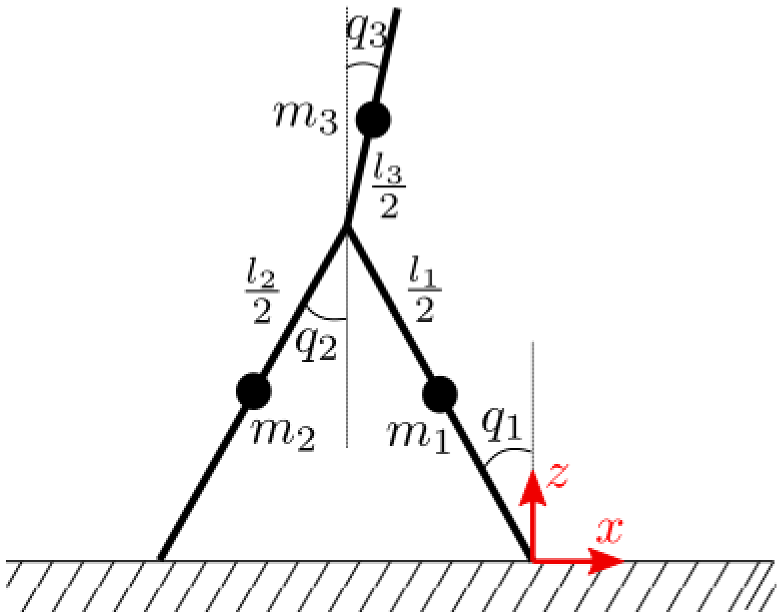
Developing a Walking Controller for a Three-link 2D Biped

(Legged Robotics EPFL, Fall 2018)

Here is a schematic of the three-link 2D biped:



Three-link 2D Biped

You will be developing a **walking controller** for this toy model. To this end, you would need to complete the following three stages:

1. (Assignment 2) Model and visualize the 3-link biped
2. (Assignment 3) Solve the equations of motion of the 3-link biped (Simulation)
3. (Assignment 4) Design a walking controller, and evaluate the resulting gait

Note: It is highly recommended that you use a version control tool (such as git) to complete this project.

Assignments	Deadline	Percentage
Assignment 1	Oct. 15, 2018	Not graded
Assignment 2	Nov. 5, 2018	35%
Assignment 3	Nov. 19, 2018	20%
Assignment 4	Dec. 17, 2018	45%

Mini-project Assignments

1. Model and Visualize (Assignment 2)

In the first step you generate the kinematics and dynamics of the three-link biped. For that you need to complete the following scripts in the presented order:

```
generate_kinematics.mlx (in the "generate_model" folder)
visualize.m
generate_dynamics.mlx (in the "generate_model" folder)
eval_M.m, eval_C.m, eval_G.m, eval_B.m
generate_impact_map.mlx (in the "generate_model" folder)
eval_A_m.m, eval_A_p.m, impact.m, eval_energy.m
```

The function `visualize.m` should plot a schematic of the 3-link biped, and it is used to help you verify your code. Later this function will be used for making an animation of the 3-link biped simulation.

To generate the kinematics and dynamics, use the generalized coordinates $q = [q_1; q_2; q_3]$ as shown in the figure above. *Pay careful attention to the positive direction of the angles and the inertial coordinate system at the stance foot (figure above).*

By the end of this task, you have calculated the formulas for the matrices M, C, G, and B in the equations of motion:

$$M\ddot{q} + C\dot{q} + G(q) = Bu$$

where, $u = [u_1; u_2]$ is the control vector. The matrices M, C, G and B will be calculated by the functions `eval_M.m, eval_C.m, eval_G.m, eval_G.m` in the `dynamics` folder (which you will complete).

Moreover, you develop an impact map `impact.m` which maps the generalized coordinates and their derivatives ($q^-, \dot{q}^-$) right before the impact to after impact ($q^+, \dot{q}^+$). To this end you will follow the instructions in `generate_impact_map.mlx` and complete the scripts `eval_A_m.m, eval_A_p.m, impact.m, eval_energy.m`.

Submission deadline: Monday November 5, 2018, at 23:59 (3 weeks, 35% of the mini-project grade)

2. Solve the equations of motion: Simulation (Assignment 3)

In this task you first will rewrite the equations of motion in the state-space form. Then you will use `ode` solver of MATLAB to solve the equations of motion. Afterwards, you will animate the simulation. To this end, you will complete the following functions:

```
solve_eqns.m
animate.m
```

Submission deadline: Monday, November 19, 2018, at 23:59 (2 weeks, 20% of the mini-project grade)