The file ‘ontology.xml’ provided contains the description of an ontology between the ‘Ontology’ XML tags. There is only one type of ontology entity used here in this scenario:

* concepts (between the ‘Concepts’ XML tags)

A concept is represented using the ‘Concept’ XML tag, and its name is specified using the “name” attribute. The representation of a concept includes several XML children, of which only one is relevant to this test - ‘DirectSuperConcepts’ – which lists all the direct parents of the concept (if a concept has no parent, this XML tag will have no XML children in the representation). A parent is represented with the XML tag ‘ConceptReference’ and its name is specified using the ‘name’ attribute:

<Concept name=”concept1” …>

…

<DirectSuperConcepts>

<ConceptReference name=”concept2” />

<ConceptReference name=”concept3” />

…

</DirectSuperConcepts>

…

</Concept>

The XML fragment above defines the concept “concept1” that has 2 direct parents: “concept2” and “concept3”. Consequently, “concept1” is a child of “concept2”, and also a child of “concept3”. A root concept is a concept that has no other concept as a parent. A leaf concept is a concept that has no other concept has a child. Ignore any other XML entities and tags in the ontology.xml file.

Develop a Java program that does the following:

1. Parses the provided ontology.xml file, extracts the name of the concepts and their parents, and represents them in an internal data structure
2. Sorts all the concepts alphabetically (case insensitive), and for each concept sorts their parents alphabetically (case insensitive)
3. Generates and writes on disk a JSON file for the sorted list of concepts, where each concept is represented as a JSON object with the following structure:

{

"name" : "<the extracted name>",

"parents" : [ "<the extracted name of parent 1>", "<the extracted name of parent 2>", ....]

}

1. Compute the following statistics and write them on disk in a regular text file an, each on a new line:

total number of concepts: <number>

average number of parents: <number>

number of leaf concepts: <number>

longest paths from a root to a leaf concept: <path length as a number of links>

- root1 -> concept1 -> … -> leaf1

- root2 -> … -> leaf2

…

For example:

total number of concepts: 40  
average number of parents: 1.10  
number of leaf concepts: 20  
longest paths from a root to a leaf concept: 4  
- concept0 -> concept1 -> concept7 -> concept17 -> concept21  
- concept0 -> concept1 -> concept8 -> concept19 -> concept30