

# Pre-assignment - TDA596 Distributed systems

Mohamed El Mouctar HAIDARA [mouctar@student.chalmers.se](mailto:mouctar@student.chalmers.se)

November 4, 2016

## I Python

The source code for the exercises is located in the root folder. There are three files : string.py, list1.py and wordcount.py.

## II Seattle framework

1. Seattle is a platform for networking and distributed systems research. It's free, community-driven, and offers a large deployment of computers spread across the world.
2. A vessel is a virtual machine in the Seattle framework. It is a controlled environment for running code (implemented using the repy sandbox).
3. The programming language used is Repty. Repty is a Python-based sandbox which restricts API calls and limits the consumption of resources such as CPU, memory, storage space, and network bandwidth. Repty has a limited API for accessing system resources. This is in order to prevent a malicious user using a bug in the Python libraries to escape from the sandbox.

4. To run locally a program, you can type :

```
python <path to repy.py> <path to restrictions file> <path to source file>
```

5. To run program remotely on the vessels or VMs, you need:
  - (a) To get some vessels on which you want to run your program. This can be done on the ClearingHouse page once you are registered. You will also get some keys (public and private) that are used to connect to vessels.
  - (b) To run the Seattle Shell (seash) `python seash.py` . This will run an interactive shell.
  - (c) To load your account keys `!> loadkeys yourusername` . You have to put the keys files (yourusername.publickey, yourusername.privatekey) in the same directory as the file `seash.py` .

- (d) To connect by typing `!> as yourusername` in seash. After this step, you can see the vessels you control: `yourusername@ !> browse`.

After these steps, you can run a program on a specific or many vessels. For example, the following command will run the helloworld file on the vessel number 1.  
`yourusername@ !> on %1 run helloworld.repy`.

6. The command `listencommhandle = waitforconn(ip,port,hello)` is used to listen connection the specified `ip` and `port`. Once a connection has been established, the callback `hello` will be called.