Organizations usually have hierarchical structures that can be represented as trees; for example, companies, governments, and schools.

The goal of HW3 is to build a tree from organizational data and answer queries on the organizational structure. Your submission has a Tree class that supports (at least) the following operations:

- `addChild(parentNode, childNode)` // to maintain alphabetical/lexicographical order
- `getChildren(node)`
- `getParent(node)`

For each node, you may not assume it has a fixed or maximum number of children.

**Input:** Input is from the command-line arguments for HW3.java in this order:

1. filename of the organizational data—the top entity in on the first line, a pair of supervisor and subordinate is on one line starting on the second line.
2. filename of queries, each line has one of the following queries:
   - `DirectSupervisor entity`
   - `DirectSubordinates entity`
   - `AllSupervisors entity`
   - `AllSubordinates entity`
   - `NumberOfAllSupervisors entity`
   - `NumberOfAllSubordinates entity`
   - `IsSupervisor entity supervisor`
   - `IsSubordinate entity subordinate`
   - `CompareRank entity1 entity2`
   - `ClosestCommonSupervisor entity1 entity2`

You may assume entities in the queries exist in the organizational data. Sample input files are on the course website.

**Output:** Output goes to the standard output (screen), each line has an answer with the corresponding query:

- `DirectSupervisor entity supervisor`
- `DirectSubordinates entity subordinate1 subordinate2 ...`
- `AllSupervisors entity supervisor1 supervisor2 ...`
- `AllSubordinates entity subordinate1 subordinate2 ...`
- `NumberOfAllSupervisors entity count`
- `NumberOfAllSubordinates entity count`
- `IsSupervisor entity supervisor yes/no`
- `IsSubordinate entity subordinate yes/no`
- `CompareRank entity1 entity2 higher/lower/same`
- `ClosestCommonSupervisor entity1 entity2 closestCommonSupervisor`

DirectSubordinates (if any) are in the alphabetical/lexicographical order. AllSubordinates (if any) are in “pre-order”. AllSupervisors (if any) are in the order of supervisor, grand-supervisor, ... `IsSupervisor/IsSubordinate` refers to any supervisor/subordinate. In an organization, an entity has a higher rank when it is closer to the top in the hierarchy. `CompareRank` prints higher (lower/same) if `entity1` has a higher (lower/same) rank than `entity2`. `ClosestCommonSupervisor` is the lowest-ranking supervisor of both entities. Sample output is on the course website.

**Submission:** Submit HW3.java that has the main method, Tree.java, and other program files. Submissions for Individual and GroupHelp have the same guidelines as HW1.

Note the late penalty on the syllabus if you submit after the due date and time as specified at the top of the assignment.