

Modeling of Control and Manipulators

Exercise 1

- 1) To start off download the repository at <https://github.com/merosss/MOCOMLAB1> (QR Code below).
- 2) You will now need to define all the model matrices, and fill the so called “biTri” matrix. Be careful to define the z-axis coinciding with the joint rotation axis, and such that the positive rotation is the same as showed in the CAD model you received.
- 3) Then, develop a function called “DirectGeometry()” which can calculate how the matrix attached to a joint will rotate if the joint rotates, using the interface:

```
function biTei = DirectGeometry(qi, biTri, linkType)
```
- 4) The last step will be useful in developing the “GetDirectGeometry()” function which will return all the model matrices given a certain **q** configuration. The boldness is to remind that these variable will now be vectors, of length equal to the number of joints.

```
function biTei = GetDirectGeometry(q, biTri, linkType)
```
- 5) Now that you have the “biTei” you can calculate all the transformation matrices between any two links, between a link and the base and the corresponding distance vectors. To do so incorporate these functionalities in the repository functions named accordingly.
- 6) Now that you’ve reached this point define a starting and ending configuration, **q_i** and **q_f**, and try to plot all the intermediate link positions in between the two (you can use the `plot3()` or `line()` functions).