A

Level 1

C

D

B

J

E

Level 2

F

I

H

G

E

K

F

Level 3

K

Q

P

H

I

K

N

M

L

I would like to have you, or me, or some mix of you and me, create a program that can accurately rate lacrosse teams in the state of New Jersey. We currently have 214 high school boys lacrosse teams in the state. Each team plays about 20 games a year. My plan is to use the web of connections between teams playing each other to determine how each team would fair against every other team in NJ.

The tree diagram above is meant to help explain my three different ideas for an algorithm, so first, let me explain what the tree diagram represents….

For this tree diagram, imagine that all the lacrosse teams in NJ played three games**. This is a tree diagram from the vantage point of team A, so all explanations are from the vantage point of team A.** As you can see, Team A has played teams B, C, and D. I call this first level of opponents “Level 1”. You can then see that teams B, C, and D, played teams E,F,G,H,I,J in addition to playing team A. I call this “Level 2”, or the opponents of opponents. Hopefully, you understand what I would mean by Level 3, Level 4, etc.

1) Win algorithm

The win algorithm does not include score at all, rather, it just uses wins and losses to determine if one team would or wouldn’t beat another team. Every “win” gives a team 1 point and every “loss” gives a team -1 points. As there are 214 teams in the state, the highest possible score would be 213 for a team, and the lowest possible score would be -213 for a team.

* Using our example above, if team A beats team B, then team A gets 1 point for its win against team B.
* If team A beats team B and team B beats team E, then team A gets 1 point for a win against team E as we would assume that A beating B and B beating E means that A would beat E.
* If team A beats team B and team B lost to team E, then there isn’t enough information in level 1 to determine who would win between team A and team E. Therefore, we would have to go to the next level where team E appears to determine if team A would or wouldn’t beat team E. Any time we come up with a score of 0 for team A vs another team in a given level, we need to look to the next available level where we can get a nonzero score.
* Notice that team K appears in level 3 a total of three times. Let’s suppose that the following paths occur: A beats B beats E beats K, A beats C beats H beats K, A loses to D loses to J loses to K. In this case, there are three nonzero paths from A to K: two winning paths and one nonwinning path. In this case, A would get (1 + 1 + -1)/3 = 1/3 points for its connection to team K. We average the nonzero paths when we have multiple different scoring paths in a given level.
* If any path from team A to another team includes wins and losses, then the entire path has to be ignored. Only a string of wins along a path or a string of losses along a path can be used to help create a score.
* Once we get to a level where we get a nonzero score between team A and another team, we stop checking levels. The idea here is that more levels add more error.
* For any teams that A does not have a link to in NJ, or for any teams where every level yields a 0 score result, then team A gets 0 points for its connection to that team.
* The 213 scores that come from the links between A and the other teams get added together for team A’s total score.

2) Score algorithm

The score algorithm uses the scores of games and the links between different teams to determine what score one team would beat another team by. To promote sportsmanship, all wins greater than 10 goals will be treated as a 10 goal win. Therefore, as there are 214 teams in NJ, the highest possible score would be 2130 for any given team, and the lowest possible score would be -2130 for any given team.

* Using our example above, if team A beats team B by 4 goals, then team A gets 4 points for its win against team B.
* If team A beats team B by 4 goals and team B loses to team E by 1 goals, then team A gets 3 points for its connection to team E as +4 + -1 = +3.
* Using the same transitive property listed above, suppose that the three paths from team A to team K in level 3 yield scores of -5, +3, and -2. In this case, team A would receive the average of these scores for its connection to team K, i.e. (-5+3-2)/3 = -4/3
* As a reminder, for any given path, the max score is +10 and the min score is -10. This promotes sportsmanship by not incentivizing teams to try to run up a score.
* If any path has a specific branch with a +10 and another branch with a negative number, then this path should be ignored. If any path has a specific branch with a -10 and another branch with a positive number, then this path should be ignored. For example, if team A beat team B by +10 and team B lost to team E by -5, then this path should be ignored rather than giving team A +5 for its connection to team E. If there are truly no other connections between a team other than branches such as the ones I mentioned above, then these branches can be used. Why do I do this? Well, for example, if Delbarton beat Bergen Tech by 20, it shows up as a +10 for them. If Bergen Tech loses to Dwight Englewood by 9, then Delbarton’s connection to Dwight Englewood would be +10 + -9 = +1. This would be ludicrous as Delbarton would, no doubt, destroy Dwight Englewood by significantly more than +1. The max and min goal differentials distort the data.
* Once we get to a level where we can come up with a legitimate score, we stop looking at further levels. The rationale is that each additional level we go to adds more error into the prediction.
* If there is truly no connection between teams, then the score shows up as 0 against that team.
* The 213 scores that come from the links between A and the other teams get added together for team A’s total score.

2) Score-Win algorithm

The score-win algorithm uses the scores of games and the links between different teams to determine if one team would beat another team. To promote sportsmanship, all wins greater than 10 goals will be treated as a 10 goal win. Every “win” gives a team 1 point and every “loss” gives a team -1 points. As there are 214 teams in the state, the highest possible score would be 213 for a team, and the lowest possible score would be -213 for a team.

* Using our example above, if team A beats team B by 4 goals, then team A gets 1 point for its win against team B.
* If team A beats team B by 4 goals and team B loses to team E by 1 goals, then team A gets 1 point for its connection to team E as +4 + -1 = +3 and this +3 goal differential suggests that team A would beat team E by 3 goals.
* Using the same transitive property listed above, suppose that the three paths from team A to team K in level 3 yield goal differentials of -5, +3, and -2. Because the sum of these scores are negative, we would give team A -1 point for its connection to team K, as these scores would suggest that team A would lose to team K.
* As a reminder, for any given path, the max score is +10 and the min score is -10. This promotes sportsmanship by not incentivizing teams to try to run up a score.
* If any path has a specific branch with a +10 and another branch with a negative number, then this path should be ignored. If any path has a specific branch with a -10 and another branch with a positive number, then this path should be ignored. For example, if team A beat team B by +10 and team B lost to team E by -5, then this path should be ignored rather than giving team A +5 for its connection to team E. If there are truly no other connections between a team other than branches such as the ones I mentioned above, then these branches can be used. Why do I do this? Well, for example, if Delbarton beat Bergen Tech by 20, it shows up as a +10 for them. If Bergen Tech loses to Dwight Englewood by 9, then Delbarton’s connection to Dwight Englewood would be +10 + -9 = +1. This would be ludicrous as Delbarton would, no doubt, destroy Dwight Englewood by significantly more than +1. The max and min goal differentials distort the data. This can lead to paths that wrongly give teams a +1 for a win or a -1 for a loss.
* Once we get to a level where we can come up with a legitimate score, we stop looking at further levels. The rationale is that each additional level we go to adds more error into the prediction.
* If there is truly no connection between teams, then the score shows up as 0 against that team.
* The 213 scores that come from the links between A and the other teams get added together for team A’s total score.

For all three of these programs, I’d like the program to spit out an Excel file, or csv, that lists the team names next to their scores. From there, I can rank it and compare it to last year’s rankings to see how accurate we were.