Project 1: Randomized Competitive TicTacOhNo

Due March 2, 2015, 2:00.00000PM (even if your group will be one of the ones presenting on 3/4 rather than 3/2—please note that carefully so you don’t get a zero or sharply lowered grade after having done lots of work!!!); also, fyi, that the first intermediate workshops on this will be 2/9 and 2/11, and the second intermediate workshops on this will be 2/23 and 2/25

UR’s President Seligman has a problem. He has been challenged by RIT’s president, Dr. William W. Destler, to provide an evaluation of a certain “game” (which we’ll call TicTacOhNo, and which we will describe in detail below). Being no fool, President Seligman, after inspecting the description President Destler provided him of the game, suspects that in fact the RIT president has in mind some actual scenario in which randomly chosen land cells, from a square grid, are awarded to one and then the other of two players, with whichever player ends up with the most adjacent-and-in-a-line along a row, column, or any diagonal being able to build the better version of some facility that is better when it is longer (a bowling alley? a linear accelerator? an archery range?...a rail gun? who knows what that crafty RIT president is thinking of). Nonetheless, filled with pride at the brightness of UR’s students, President Seligman has agreed to undertake the analysis and has, of course, delegated it to his most powerful secret resource—the students of CSC200/200H.

Your task (each group’s task, that is) is to write (following the rules of the course) a research paper exploring this game and in particular the issues mentioned below. Ideally, your paper will be written in \LaTeX, since is flexible and powerful and the norm for scientific papers (and you can build in pictures/images/etc.; see the book “The \LaTeX Graphics Companion” for how); however, if you really must, you may also write it in some other system (such as Word or any other decent system). (Note: I will put up on the course web site one of my own \LaTeX papers, as source code, so that you can use it as your starting point, if you wish.) As mentioned above, we’ll call the game TicTacOhNo.

The game TicTacOhNo is played on an $N$ by $N$ board ($N$ can be 1, 2, 3, etc.). To be careful and build the $N$ into the name of the game, let us during this project when being careful write TicTacOhNo$_N$ when speaking of the $N$ by $N$ version. Let us call the first player RIT and the second player UR.

Here is how the game goes, and how one determines who wins. During the first step, RIT chooses a random cell (each with equal probability, e.g., the player flips a fair $N^2$-sided die) and puts a marker labeled RIT on it. During the second step, the other player, UR, chooses a random cell from among all cells that do not yet have a marker on them (each chosen with equal probability, e.g., the player flips a fair $N^2 - 1$-sided die; and I’m not intending any independence tricks here, so you can use your natural intuition as to the flips being independent) and puts a marker labeled UR on it. And so on, with the players taking turns: RIT, UR, RIT, UR, .... The game ends when the board is completely filled with markers. At this point, whichever player has the most markers adjacent along a row, column, or diagonal wins; if both tie for most, the game is a tie.

Oh... just so I don’t trick you, the “sequential” nature of the above game-play and definition of the game is just for dramatic effect, and so is a red herring of sorts (shame on you, President
Destler—how dumb do you think UR people are?). Really, all you have to do, I believe, is keep in mind that each of the \( \binom{N^2}{\lfloor N^2/2 \rfloor} \) ways that one can mark \( \lceil N^2/2 \rceil \) of the cells of an \( N \times N \) matrix “0” (meaning “RIT”) and mark the rest “1” (meaning “UR”) is equally likely as an outcome (unless I myself am confused regarding this, but I don’t believe that I am as this is bright-line clear; and of course you in your papers may well want to state and prove this; note that this also simplifies and speeds up your attempts at brute-force (or perhaps other) exact computations of values for particular \( N \)'s, wildly (although perhaps just from super-insanely slow to just insanely slow), since you just have to deal with each legal final board configuration once, rather than worrying about each of the literally exponential number of ways that it can be reached).

By “diagonal” in this game, we mean not just the “major diagonals,” but all of the natural diagonals. So for \( N = 1 \) there is 1 diagonal, although this is sort of a degenerate case. And for \( N \geq 2 \), there are exactly \( 2(2^n - 1) \) diagonals. For example, on a 3 by 3 board, and to make things clear in this example let us number the cells as follows: 

\[
\begin{array}{ccc}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{array}
\]

the 10 diagonals are these collections of cells:

1. 1
2. 4 2
3. 7 5 3
4. 8 6
5. 9
6. 3
7. 2 6
8. 1 5 9
9. 4 8
10. 7.

And of course, we are concerned with adjacency. So if UR has markers on 7 and 5 but not 3, that is 2-in-a-row (the “row” here not really meaning row, in this case; in fact—introducing a notation that we all should use (but define it in your papers for clarity of course)—let us call a 2-in-an-adjacent-row/column/diagonal of your marker a “2-chain”, and similarly for each value \( k \), \( 1 \leq k \leq N \)), but if UR has markers on 7 and 3 but not 5, the best UR can claim regarding that diagonal is a 1-chain.

So for example, if the game ends with the board looking like this: 

\[
\begin{array}{ccc}
\text{UR} & \text{RIT} & \text{UR} \\
\text{UR} & \text{UR} & \text{RIT} \\
\text{RIT} & \text{RIT} & \text{RIT}
\end{array}
\]

RIT has 3-chain (due to the bottom row) as its best, and UR achieves as its best claim that of a 2-chain (in exactly 4 different ways, by the way; please make sure you see them, as a check that you understand the rules of this game). So RIT wins. If the board had instead been, and this
is on a 4 by 4 board:

\[
\begin{array}{cccc}
UR & RIT & UR & UR \\
RIT & UR & UR & RIT \\
UR & RIT & RIT & RIT \\
RIT & UR & RIT & UR \\
\end{array}
\]

3-chain (it achieves that exactly two ways) and UR has as its best success the claim of a 3-chain (it achieves that exactly one way). And so this play of the game is a tie. (The fact that RIT achieved its victory in more ways doesn't break the tie in its favor; the way game was defined by Dr. Destler simply doesn't take how many ways you achieve your “best” into account. Apparently, he thinks each school will seek to build at most one of whatever the secret “linear” facility is going to be.)

We are primarily interested in the following questions, which we may at times call our “focus” questions.

1. What can we say about the probability that the game ends in a tie?

2. What can we say about the probability that RIT wins?

3. What can we say about expected (in the sense of averaging over all possible game-plays) value of the winning/tieing chain length?

So, in this project, you should study some or all of these issues. Note that there are many other questions to potentially explore (e.g., what is the lowest possible winning chain length in TicTacOhNo\(_N\)?; what is the lowest possible tieing chain length in TicTacOhNo\(_N\)?) and you may in addition look at other questions if you wish. But the core goal here is to gain insights into some or all of the three above issues.

It will be great if you can state and prove crisp theorems. Perhaps you'll completely resolve some issues. If not, perhaps you'll prove upper and lower bounds, and will work hard to make them as strong as possible. Or if on an issue you are failing to get any traction, maybe you'll program up a simulation, and see what insights that gives. Also, note that these issues can be studied in various ways. One issue is what values hold for a particular, fixed \(N\), e.g., \(N = 1\) (pretty easy!), or \(N = 2015\), or \(N = 2\), or etc. Very nice would be to get a precise formula that holds over all \(N\), e.g., “For each \(N \in \{1, 2, 3, \ldots\}\), the expected length of the longest chain in TicTacOhNo\(_N\) is \(N\)” (which by the way happens to be an outright theorem for \(N = 1\) and \(N = 2\)—so you see, “theorem” isn’t as intimidating a word as one might think; go prove some!... but of course that claim provably fails for each \(N > 2\)... oh wait... that is a theorem of sorts too; see... more evidence that proving theorems isn’t something to be spooked by!). But if you can’t get a precise formula, you might want to deal with the asymptotics (e.g., “ExpectedWinTimeInTicTacOhNo\(_N\) = \Theta(BLORT),” where for BLORT will be some expression in terms of \(N\) that you have brilliantly proven to hold in your paper, or perhaps you’ll not get a \(\Theta\) result but will provide \(O\) and \(\Omega\) results; or maybe you’ll prove that the probability of a tie asymptotically goes to 0, or to 1; or maybe you’ll prove that the probability that RIT wins goes to 1, or that it goes to 1/2, although if you show that that goes to 1 and the probability of a tie goes to 1/2, you probably should recheck your proofs!; by the way, these problems are the types of things where programming up a simulation might give you some informal sense of what the correct theorem to seek to prove might be; as a side comment, perhaps you’d like to, as nice little part of your paper, give a proof that the probability that RIT wins is greater than or equal to the probability that UR wins... although you certainly will also, if you do that, want to note that when \(N\) is even, UR and RIT obviously have the same chance of winning), or perhaps even if you don’t get an asymptotic result you will find/prove upper and lower bounds
on particular $N$s. What that last comment is mentioning is that, as something very concrete, you might look at finding the exact answers to some or all of the three focus questions for as many values of $N$ as possible. For example, for $N = 1$ the answer to the 3 questions are 0, 1, and 1; for $N = 2$ the answer to the 3 questions are 1, 0, and 2; and I'll bet that every group will very easily get the answers for some additional values if the group tries to... but I'd also guess that among those groups that get exact values, different groups will differ sharply in how far they can go and still get exact answers to some or all of the questions (and what will make the difference?... perhaps using symmetries will make things simpler?... perhaps other insights in how your approach this might help?... perhaps excellent data structures and clever programming will help as to implementing well your approach?... I do realize that there is also the issue of how much computing power you and bring to a problem, but please do make sure to not, for example, violate any of the department’s rules/guidelines as to how to use its computers).

Don’t despair if you find the problems extremely hard or nearly impossible. They probably are impressively hard—but that doesn’t mean you won’t be able to learn things that address them, or special cases of them.

Basically, poke around, do your best to converge to strong results by proving at first what you can, and then strengthening more and more and more. We’ll discuss the flavor of this in class a bit on the starting date. And of course, as always, we’ll before the presentation day have two intermediate day-pairs (February 9 and 11, and February 23 and 25) during which you share, present, defend, discuss, etc. your ideas/progress/etc. (e.g., the first four groups—in writing that I’m assuming that the number of groups will be 7, otherwise the split might differ—will speak on 2/9, and the others will speak on 2/11). Your group’s hand-in must be sent in using the rules in the course information document (so one group member will be sending in, all zipped together in a single email to cs200 at cs.rochester.edu that is cc’ed openly to every group member including the person doing the emailing, your paper, and the slides you’ll be using in your 3/2 or 3/4 talk, and possibly code and sample runs; that email must arrive by 2:00.000000PM—no extra seconds—on 3/2, even if your group is one of the ones speaking on 3/4, as if it is late the extremely harsh lateness penalties from the course information document will be applied, strictly; and do read the hand-in rules in the course information document as there are lots of things you must do, e.g., including also a good pdf version of your slides if your slides are in PowerPoint).

In what order will the houses will go? On both intermediate workshop day-pairs and at the final conference presentation day pair, we’ll go in alphabetical order by house name, so the first four houses will go on the first day and the rest will go on the next day. (Note that your group will come in to each of your intermediate workshops and your final conference day with a slide presentation, timed to take about 15 minutes (even if there are questions), that your group will present to the class. And, again, as per the course information document, you’ll also email those slides to “cs200” by 2PM on the day of your intermediate talk for intermediate talks but by 2PM on 3/2, along with your paper, for your conference talk. And of course, at each of the workshops and the conference talks that you give, as per the course information document, do bring in to class 4 printouts—1-up but 2-sided is ok, and of course each stapled together—so that the TAs and I can have those in hand.)

Good luck, and do well. President Seligman has put the honor of UR in your hands!