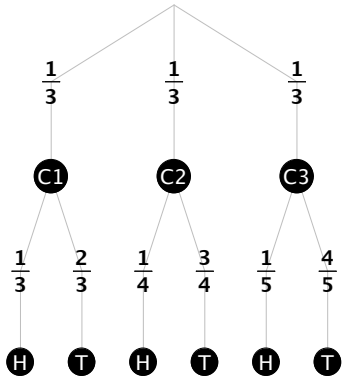
CS MS Intake Exam

Name: _____

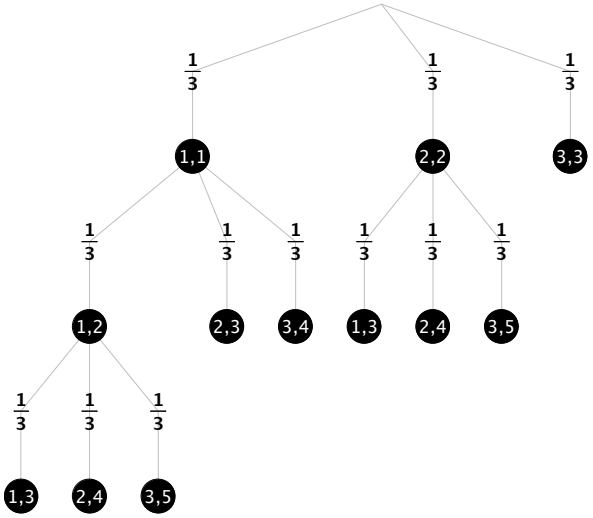
Instructions:

- *Answer each question the best you can.*
- *It is your responsibility to ensure your code is understandable; if we can't understand how your code works or why some part is written the way it is, you may not receive credit, even if the code technically works. Write neatly and organize your code properly.*
- *For coding questions, you may use any* real *programming language of your choice, but you may not use pseudocode. You may use the standard library of your language of choice.*
- *This is a closed-book, no computer, no calculator, no notes, no smartphone, no neighbor, do-it-all-by-yourself exam.*
- *If you need extra space, you may use the back of the page, provided you indicate this clearly.*
- ***Read all of the questions before answering any of them.***
- *If you have a question, please ask a proctor.*

One way to conceptualize a randomized experiment is to break it down as a series of steps, with each step having its own probability. We can organize these possiblities into a tree to visualize all possible outcomes of the experiment and track which options have which probability of happening. For example, here's an experiment where a person selects (with equal probability) from one of three biased coins and then flips that coin (with "H" and "T" referring to the two outcomes of the event of flipping a coin, namely that it lands on the "heads" side or the "tails" side).



The numbers on the edges leading out of any given node sum to 1; they indicate the probability that that given edge is taken (once we've reached that node). So, we can see the choice of the coin is fair, but the coins themselves are not fair (and also different from each other). Here's another example where a person repeatedly rolls a fair three-sided die until the sum of the numbers rolled is at least three. The nodes show the individual die rolls and the sums, separated by a comma.



1. Design a data structure or data type in the language of your choice for representing randomized experiment trees (as described on the previous page). Don't include any functions or methods in your design at this point, because we just want to understand how the information is represented. (In some languages you'll have to write a constructor.)

Hint: **Read the rest of the exam before writing anything on this page.**

2. Demonstrate how to represent the **first** example from p. 2 using your data type from p. 3. That is, show how to construct a value or instance representing that example.

3. Design a function or method on randomized experiment trees that counts the number of different terminal nodes (also known as leaf nodes).

   For example, given the first experiment on p. 2, the result should be 6 (because there are 6 nodes where the experiment ends; they are the ones labelled “H” and “T”). For the second experiment on p. 2 the function or method should return 9.

4. Design a function or method that is given two arguments, a randomized experiment tree and a sequence of events (represented as a list or a vector, as you choose). It must return the liklihood of reaching the corresponding node in the tree. Note that the reached node might not be a terminal/leaf node. Also, if one of the given outcomes is not possible, the result should be 0.

   For example, given the first example on p. 2 and the sequence “C1”, “H”, the result should be 1/9. Given “C3”, “T”, the result should be 4/15. Given the second example on p. 2 and the sequence containing “2,2” and “1,2”, the result should be 1/9.