ECSE/CSDS/EMAE 489
Problem Set 4: Inverse Kinematics
Assigned: 2/24/21
Due: 3/3/21

1) Motoman kinematics
For the Motoman robot, assume the following DH values, joint limits, and offsets from DH home thetas to manufacturer's home thetas.

```
const double DH_a1=0.145; //offset from J1 to J2
const double DH_a2=1.150; //length of humerus
const double DH_a3=0.250; //offset from J3 to J4
const double DH_a4=0.0;  //spherical wrist...
const double DH_a5=0.0;
const double DH_a6=0.0;

const double DH_d1 = 0.540; //height from baseplate to J2
const double DH_d2 = 0.0;
const double DH_d3 = 0.0;
const double DH_d4 = -1.812; //length of forearm
const double DH_d5 = 0;
//since z5 points INTO flange, d-value for origin at flange face is negative
const double DH_d6 = -0.100;

//turret--> positive rotation is "up"
//z0 x z1 points backwards; so define alpha instead as -3pi/2
//shoulder axis--> bends FORWARD with positive rotation--> alpha = negative pi/2
// also, Motoman home has humerus vertical--offset of pi/2 from DH home;
const double DH_alpha1 = 3.0*M_PI/2.0;

//elbow + rotation is antiparallel to shoulder + rotation, so twist = pi
// also, offset from J2 to J3--> x3 axis; is parallel to x2 axis when elbow
// angle is 90 deg, i.e. same as Motoman home pose
const double DH_alpha2 = -M_PI;

//forearm +rotation points from wrist towards elbow
const double DH_alpha3 = -M_PI/2.0;

// at home pose of J4, J5 is parallel to J3; DH and Motoman home consistent for J4
// if choose alpha +pi/2 (and thus x4 points UP in Motoman home pose)
const double DH_alpha4 = M_PI/2.0;

// pos toolflange rotation is INTO flange (not out of).
// use alpha = -3pi/2; DH home has wrist bend pointing straight out;
// but  x5 = z4 x z5  points "down" in motoman home pose
// so define alpha as -3pi/2, not pi/2, which places x5 pointing "up"
//   then there is no angle offset from motoman to DH
const double DH_alpha5 =  3.0*M_PI/2.0;
```

```
//prefer toolframe axis to have z_flange pointing OUT:
const double DH_alpha6 =   M_PI;

//theta_offsets: to transform between Motoman coords and DH coords;
// Motoman J2 home (upright) is 90deg offset from DH home
// Motoman J5 home (bent 90 deg w/rt forearm) is 90deg offset from DH home (straight out)
//conversion: double q_DH = q_motoman + DH_q_offsets[i];
const double DH_q_offset1 = 0.0;
const double DH_q_offset2 = -M_PI/2.0;  //at Motoman home, q_DH2 is -pi/2
const double DH_q_offset3 = 0.0;
const double DH_q_offset4 = 0;
const double DH_q_offset5 = 0; //GP20: home--> toolz is parallel to forearm axis;
const double DH_q_offset6 = 0; //range is actually +/- 360deg

//express these in MOTOMAN coords, not DH coords
//THESE ARE RESTRICTED TO LESS THAN PHYSICALLY POSSIBLE RANGE
const double DH_q_max1 = 1.57;
const double DH_q_max2 = 2.35;
const double DH_q_max3 = 1.45;
const double DH_q_max4 = 3;
const double DH_q_max5 = 2;
const double DH_q_max6 = 6;

// expressed in MOTOMAN coords,  not DH coords
const double DH_q_min1 = -1.57;
const double DH_q_min2 = -1.4;
const double DH_q_min3 = -1.39; //somewhat musclebound; can't fold forearm parallel to humerus
const double DH_q_min4 = -3;
const double DH_q_min5 = -2; //wrist home is straight out; can bend +/- 150deg  (but limit this)
const double DH_q_min6 = -6;
```

Write a program that computes inverse kinematics as a function of given desired toolflange pose (DH6 frame) with respect to the base link (DH0 frame).

Constrain your solutions to the above joint limits.
Also, constrain your solutions to "elbow up" (i.e., q3<0 in Motoman joint values).

Use this function to compute a straight-line path defined by the following:

Constrain the tool (DH6) frame orientation to be constant along the path with values:

```
R_6/0 =
[ 0.707107         0  0.707107
        0        -1         0
 0.707107         0 -0.707107]
```

The origin of the toolframe at sample points "i" should be **o**_6/0(i) = [2,-2+0.1*i, 0.2] for i=0 to 40. That is, the toolflange should sweep out a straight line parallel to the x axis from [2,-2,0.2] to [2,+2,0.2].

Choose a sequence of solutions, **q**(i), that moves the robot through this desired sequence of poses. Plot out your chosen joint angles (six traces) as a function of index “i”.

Prove that your solution is viable by computing forward kinematics for each solution and plotting out x,y,z as a function of of index “i”.

Suggestion: have your team work in parallel to make functions for:

- computing q3 from wrist coordinates
- computing q2 from wrist coordinates
- computing q1 from wrist coordinates
- computing coordination of q1,q2,q3 combinations
- computing (q4,q5,q6) solutions given q1,q2,q3

Challenge question:
For the robot in Q1, and for the same toolflange orientation and z-height, what is the longest continuous path you can find from (x0,y0) to (xf,yf). A viable path must have no joint-angle discontinuities.

Describe your solution. Plot out your joint angles vs sample number (6 traces). Plot out x,y,z vs sample number for your solution.

Deliverables:
Include a description of your approach to your solutions. Include your code (presumably in Matlab). Include your plots. All members of a group will submit the same solution (redundantly). Don’t forget to enter your group/self rankings with your submission.