Matlab Functions for Ranking KHSAA teams

* main()
  + Calls all of the functions and generates the global variables
  + Can change the year and the sport to generate from different years
* loadingMatrices(year,sport)
  + Takes year and sport as parameter (Can also take number of weeks if you wish to isolate this)
  + This function begins by opening the corresponding folders for sports and years
  + Stores number of weeks (whole season if not specified by varargin)
  + Iterates through the list of game files
    - Cuts off after we reach total number of weeks
    - Only stores files that end in .txt
    - Loads info for all the matrices; also stores the next week as well
      * Works for out of state records as well
      * Lastly, we load the number of forfeits
  + Changes any overtime point differentials to ½
  + Returns Game,PD,Loc,GameNW,PDNW,LocNW,OofS,forfeits
    - NW stands for the data next week
* getNameAndRegion(year,sport)
  + Uses readtable function to open and read Shelby’s file
    - Some years were formatted weirdly so we had to go back and fix them
  + Since some teams’ regions are stored as variables and some as doubles, we had to correct the data
    - Use loop
  + Returns Name,Region
* removeTeamsNoGames(Game,Name,Region)
  + We need to remove any teams that have no games
    - Ex: Trimble County or the first few weeks of Trinity’s 2018 season
  + Use sum to add up number of elements in all the columns of the game function
    - If the sum is zero, we remove from Game, Name, Region
      * Since there are no games, there will be no point differentials or locations
  + We also need to iterate through the Game matrix to figure out who won or lost
    - Store the number of wins and losses of each team as an element of that matrix
      * Called WorL
  + Returns Game,Name,Region,WorL
* calculateRecord(Name,OofS,Game,PD,Region)
  + Uses Name, OofS, and Game matrix to determine teams’ records
    - First create a matrix of strings with the in state record
    - Then calculate all out of state games
  + Creates a final matrix of strings that has all the teams’ records
  + Returns Name
* removeForfeits(Game,PD,Loc,Forfeits)
  + Iterates through the Game matrix to find if there were any forfeitures
  + If there were forfeitures, remove game, pd, and loc
  + Returns Game,PD,Loc
* initialElo(Game,firstYear,Elo,Name,OldName,year,Region,sport)
* masseyRating(Game,PD,Loc)
  + Uses the Game, Point Differential, and Location to generate the massey rating for that set
  + Initialize:
    - Home field advantage (hfa) as a point differential
    - pointGap : maximum number of points we will allow a team to score to be counted
  + Iterate through loop
    - Replace huge differentials by pointGap
    - If home team wins, add home field advantage
  + Make Calculations
    - Add a one to the bottom of the Game Matrix
    - Add a zero to the bottom of the Point Differentials
    - M = transpose(Game) \* Game
  + Final Calculation
    - Massey = inv(M) \* transpose(Game) \* PD
  + Returns Massey
* colleyRating(Game,PD,Loc)
  + Initialize:
    - Diagonal matrix with 2s and 0s
    - Vertical matrix with all 1s (this will be the win or loss tracker)
    - Differential (how much a win is worth: max is ½)
  + Iterate through Game matrix
    - Find who played
    - Add one along diagonal to teams who played
    - Subtract one to one off diagonal to teams that played
    - Calculate who wins and loses
  + Final Calculation
    - Colley = inv(diagMat)\*WorL;
  + Returns Colley
* eloRating(Game,PD,Loc)
  + Initialize the rating for every team at the start of the season
  + Initialize parameters:
    - K: max that elo can change as a result of one game
    - S: If S = 400, then a team with 400 more elo points will be 10x as good
    - HFA
      * ROUGH ESTIMATE: 20 Elo Points is one game point (Calculate?)
  + Iterate through loop
    - Find the team and rating
    - Find who won/lost
    - Create a prediction based off of ratings
    - Update the ratings
  + Returns Elo
* sortByRegion(Name,Region,Massey,Colley,Elo)
  + Initialize:
    - An array to store the names of every team in region
    - An array to store the Massey, Colley, Elo ratings of every team in region
  + Iterate through the length of the region
    - Separate into single, double, triple, etc.
  + createTable(Name,Massey,Colley,Elo) [Called within sortByRegion function]
    - Sort Massey,Colley,Elo elements
    - Generates the order for massey, colley, elo for all regions (or whole if called)
    - Creates a table for elements called
    - Returns myTable
  + Returns A1,A2,A3,A4,A5,A6,total