ROBT 414 Programming Assignment 2

This is an individual assignment.
Code submission deadline: 12/10/2021 23:55.
Assignment is worth 15% of overall course grade.

This assignment requires you to implement a "Follow Me" application on a Pepper robot using Python NaoQi. Check out an example of the "Follow Me" application at the following link:

https://www.youtube.com/watch?v=K-fXHvKL86w

As you can see the motion is smooth and you should aim for that. Upon the start of the application, the robot should wave hello and say: "Hi, human! I want to come with you". Then, the robot raises one of its arms for the user to hold it. You could use robot's wrist as a steering wheel to change direction (i.e. to turn right or left), and robot's shoulder to control the robot's speed (e.g. when its arm is up, the robot moves faster, when it is at 90-degree angle, the robot moves slower; when it is dropped, the robot stops). These robot joints can be used with manual stiffness control:
RShoulderRoll - RShoulderPitch - RElbowYaw - RElbowRoll - RWristYaw - RHand or
LShoulderRoll - LShoulderPitch - LElbowYaw - LElbowRoll - LWristYaw - LHand
For documentation and usage details, please see:

http://doc.aldebaran.com/2-4/naoqi/motion/control-stiffness.html

You have to program your application in Python NaoQi and test with a virtual robot in RVIZ, Gazebo, or Choreographe. There is a video on moodle on how to install and setup Python NaoQi. We also provide a simple code of how to read coordinates of robot's joints and make the robot move. However, you are advised to use Choreographe to create robot animations (e.g. waving hello) on a virtual robot and then export them to python -> biesier to be integrated in your program in Python. **Consult the uploaded books on Moodle to go over the tutorials on how to create robot animations using the timeline (e.g. Macarena dance is in NAO Programming Book 2 p.56).** Here is the link where you can download Choreographe installer according to the Operating System of your computer. Choreographe is a desktop application, which allows to create behaviors and test them on a simulated NAO robot.

https://drive.google.com/open?id=0B2vBVPaYI0KqNW4zaEUweEFHcUU

Choreographe uses NaoQi, which is a software library for programming NAO. Choreographe, NaoQi documentation and API are very useful when programming the robot and can be accessed at:
http://doc.aldebaran.com/2-1/software/choregraphe/index.html
http://doc.aldebaran.com/2-1/index_dev_guide.html

Instructions for integrating Choreographe's virtual robot and your Python code: your virtual robot's IP is 127.0.0.1 and its port can be found in Choreographe, but default is 9559.

Once you are finished with the task, you need to record yourself and your screen where you demonstrate your program and explain your code. Your videos should not be more than ten minutes long. You can use the zoom, the loom, or any other software that you like for the recordings. Don't upload the videos to Moodle, only share the link to the Google Drive with your videos at the Moodle's submission text field. You will be assessed on how correct your solution is in addition to how well you explain the solution. Note: your code needs to be well commented. And your submission needs to have a README file.

Rubrics: 20 points for the use of exported animation values + 20 points for robot's going forward and stopping + 20 points for the speed change + 20 points for turning right and left + 10 points for the use of TTS + 10 point for code and video explanation clarity.

For questions, please feel free to email our TA, Nazerke Rakhymbayeva at nazerke.rakhymbayeva@nu.edu.kz (and cc me).

Extra points (10%): integrate your application with your Assignment 1 to make the robot move according to human pose recognitions (e.g. a raised left arm to turn left, a raised right arm to turn right, etc.)