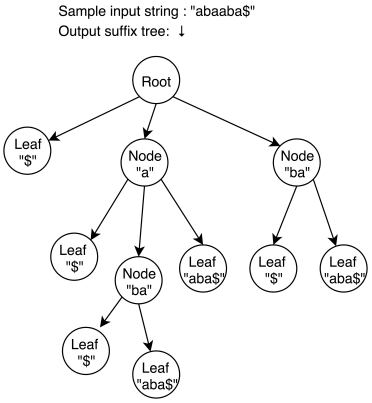
Project title: Suffix Tree provided in Sample project 2 (Bioinformatics)

Project members: dhuang(Di Huang), yhe28(Yifan He)

Accomplished:

- Everything proposed in the project proposal except MUMs
- Two approaches to build a suffix trie
  - Create suffix trie directly using tuple and map
  - Generate a new suffix trie by inserting a new suffix string into the old one (start with Root [])
- Convert suffix trie into suffix tree
- Indexed suffix tree
- Find whether a given string s0 is a substring of s1



Not completed: MUMs

Code accounting: 229(SfxTree.hs) + 37(Main.hs) = 266

Sample inputs and outputs:

Run main and main2 functions to check the sample outputs:

```
*Main> main
----------------------Simple Test-----------------------
------------------------Testing-----------------------
Input String: abaaba
Parse: abaaba$
Index: [6,5,4,3,2,1,0]
...
```

```
*Main> main2
-----------------------DNA Test-----------------------
------------------------Testing-----------------------
Input String: gggacaatacgtca
Parse: gggacaatacgtca$
Index: [14,13,12,11,10,9,8,7,6,5,4,3,2,1,0]
...
```

Also, we can type “build <input string>” to check the result:

```
*Main> build "gacttg"
------------------------Testing-----------------------
Input String: gacttg
Parse: gacttg$
Index: [6,5,4,3,2,1,0]
...
```

In addition, “sfxtrie1 $ parse <input string>” or “sfxtrie2 $ parse2 <input string>” can generate a suffix trie of the input string; “sfxtree1 <input string>” or “sfxtree2 <input string>” can generate a suffix tree of the input string; “indexTree <input string>” can generate a indexed suffix tree of the input string; “insertTrie <new suffix> <old suffixTrie>” can generate a new suffix trie after suffix insertion; “trieToTree <suffixTrie> <suffixTree>” can convert a suffix trie into a suffix tree; “isSubstr <string A> <string B>” can find whether string A is a substring of string B.

Instructions for running the code:

Only Need: **ghc**, emacs (ghc is enough actually since we can run the code via terminal or cmd)

Details: just go to the directory of the source fils and type following commands:

```
jk:production dihuang$ ghci
GHCi, version 8.2.2: http://www.haskell.org/ghc/  :? for help
Prelude> :l Main.hs
[1 of 2] Compiling SfxTree          ( SfxTree.hs, interpreted )
[2 of 2] Compiling Main             ( Main.hs, interpreted )
Ok, two modules loaded.
*Main>
```