Exercise I:

Summary:

In this exercise we manipulate a data file via batch processing. The processing is performed in three different languages: Matlab, R and Python. The data file and the computational processing effort in this example is small so it could be completed easily just using a text editor. On the other hand, when the data files and processing effort are much, much bigger, such as in advanced bioinformatics research, a “manual” approach is unfeasible. However, the *arrangement* of such large scale computations happens in the same way as we do it in this exercise.

Below are the steps to be taken.

Open a terminal (Activities->right-click terminal icon):

Copy data package for exercise 1 to your Desktop (**type at the command prompt end enter strings in italics**):

*cp /tmp/ex1.tar.gz ~/Desktop*

Navigate to Desktop:

*cd Desktop*

Unpack the package:

*tar -xzvf ex1.tar.gz*

Inspect content of the directory:

*ls -la .*

Please note that Linux commands can be aggregated and scripted into a text file. The file can be executed as a new command then.

Next you design and configure a compute job. Use the job file template to create a combined Matlab, R and Python job. Open the file batchfile\_1 in a text editor and visually inspect the content of the file:

*gedit batchfile\_1*

No changes have to be made in the file at this point.

Submit the compute job to the wait queue:

*qsub -m abe batchfile\_1*

Monitor the queue and the submitted job:

*qstat*

*showq*

Upon completion (C status in qstat) examine the result and output/error files (replace “<jobid>” with proper number from qstat output):

*more Ex\_1\_script.e<jobid>*

*more Ex\_1\_script.o<jobid>*

Also inspect the output file that was created by the programs:

*more Spill\_Matlabnew\_1.txt*

Assess the outcome of your compute job. Describe how the batch scheduler was used to run the computation. What happened at each step of the procedure?