CMSC 420 Fall 2024: Coding Project 5
Compressed Tries

1 Due Date and Time

Due to Gradescope by Sunday 17 November at 11:59pm. You can submit as many times as you wish before that.

2 Get Your Hands Dirty!

This document is intentionally brief and much of what is written here will be more clear once you start looking at the provided files and submitting.

3 Assignment

We have provided the template `trie.py` which you will need to complete. More specifically you will fill in the code details to manage insertion, deletion, and search in a compressed trie. More details are given below.

4 Details

The class methods should do the following:

- `def insert(self,word,value):`

  Insert the word `word`, `value` pair into the compressed trie.

- `def delete(self,word):`

  Delete the `word` (and its associated `value`) from the compressed trie. The word is guaranteed to be in the compressed trie.

- `def search(self,word):`

  Search for the `word` in the compressed trie and print the associated `value`. The word is guaranteed to be in the compressed trie.

5 Additional Functions

You will probably want some additional functions as well as helper functions to handle the necessary operations.

6 What to Submit

You should only submit your completed `trie.py` code to Gradescope for grading. We suggest that you begin by uploading it as-is (it will run!), before you make any changes, just to see how the autograder works and what the tests look like. Please submit this file as soon as possible.

7 Testing

This is tested via the construction and processing of tracefiles.

- Each non-final line in a tracefile is either `insert,word,value` or `delete,word`. All together these lines result in the creation of a compressed trie.
- The final line is either `dump`, which dumps the compressed trie, or `search,word`, which finds and prints the `value` associated to the `word`.

You can see some examples by submitting the `trie.py` file as-is.

8 Local Testing

We have provided the testing file `test_trie.py` which you can use to test your code locally. Simply put the lines from a tracefile (either from the autograder or just make one up) into a file `whatever` and then run:

```
python3 test_trie.py -tf whatever
```