Report exercises 3 – 4

912404 – Marco Edoardo Santimaria
912759 – Nicolo' Vanzo
926194 – Mia Zimonjic

1 - How to compile our project!

To compile our program go to the root directory of the project and launch the following command to compile the program:

- $ ant demo

After this you can execute the program by launching the following command:

- $ java -jar build/GraphDemo.jar <yourFile>.csv

To execute UnitTests write the command:

- $ ant

2 – Data structures used

Along the project we used many different data structures.
Here's a list of our data structures:
1 – For the UnionFindSet we used a HashMap to contains the list of the Graph nodes, every element of this list has a pointer to its representative (another node, called "parent"), a value and a rank. You can find this HashMap inside the Graph.java file and the structure of the elements inside the HashMap in the UnionFindSetElement.java;
2 – For the Graph structure we use an adjacent list implemented with an HashMap with different methods required by the exercise;
3 – For the Kruskal algorithm we used: a graph which contains the elements that can be read by the .csv file; an UnionFindSet structure used to store the nodes of the graph; two ArrayLists, on contains the graphLinks and the other is used to store the MstGraph (the output of the Kruskal algorithm)

3 – A note on generic types used

Throughout the development we used two generic types:
- T : the type of the node
- G : the type of the Link

In order to apply Kruskal on a Graph the type of the link must be on of the following type:
- Byte
- Short
- Integer
- Long
- Float
- Double

Other types are not allowed since they cannot be ordered with arithmetic operators.