NCKU Programming Contest Training Course
2013/07/19

Pin-chieh Huang (free999)

pinchieh.huang@gmail.com

http://myweb.ncku.edu.tw/~p76014143/20130719_Trie.rar

Department of Computer Science and Information Engineering
National Cheng Kung University
Tainan, Taiwan

Problem

Store this collection of words in a memory efficient way :

{abacus, abode, dreaded, dust, dusty, planar, east}

How to proceed?

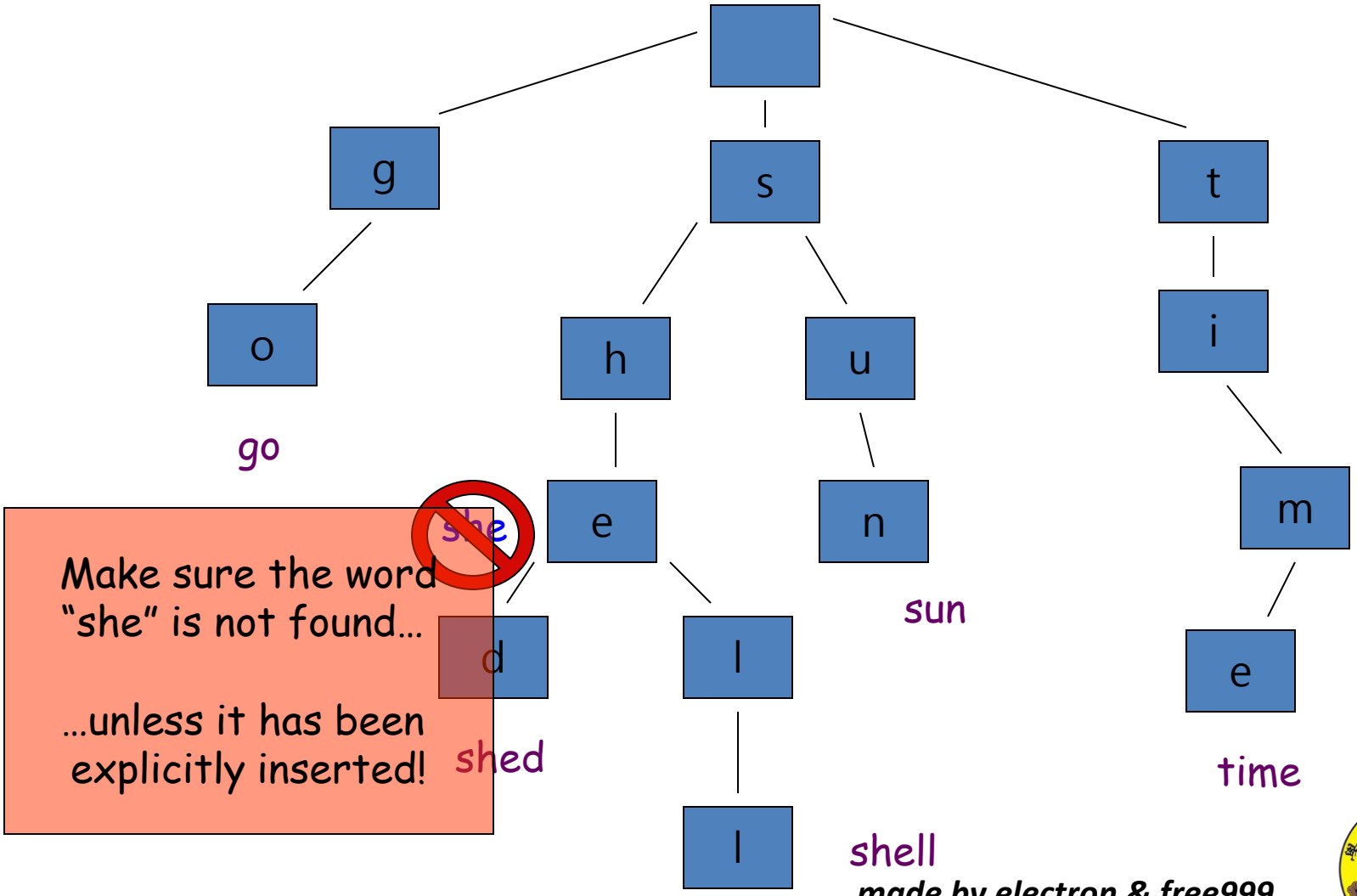
Idea

- Avoid the Duplicate Part
  - Can we find a way to avoid search the same part?

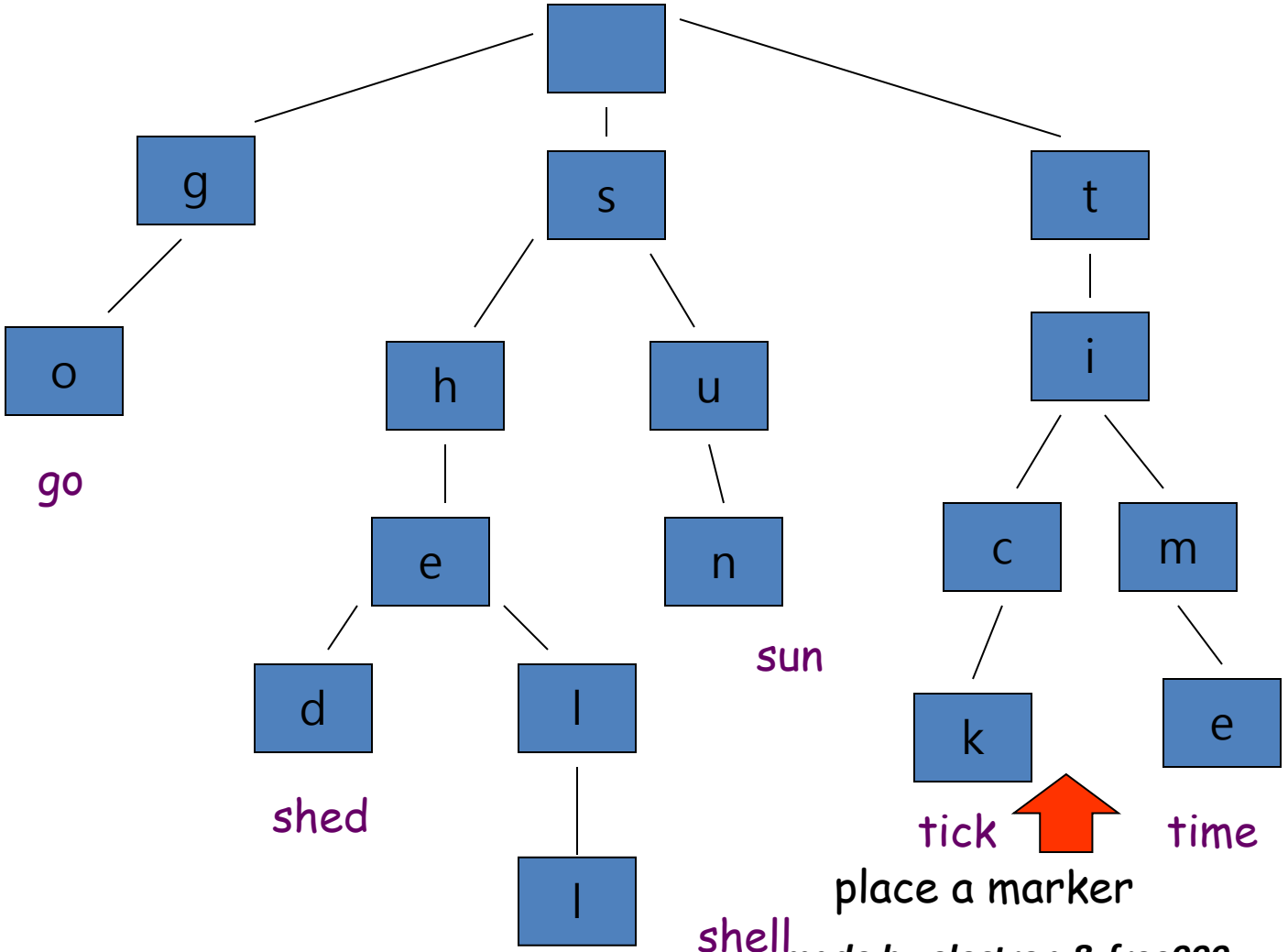
Draw a **trie** corresponding to the collection of words:

{abacus, abode, dreaded, dust, dusty, planar, east}

Example

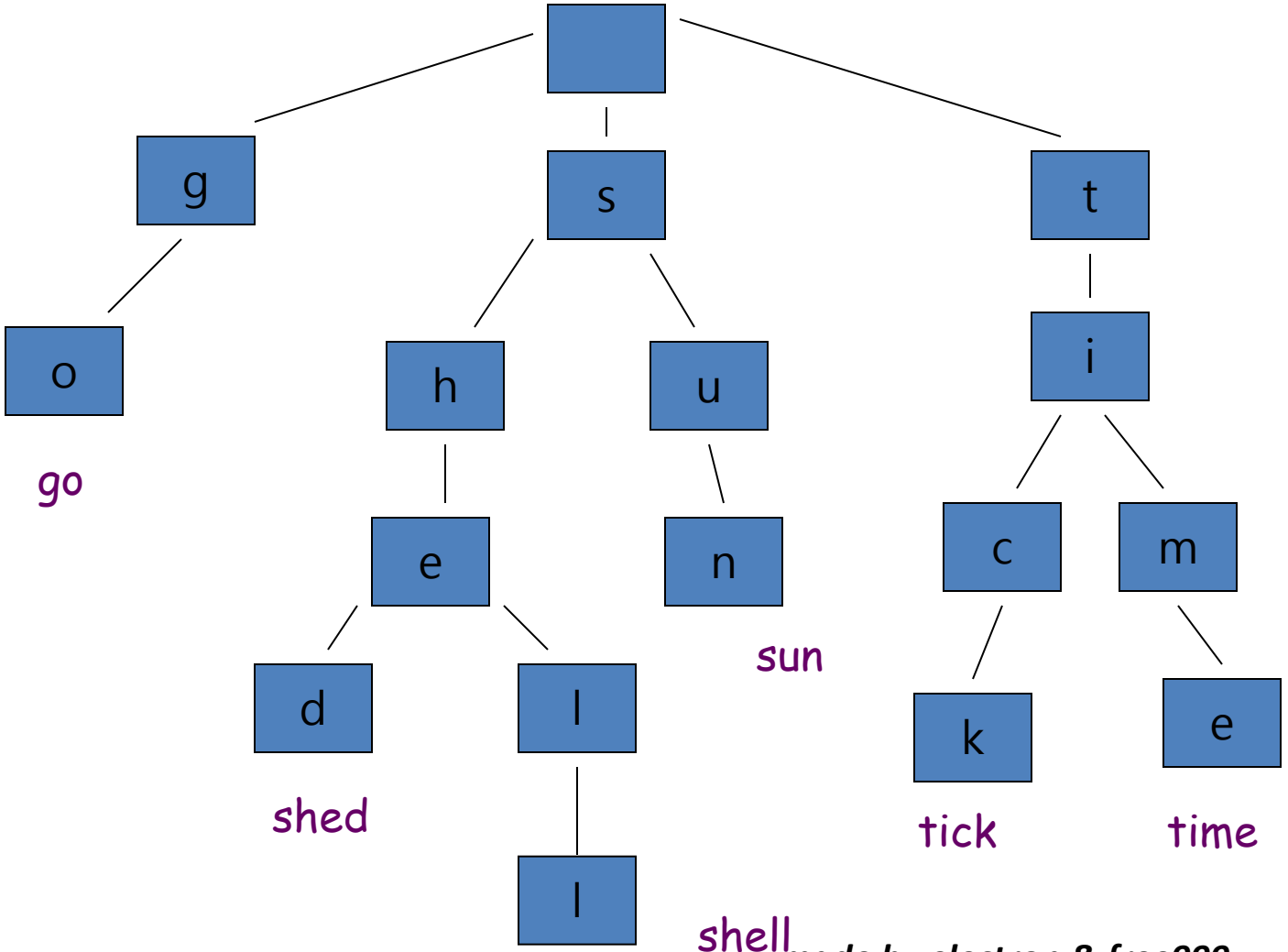

g
s
t
o
h
u
i
go
she
e
n
m
Make sure the word "she" is not found…
…unless it has been explicitly inserted!
d
l
sun
e
shed
time
l
shell

T.insert("tick")

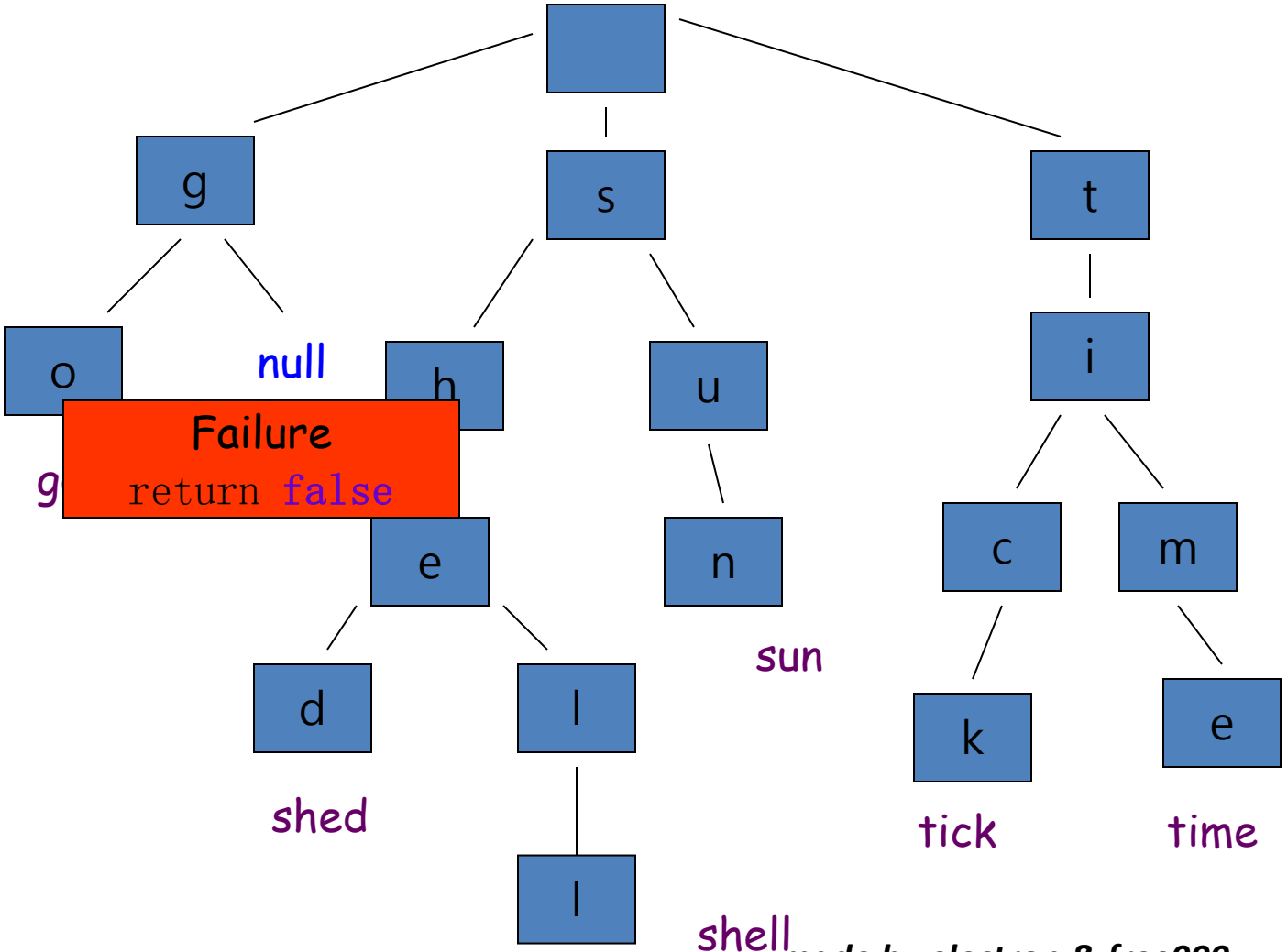


made by electron & free999

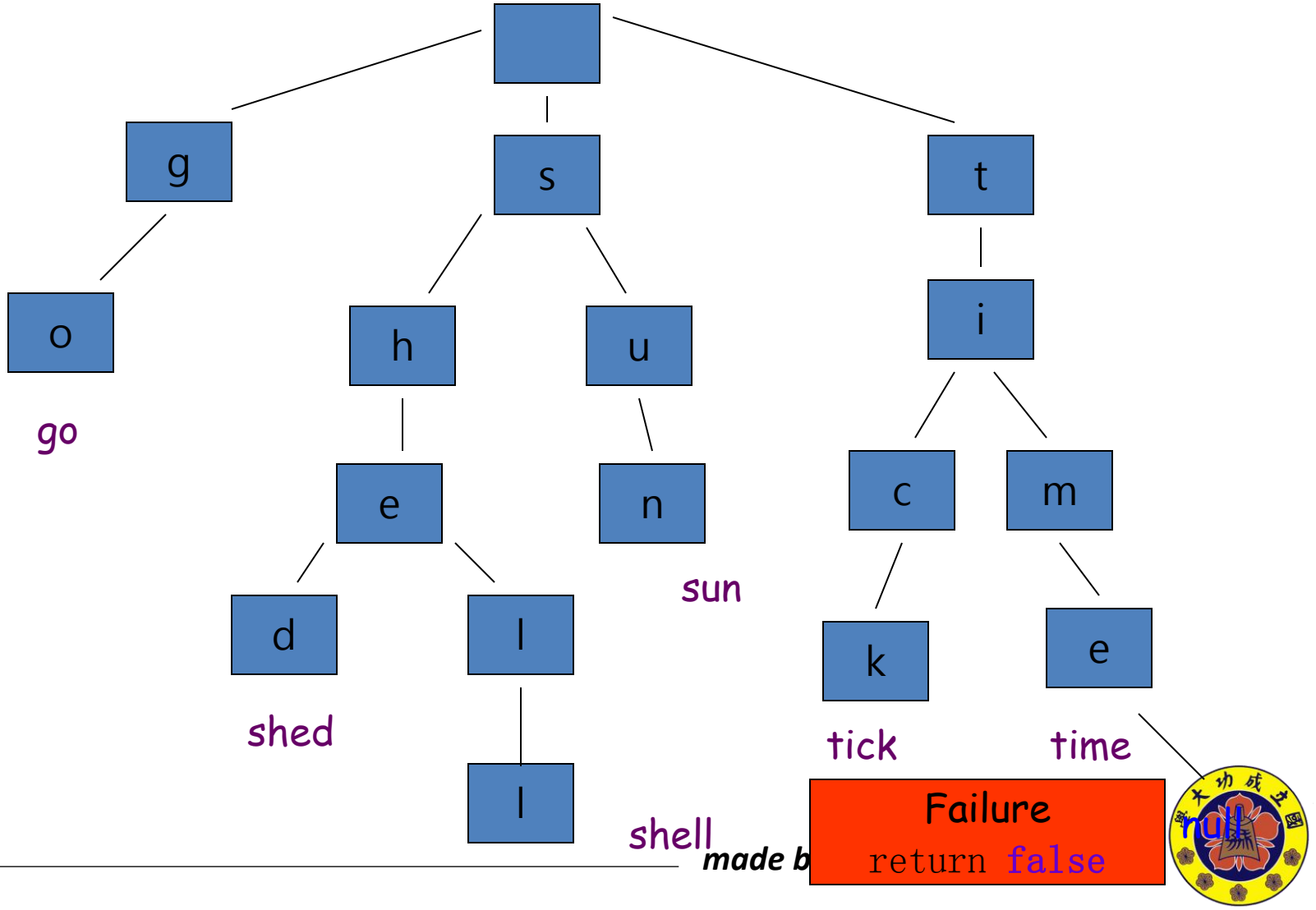
T.insert("tick")



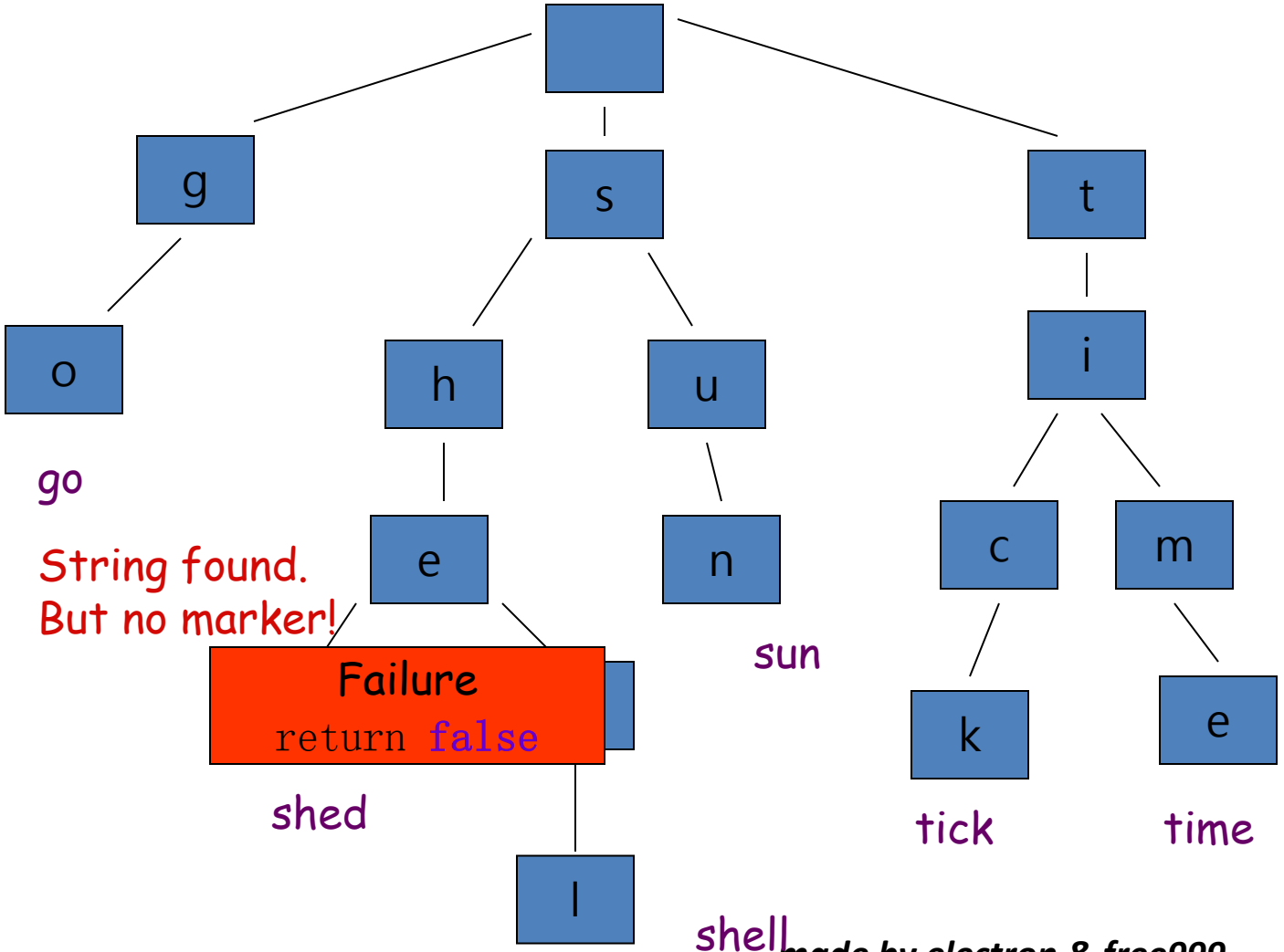
T.search("group")



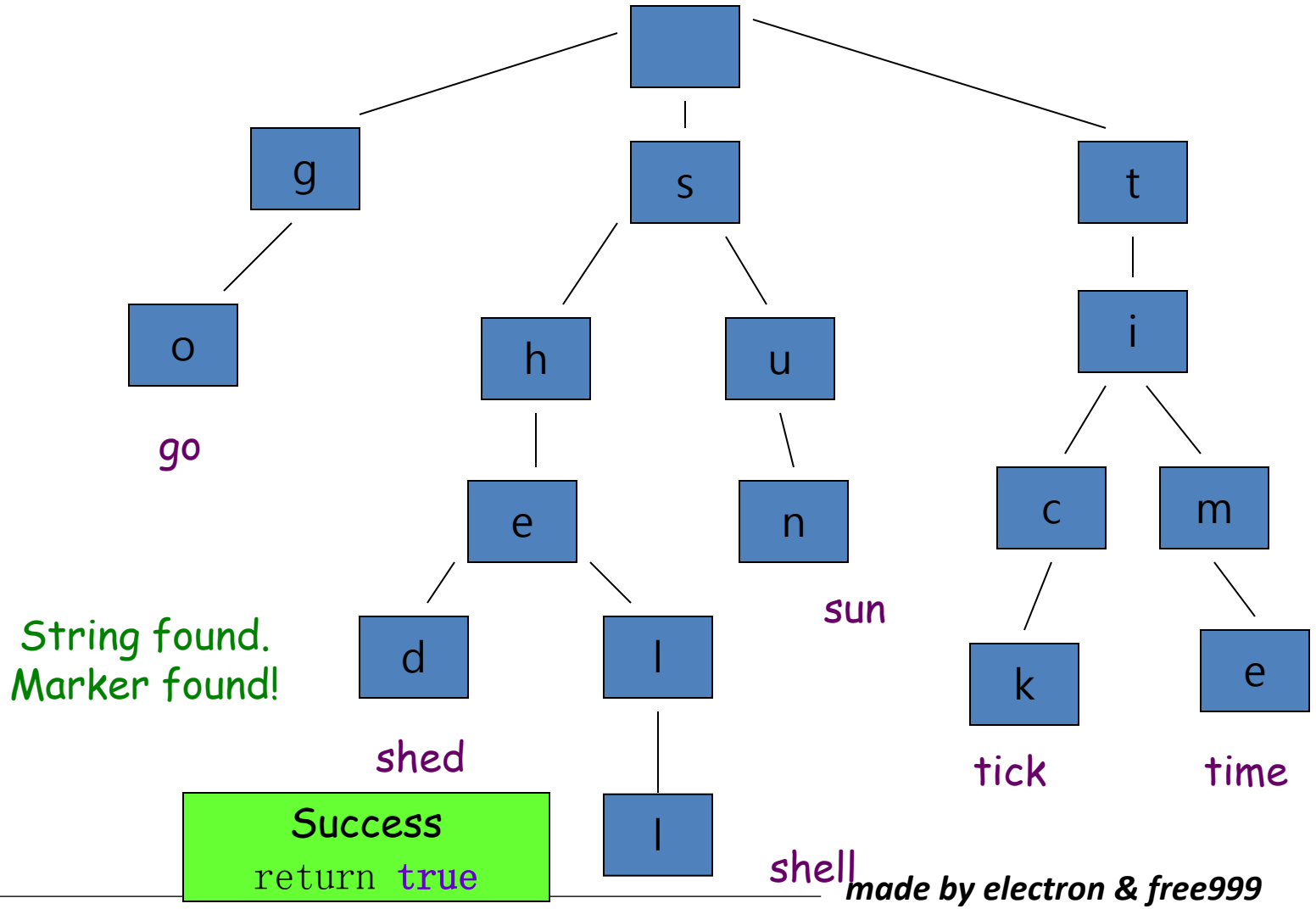
T.search("timer")



T.search("she")


g
s
t
o
h
u
i
go
String found.
But no marker!
e
n
c
m
Failure
return false
sun
k
e
shed
tick
time
l
shell

T.search("shed")



Trie

```
int insert(char word[])
{
    int i, l;
    NODE *ptr;          //ptr is for the traversal.
    l = strlen(word);

    for(i=0, ptr=root; i<l; i++)
    {
        if(ptr->next[word[i]-'a']==NULL)    //new node allocation
            ptr->next[word[i]-'a'] = (NODE*)calloc(1, sizeof(NODE));
        ptr = ptr->next[word[i]-'a'];       //"path" represent the word[i]
    }
    if(!(ptr->isword))  //create new id for the word.
        ptr->id = NW++, ptr->isword = true;
    return ptr->id;     //return the id of the word.
}
```

Example

PkU 3630

Homework

Total 7 Problems

- POJ
  - 2503
  - 2513
  - 3630
- UVa
  - 902
  - 10226
  - 10391
  - 10745

Thank You For Attention!

made by electron & free999