Simulation using a priority queue

March 20, 2008

1 Project description

Players in a variety of team games (basketball, hockey,volleyball) can be rated using the following method, called *plus-minus rating*.

- If the team scores a goal(basket,point,etc) when the player is in the team, player's rating is increased by one.
- If the opposing team scores a goal, player's rating is decreased by one.

Negative ratings indicate a poor player. The plus-minus rating is preferred over points for goals and assists, because it recognizes that the contribution of defensemen is not in scoring goals.

For this project, you will implement a program that reads in the players' statistics, stores them in the players' records and stores the records themselves in a priority queue. The priority queue will allow the coach to pick a player with the highest plus-minus rating.

2 Priority queues

A *priority queue* is an abstract data type in computer programming, supporting the following three operations:

- add an element to the queue with an associated priority
- remove the element from the queue that has the highest priority, and return it
- increase the priority of a given element by a fixed amount.
- (optionally) peek at the element with highest priority without removing it.

You can (and shoud) implement a priority queue class using linked lists.

3 What your program should do

The program should simulate a game between two teams. Its output should list several events of the following type:

- Team rosters.
- Team starting lineups.
- Goals: "Team [name] player scores".
- Changes: "Player [name] on Team [name] is replaced by player [name]".
- A listing of player statistics (for all players whose ratings change, whenever they change).

4 What to turn in

You will turn in a report containing:

- A description of the significant choices/issues in the design of your code and experiments.
- The results of your experiments, followed by a few lines describing your conclusions.
- The listing (source code) of your programs.

You may turn in the document in class, or via email (gabriel.istrate@gmail.com).

5 Coding standards

A percentage of your grade will be based on the quality of your code, so pay attention to it. Discuss changes (if any) you made to programs presented in class. Take extra care in documenting the code you are implementing on your own. Properly modularize the code (for instance implement separate functions for significant parts of the program).

6 Deadline

Two weeks from now (April 2, 2008, 4PM local time). This is a strict deadline. No credit will be given for homework turned in late.