

ITU Computer Engineering Department  
BLG 223E Data Structures, Summer 2021  
Homework #6  
Due August 5, 2021 23:59

### Definition

You are asked to implement a program that builds an index for a text file using a trie structure. Please pay attention to the following details in your implementations:

- You should implement a character-by-character trie that follows the principles in your text book Chapter 12.5.1. You are not asked to implement compressed tries or suffix tries.
- **You are asked to implement a trie data structure, other possible solutions will not be graded!!!**
- A large timeout value will be given, so please do not focus on the efficiency of your code.
- Your trie should be constructed **only for** lowercase English alphabet letters. You may use the code below or similar to eliminate all the remaining characters:

```
void Trie::add_word(string s, int page){
    string s_low="";
    for(char c:s){
        if(tolower(c)<'a' || tolower(c)>'z' )
            continue;
        else s_low+=tolower(c);
    }
    if(s_low.length()==0)
        return;
    this->root->add_rec(s_low, page);
}
```

### Input-Output

Your program is going to accept an input txt file name, followed by a number of words from the command line and produce line numbers that of the text file for each parameter word. If the word is not present a single -1 will be output. If the word is present ordered list of page numbers will be output. For more details please check out the provided test files. The input will be syntactically and semantically correct.

### Deliver

Please zip and deliver the directory structure defined below:

- HW6: Topmost folder, that will contain all the folders in your submission. No other files should be present under this folder in your submission.
- HW6/src: Contains all the \*.cpp files
- HW6/src/main.cpp: Contains your main function and other code you want to deliver.
- HW6/src/Trie.cpp: Contains the code for your trie.
- HW5/src/TrieNode.cpp: Contains the code for your trie nodes.
- HW5/include: Contains all the header files you use
- HW5/bin: An empty directory that will contain objective files when your project is compiled

Please check the calico test file in the homework definition to see how your files will be compiled and tested.

## Restrictions and Guidelines

- Compilation environment: Only the code that can be compiled by the environment of the container definition provided in ninova will be accepted.
- Testing of your program will be performed using Calico (<https://calico.readthedocs.io/en/latest/>). Test cases that will be used to test your homework is provided as an attachment in ninova.
- **STL usage is allowed except for the trie data structure. Using vectors, sets, maps, etc. is alright.**
- **This homework is for individual submissions.** Any kind of code sharing or code adaptation from an external source is strictly forbidden. Submitted code will undergo a plagiarism check process and plagiarism penalties may be given if the submitted code's similarity is approved by the instructor.
- Make sure you write your name and number in all of the files of your project, in the following format:  
/\* @Author  
Student Name:<studentname>  
Student ID :<studentid>  
Date:<date>\*/
- Only electronic submissions through Ninova will be accepted no later than deadline
- Use comments wherever necessary in your code to explain what you did.