CSE 2010, HW3

Due Thu Feb 22 at the start of your lab section; Submit Server: class = cse2010, assignment = hw3S*x*Individual

Due Thu Feb 22 at the end of your lab section; Submit Server: class = cse2010, assignment = hw3S*x*GroupHelp

x is 14 or 23—your merged section number.

Many entities, such as Olympic events, can be organized in a hierarchy. How would you design a system that stores the hierarchy and allow queries on the hierarchy?

The goal of HW3 is to build a tree from Winter Olympic events and answer queries on the tree. The Winter Olympics have multiple sports, each sport has multiple events, and each event has three winners (usually). In the tree, the sports and events are ordered alphabetically/lexicographically and the winners are ordered in gold-silver-bronze order. For simplicity, assume each event has only 3 winners. Your submission includes a Tree class that has a linked structure of tree nodes and supports (at least) the following operations:

- insertChild(parentNode, childNode) // to maintain alphabetical/lexicographical order
- appendChild(parentNode, childNode) // to maintain order of addition
- getChildren(node)
- getParent(node)

For each node, you may not assume it has a fixed or maximum number of children. Sample input and output are on the course website.

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

1. filename of the data—each line has an entity followed by its childern: the first line has the Olympics year followed by its sports, each sport is followed by its events, each event is followed by its winners in gold-silver-bronze order. Each winner has the athlete and country separated by a colon.
2. filename of queries, each line has one of the following queries:
   - GetEventsBySport *sport*
   - GetWinnersAndCountriesBySportAndEvent *sport event*
   - GetGoldMedalistAndCountryBySportAndEvent *sport event*
   - GetAthleteWithMostMedals
   - GetAthleteWithMostGoldMedals
   - GetCountryWithMostMedals
   - GetCountryWithMostGoldMedals
   - GetSportAndEventByAthlete *athlete*

You may assume each query is valid (*sport* and *event* exist in the data). If ties exist for most (gold) medals, output ties in alphabetical order.

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

- GetEventsBySport *sport event*1 *event*2 ...
- GetWinnersAndCountriesBySportAndEvent *sport event athleteG*: *countryG athleteS*: *countryS athleteB*: *countryB*
- GetGoldMedalistAndCountryBySportAndEvent *sport event athlete*: *country*
- GetAthleteWithMostMedals *numberOfMedals athlete* [*athlete*2 ... in alphabetical order if ties exist]
- GetAthleteWithMostGoldMedals *numberOfGoldMedals athlete* [*athlete*2 ... in alphabetical order if ties exist]
- GetCountryWithMostMedals *numberOfMedals country* [*country*2 ... in alphabetical order if ties exist]
- GetCountryWithMostGoldMedals *numberOfGoldMedals country* [*country*2 ... in alphabetical order if ties exist]
- GetSportAndEventByAthlete *athlete sport*1: *event*1 ... in alphabetical order [none]

Submission: Submit HW3.java that has the main method, Tree.java, and other program files. Submissions for Indvidual 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.