**University of Leeds**

**School of Computing**

**COMP3900 Distributed Systems**

**Artur Oliveira Rodrigues**

**SID: 200622082**

**Coursework 2**

**Question 1:**

Submitted online (tagged as 1.1)/not relevant for this document.

\*\*\* The code in question is supposed to implement a concept of a city, which has a name and is part of a Country. This class also holds information about its minimum and maximum temperatures. Looking at the bigger picture, there is a skeleton of an RMI architecture, where I believe clients will be able to register new cities and get their information.

**Question 2:**

The server program in the file CityServer.java is a simple implementation of a RMI server that expects one argument, which will be the name of the City, represented by the object. This object is then registered and the user is informed of the process. Boilerplate error handling is also present in the file.

No files were submitted online as no changes were made to the server.

**Question 3:**

The code was submitted online (tagged as 2.1). What was done was the implementation of the client. It was based on the Bank example of Lab Sessions week 4. It basically locates the registry based on the specified host, and then looks for the City object passed as argument. It then updates the objects private fields, country, minimum temperature and maximum temperature through the public setter functions. Reports are printed on the screen in order to inform the user.

**Question 4:**

Submitted online (tagged as 3.1)/not relevant for this document.