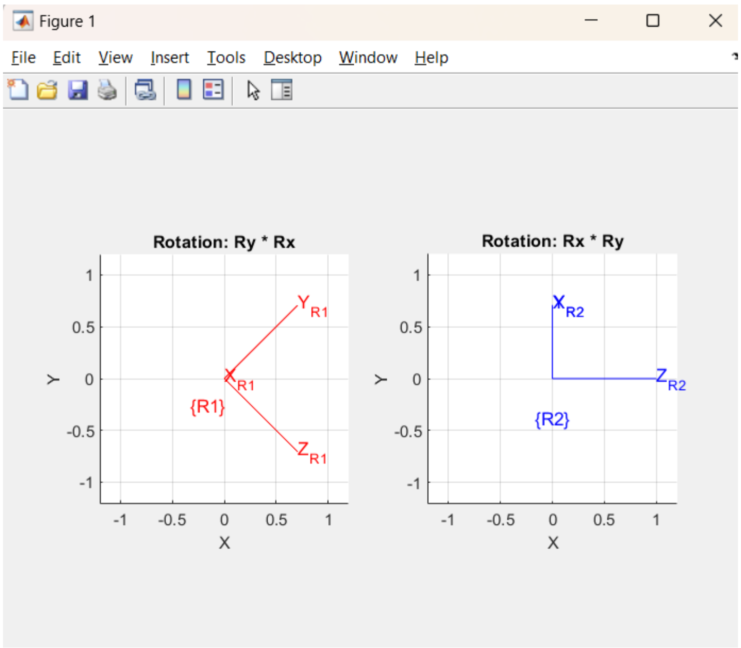
3) Justify the statement that "Rotations are non-commutative in 3D".

Instruction: Show the matlab operations with example. You can put snapshots in your answer sheet and give relevant explanation.

Answer :

CODE :

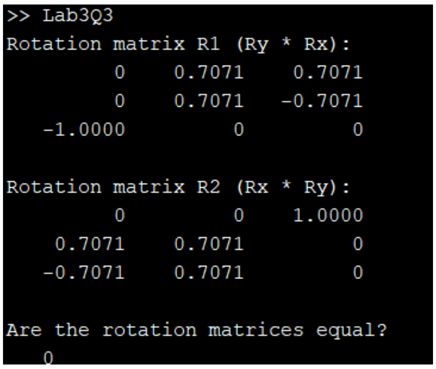
```
Rx = rotx(45, 'deg');
Ry = roty(90, 'deg');
R1 = Ry * Rx;
R2 = Rx * Ry;
disp('Rotation matrix R1 (Ry * Rx):');
disp(R1);

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');
```



Rotation: Ry * Rx

Rotation: Rx * Ry

OUTPUT :



```
>> Lab3Q3
Rotation matrix R1 (Ry * Rx):
         0    0.7071    0.7071
         0    0.7071   -0.7071
   -1.0000         0         0

Rotation matrix R2 (Rx * Ry):
         0         0    1.0000
    0.7071    0.7071         0
   -0.7071    0.7071         0

Are the rotation matrices equal?
   0
```

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.