

Due Wed Oct 3 at the start of your lab section; Submit Server: `class = cse2010, assignment = hw3$Individual`

Due Wed Oct 3 at the end of your lab section; Submit Server: `class = cse2010, assignment = hw3$GroupHelp`
`x` is 2, 3, or 4—your section number or “j” for java submissions.

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 struct` that has a linked structure of tree nodes and supports (at least) the following operations:

- `addChild(tree, node, childNode)` or `addChild(node, childNode)` // to maintain alphabetical/lexicographical order of the children
- `getChildren(tree, node)` or `getChildren(node)`
- `getParent(tree, node)` or `getParent(node)`

For each node, you may assume the maximum number of children is 100. (Optionally, `Tree struct` and its operations are in `treeInterface.h`, `treeTypes.h`, `treeImpl.c` similar to Programs 7.4 and 7.19 in the book. `gcc -o hw3 hw3.c treeImpl.c`) We will be evaluating your submission on `code01.fit.edu`; we recommend you to ensure that your submission runs on `code01.fit.edu`.

Input: Input is from the command-line arguments for `hw3.c` 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 each 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` (if any) is the lowest-ranking supervisor of both entities. Sample output is on the course website.

Submission: Submit `hw3.c` that has the main method, (optionally separate `treeInterface.h`, `treeTypes.h`, `treeImpl.c`), 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.