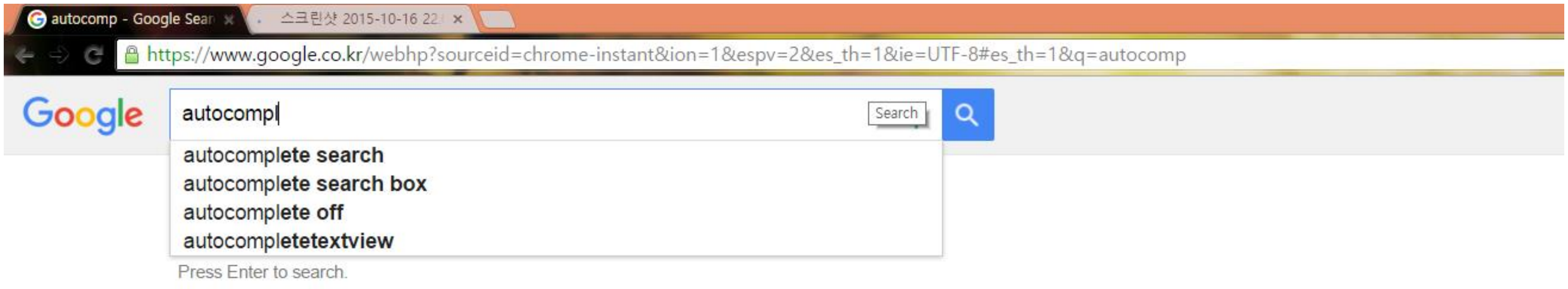
430.217
Introduction to Data Structures

Assignment 3. TRIE

Seoul National University

Advanced Computing Laboratory

실습과제 : Autocomplete 구현


autocomp - Google Sear
스크린샷 2015-10-16 22
https://www.google.co.kr/webhp?sourceid=chrome-instant&ion=1&espv=2&es_th=1&ie=UTF-8#es_th=1&q=autocomp
Google
Search

autocompl

autocomplete search
autocomplete search box
autocomplete off
autocompletetextview

Press Enter to search.

실습과제 : Autocomplete 구현

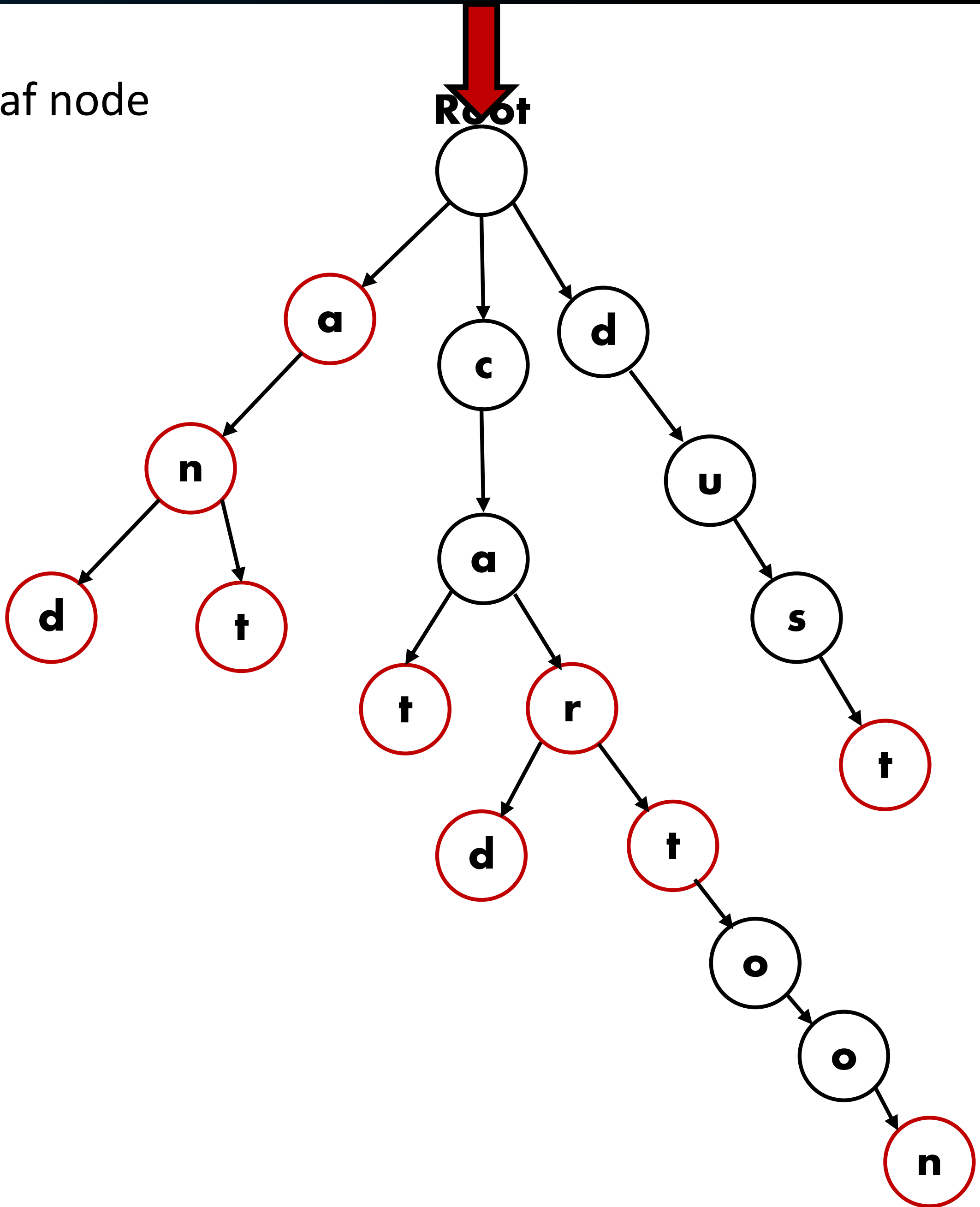
- Calling every child node to leaf node

Input

cart

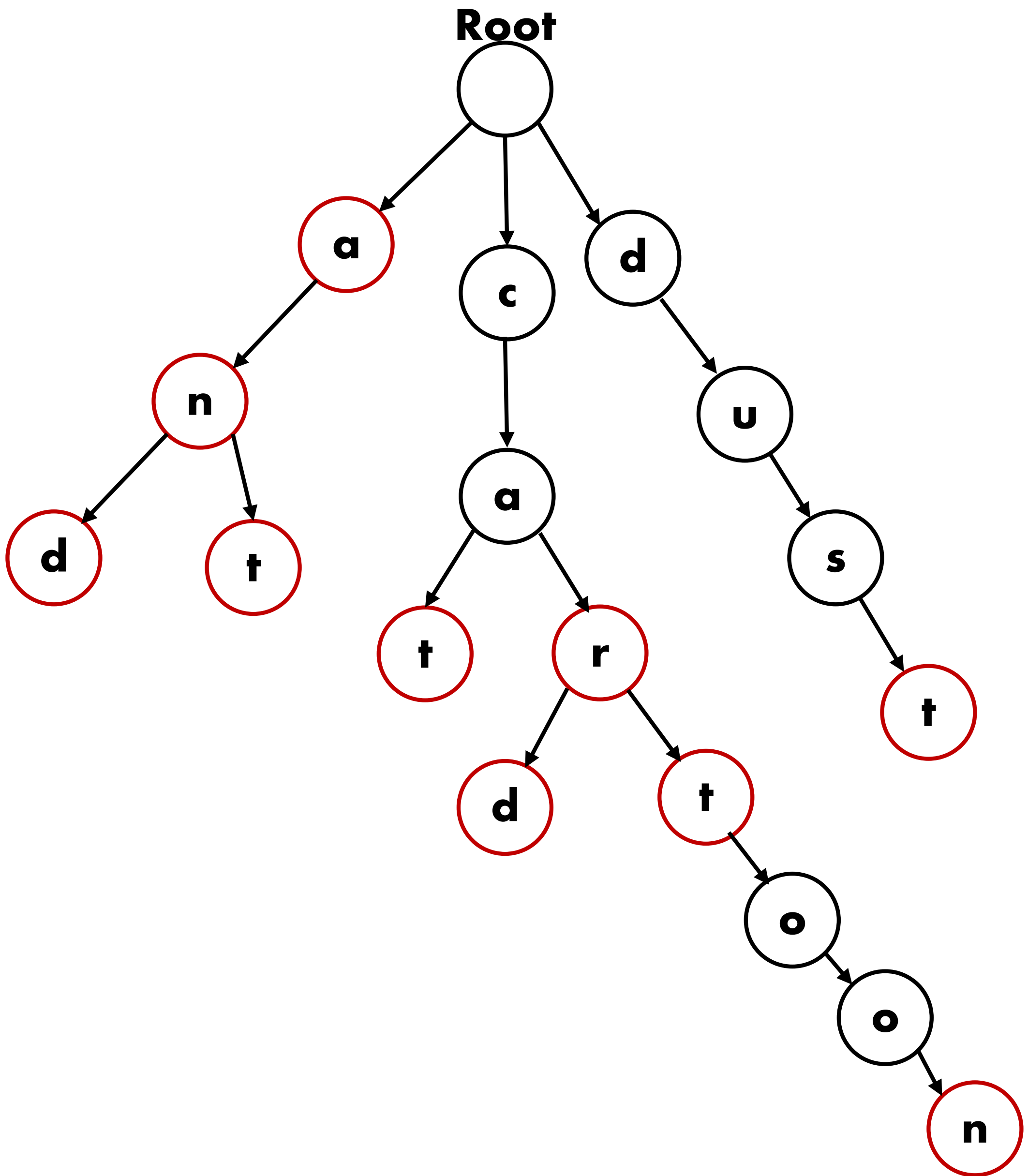
Suggested words

cart
cartoon



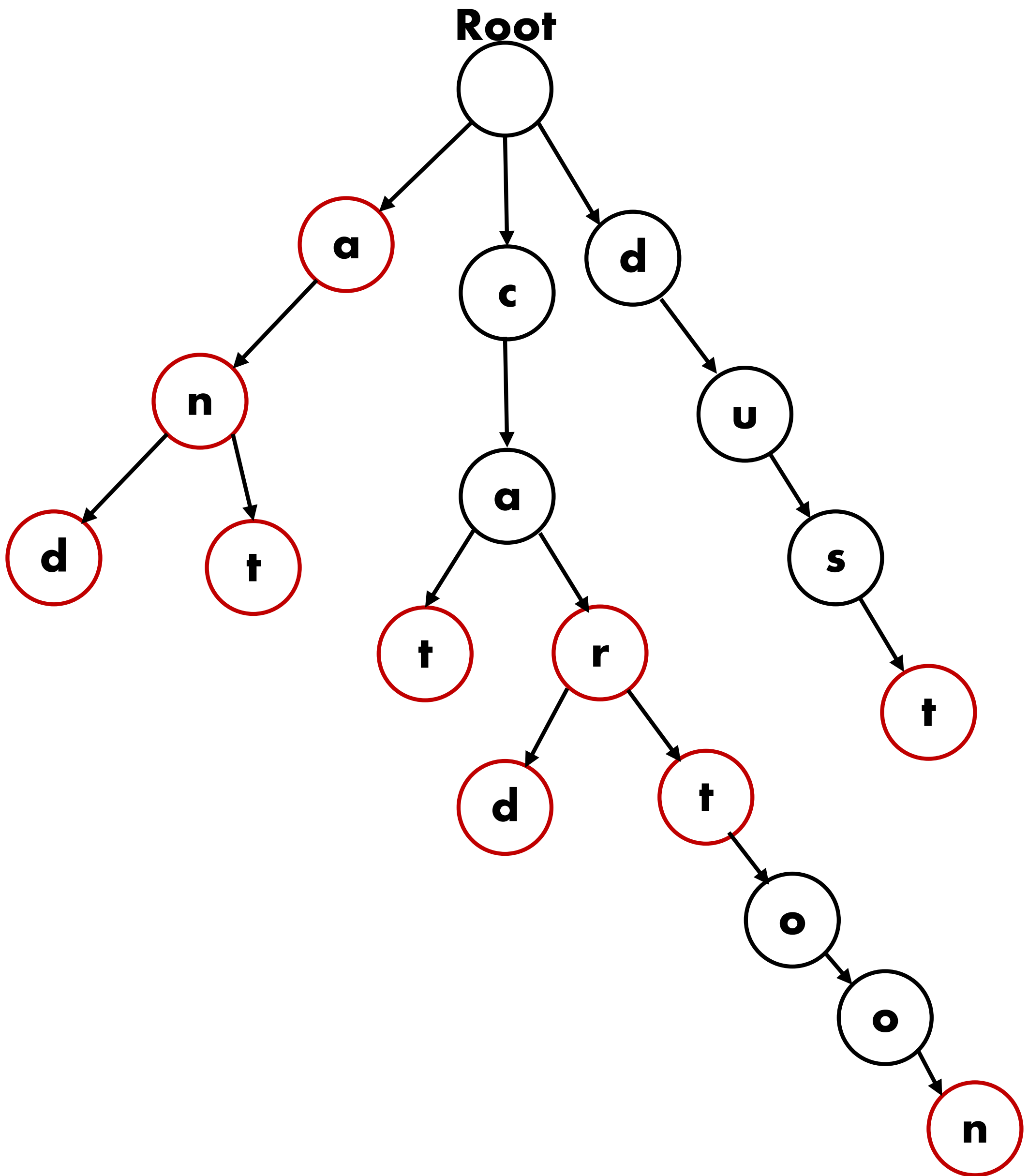
TRIE

- Data structure
  - From retrieval
  - Digital tree, prefix tree, …
- Analysis
  - word length : M
  - Insertion: O(M)
  - Search : O(M)



TRIE

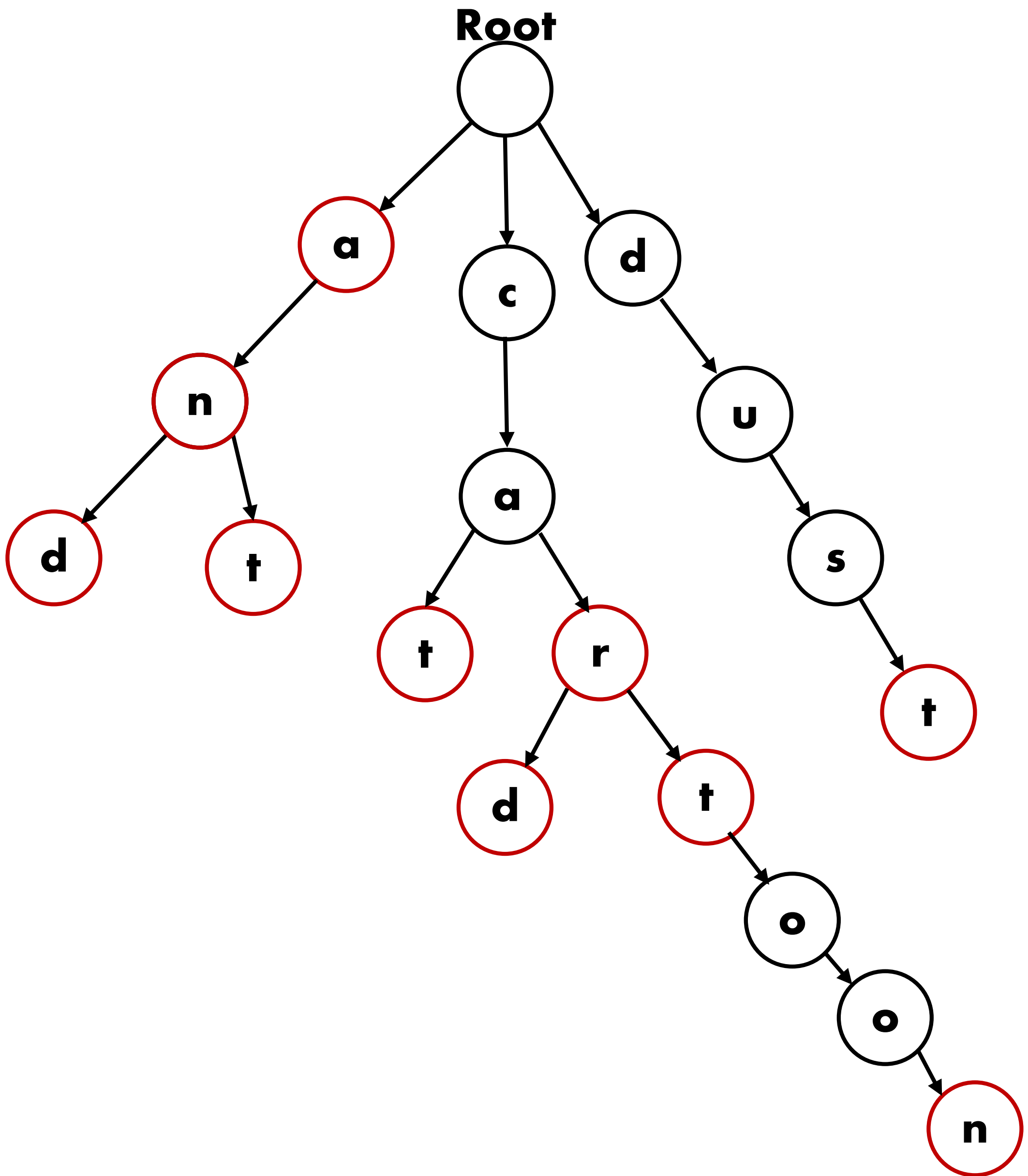
- Functions
  - Insertion
  - Deletion
  - Print



TRIE function : Insert

Insert

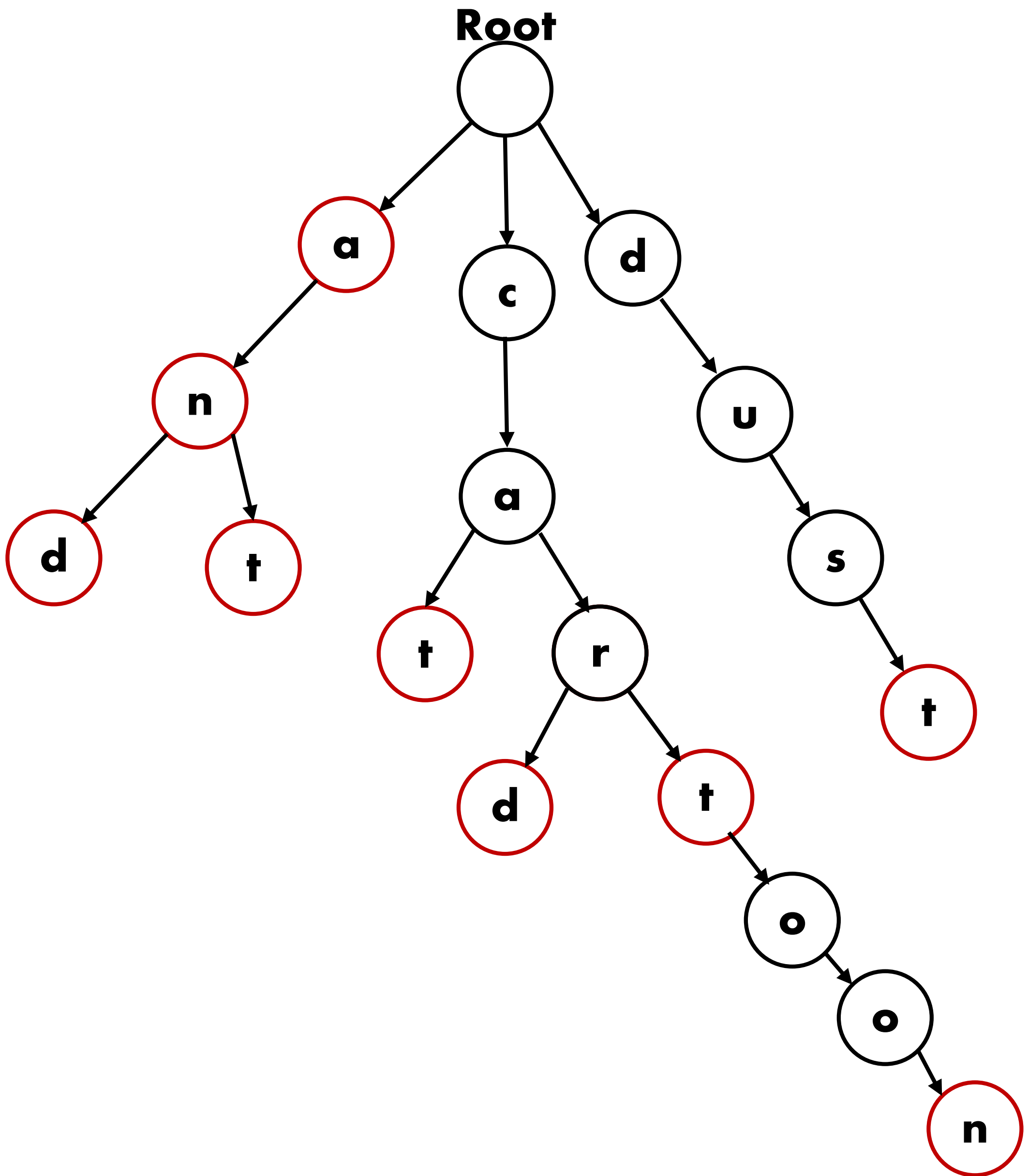
and



TRIE function : Delete

Delete

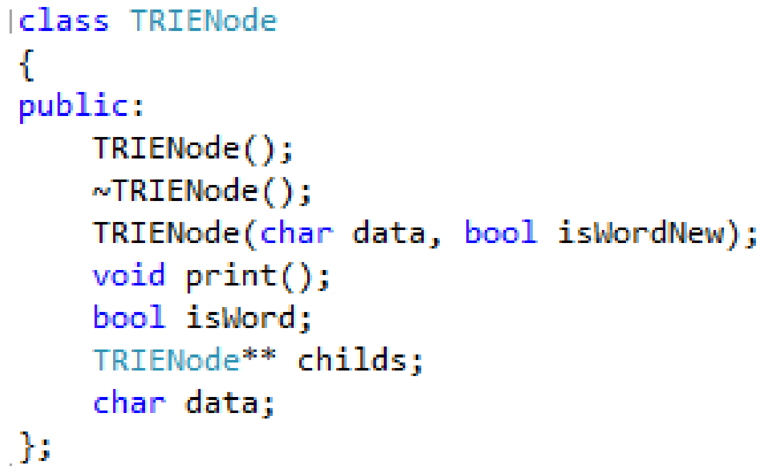
card



TRIE Node

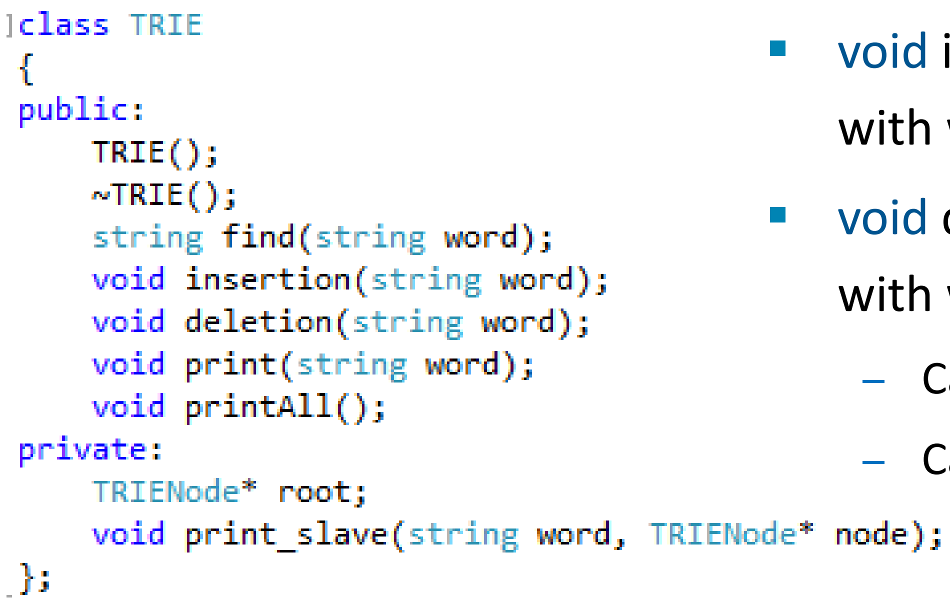
- Member variables
  - TRIENode** childs : childs with number of alphabet
    - Dynamic allocation (length = 26)
  - bool isWord : the variable to check whether the word exists
  - char data : for convenience saving the alphabet of current node

- Member function
  - void print() : print the data of the node



```
class TRIENode
{
public:
	TRIENode();
	~TRIENode();
	TRIENode(char data, bool isWordNew);
	void print();
	bool isWord;
	TRIENode** childs;
	char data;
};
```

TRIE Implementation



```
class TRIE
{
public:
    TRIE();
    ~TRIE();
    string find(string word);
    void insertion(string word);
    void deletion(string word);
    void print(string word);
    void printAll();
private:
    TRIENode* root;
    void print_slave(string word, TRIENode* node);
};
```

- void insertion(string word) : inserting node with word
- void deletion(string word) : deleting node with word
  - Case 1 : If has child, just make ‘isWord’ false
  - Case 2 : If has no child, delete the node

- void print(string word) Autocomplete
  - Print the whole node from the word
  - Recursion
    - void print_slave(string word, TRIENode* node)
  - Loop

- void printAll(string word)
  - Print the whole words in TRIE
  - With help of function print