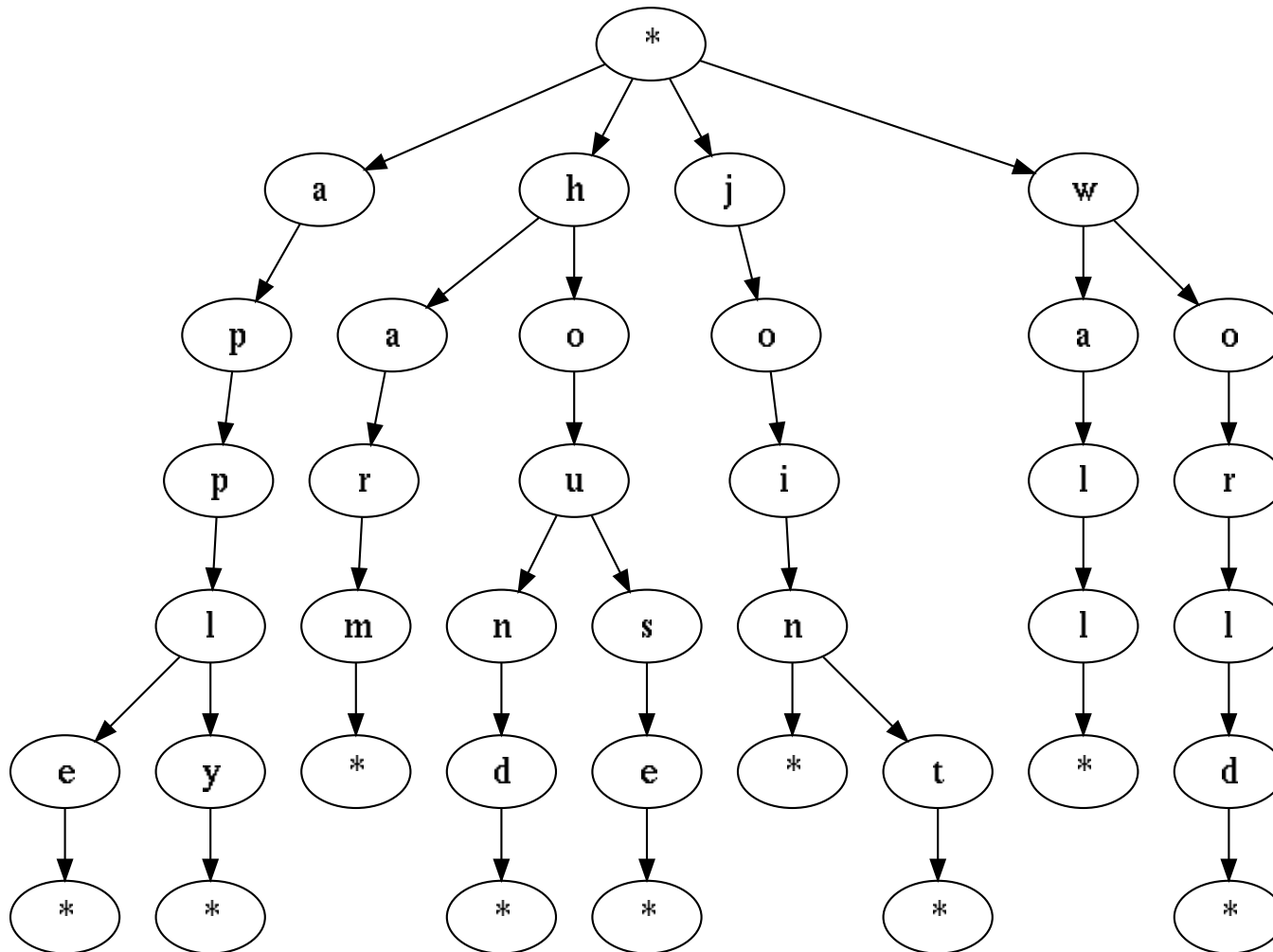


GHOST: A Multiplayer Word Game

Fall Semester

Topic Outline

- Representing the Word List
 - Trick, use a tree
 - Game tree equivalency
- Statement of the Problem
- Human Players
 - Module for picking a move
 - Handling challenges
 - Keeping score
 - More than two players
- Knock Out Correctly
- Computer Players
 - Suggest a move
 - Play one-on-one
 - Play selfishly
 - Team up together
 - Minimal word list
- Up Next: Tic-Tac-Toe



Game Tree

- apple, apply, harm, hound, house, join, joint, wall, world
- Trick, this method of storing the word list is equivalent to the game tree for the game GHOST. In general we will not have a concrete representation of the game tree but will instead generate only portions of the tree dynamically.

Statement of the Problem

Players alternate naming letters which are appended to a string. If a player names a letter that forms a word, or if he or she names a letter that makes it impossible for the string to eventually form a word, and the next player challenges, then the offending player receives a letter (in order, G-H-O-S-T) and the challenging player begins a new string. If a player challenges incorrectly then he or she loses the round and the next player begins a new string. If a player receives all five letters to spell GHOST then he or she is out. Play continues until only one player is left, the winner.

Example Round

Player 1: C

Player 2: C-A

Player 3: C-A-T

Player 1: C-A-T-E

Player 2: C-A-T-E-G

Player 3: C-A-T-E-G-O

Player 1: C-A-T-E-G-O-R

Player 2: C-A-T-E-G-O-R-I

Player 3: C-A-T-E-G-O-R-I-C

Player 1: C-A-T-E-G-O-R-I-C-A

Player 2: C-A-T-E-G-O-R-I-C-A-L

Player 3: *challenge*

- Player 2 gets a G and Player 3 starts a new string.
- Words must have $len > 3$ so Player 3 doesn't lose on CAT.

Human Players

- Either begin with one-on-one play or start right away with the general case of N players (you can always set $N = 2$).
- For each round a pointer begins at the top of the word tree (game tree) because any word could be spelled. As letters are added the pointer descends downward through the tree as fewer and fewer words become reachable.
- If the players are really bad and the pointer leaves the tree entirely, and without challenge, just set a flag that the current string is invalid and wait for someone to challenge.

Module for Picking a Move

- Important coding matter, within the moderator loop make sure the selection of the next move is implemented as a function call so that later we can replace calling the human module with a call to the computer module.
- Python treats functions as first-class objects so you can store a list of references to functions (just use the function name with no parentheses) only one of which will eventually be the human module, plus any number of computer strategies.

Handling Challenges

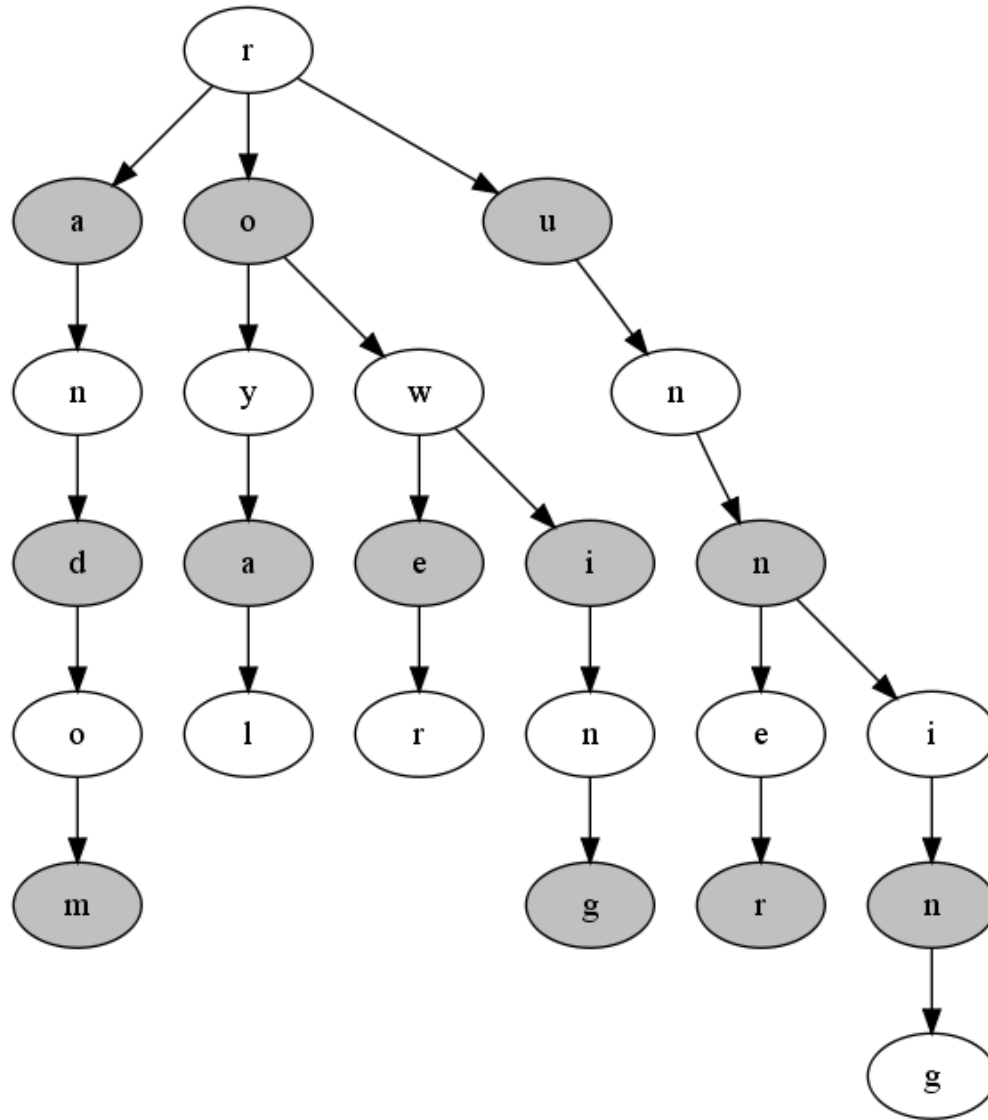
- If a player challenges then two cases are possible:
 - That player believes the current string is a word.
 - That player believes the current string could not form a word.
- In the first case the moderator can simply respond yes or no. If it is a word then the previous player loses the round and the challenging player starts the next round.
- In the second case the previous player must name a word. If they succeed then the challenging player loses the round and the next player after them starts the next round. If they fail then the challenging player starts the next round.

Keeping Score

- Each time a player loses a round they receive another letter in the word G-H-O-S-T. If a player gets all five letters then they are out.
- Also, more than two players, then transition correctly from round-to-round and knock out correctly.

Suggest a Move

- Cheat feature for human players, strategy for computer players.
- Primitive, letters that immediately lose, look one step ahead.
- Better, search the rest of the game tree (word tree).



Computer Player: Selfish (1)

- random royal rower rowing runner running
- Player One's letters in white, Player Two's gray. Assume that Player One has already chosen R. Then, should Player Two choose A, O, or U?
- And how does Player Two determine this? Recursion!!!
- Easy base case, keep track of whose turn it is. Assume all words are challenged. So, whoever's turn it is on a word, they lose.
- So, Player One will lose on royal, rower, and running.
- And Player Two will lose on random, rowing, and runner.

Computer Player: Selfish (2)

- During our recursive search the return value indicates who loses if we follow this branch of the game tree.
- If there is only one letter to choose then the move is forced and we simply return the recursive result. For example, when we analyze what happens if they choose A then there are no decisions to make, the base case says Player Two loses (by spelling RANDOM), and each intermediate recursive call must return the fact that Player Two has lost at the end of this branch.
- Note, this can also be handled as a special case of the loop described next.

Computer Player: Selfish (3)

- If there are multiple letters to choose from, at any point in the search, then the current player will choose a path that is most favorable to themselves.
- Read the previous sentence again. This is the key idea of the recursive search.
- So, we loop over each available letter, make a recursive call to determine who would lose if we follow that branch, and then there are two possibilities:

Computer Player: Selfish (4)

1. If our opponent loses then we can immediately stop the loop and return that they will lose, that we will select the current letter in the loop, and we do not need to check any other letters.
 2. If we lose instead then we must continue the loop and try the next letter. If none of the letters available to us results in our opponent losing then we are forced to return that we will lose, the letter we select is irrelevant, and this return statement happens after the loop is over.
- As shown, Player One prefers E to I but can't win with either Y or W, and Two also prefers E to I and ultimately chooses O.

Lab Assignment: GHOST

- First, build the word tree from the dictionary file.
- Then, moderate a two-player game between humans.
- Option, allow players to add words to the dictionary.
- Multiplayer humans (or just do this from the start).
- Computer hints or suggests a move (cheat feature).
- Selfish, as described plus pruning, two-player game.
- Strategic, team-up in a multiplayer game. Not easy!
- Bluffs! Deception! See also, the Turing test.
- Start with minimal word list \Rightarrow learns.