EPFL Fall 2018
Doctoral School
Civil and Environmental Engineering

Scientific Programming
for Engineers

Week 14 — *Pybind: Python bindings for Particles Code*

The goal for this exercise is to use external library Pybind11 made for creating Python bindings of C++ code. We will use it to create Python bindings for our Particles code.
Use the starting point from GIT for this exercise

We have provided a **main.py** in the starting point. This script shows all the C++ classes and their respective functions to be exposed to Python. The objective of the exercise will be to ensure that the script works for all types of particle : Planet, PingPong and MaterialPoint.

Exercise 1: *Factory Interface*

1. Create python bindings for all factory interface classes : `ParticlesFactory`, `MaterialPointsFactory`, `PlanetsFactory` and `PingPongBallsFactory`.
2. In class `MaterialPointsFactory`, `createSimulation` function has been overloaded to take functor as one of its argument. Comment on what this function is doing?
3. Create python binding for `createSimulation` function. You will have to use `overload_cast`. For more help, please refer : Overloading.

Exercise 2: *Compute*

1. Create python binding for classes: `Compute` and `ComputeTemperature`.
2. How will you ensure that references to Compute objects type are correctly managed in the python bindings?
3. Some of the private members of class `ComputeTemperature` are made accessible in **main.py**. Create python bindings to access these variables and set their values.

Exercise 3: *Other Classes*

1. Create python bindings for other necessary classes and their respective functions according to **main.py**.