Write a program that reads an integer n, and then n pairs of names, and finally an additional name B. Each name is a sequence of non-whitespace characters delimited by whitespace.

In each pair the first name is the name of an employee and the second name is the name of the boss of the employee. Assume that each employee has at most one boss.

Say that an employee F is "below" an employee E if E is the boss of F, or E is the boss of the boss of F, or the boss of the boss of the boss of F, and so on.

Your program is to print the number of employees below B.

Hint: Use a map to store for each employee E the names of the employees whose boss is E. Thus you need a map<string,vector<string>>. Now observe that the number of people below E = number of people whose boss is E + number of people below each of the persons whose boss is E. So you should be able to do this using recursion.

Sample Test Cases

<table>
<thead>
<tr>
<th>Test Case 1</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>e1</td>
<td>e2</td>
</tr>
<tr>
<td></td>
<td>e1</td>
<td></td>
</tr>
</tbody>
</table>
Test Case 2
7
e6 e7
e1 e2
e2 e3
e3 e4
e4 e5
e5 e6
e7 e8
e8

Test Case 3
3
e2 e3
e1 e2
e3 e4
e3

The due date for submitting this assignment has passed.
As per our records you have not submitted this assignment.
Sample solutions (Provided by instructor)

```cpp
#include <iostream>
define repeat(x) for(int _iterator_i = 0, _iterator_limit = x; _iterator_i < _iterator_limit; _iterator_i++)
define main_program int main()
#include <cmath>
using namespace std;
#define repeat(x) for(int _iterator_i = 0, _iterator_limit = x; _iterator_i < _iterator_limit; _iterator_i++)
define main_program int main()
#include <iostream>
#include <string>
#include <iterator>
#include <map>
#include <vector>
using namespace std;
int below(map<string,vector<string>> &subs, string b){
    if(subs.count(b) == 0) return 0;
    int count = subs[b].size();
    for(auto s : subs[b])
        count += below(subs,s);
    return count;
}
using namespace std;
int main(){
    map<string,vector<string>> subs;
    int n; cin >> n;
    for(int i = 0; i < n; i++) {
        //repeat(n){
            string s,b; cin >> s >> b;
            subs[b].push_back(s);
        }
    string B; cin >> B;
    cout << below(subs,B) << endl;
}
```
Lecture 25:
Medium size programs:
Part 2: Manual algorithm for new marks display (unit?unit=96&lesson=151)

Lecture 25:
Medium size programs:
Part 3: RSMTAB and rest of the program (unit?unit=96&lesson=152)

Lecture 25:
Medium size programs:
Part 4: Sophisticated solutions to marks display (unit?unit=96&lesson=153)

Download Videos (unit?unit=96&lesson=187)

Weekly Feedback (unit?unit=96&lesson=199)

Quiz: Week 11 Quiz (assessment?name=228)

Week 11 Programming Assignment 1 (/noc20_cs53/progassignment?name=229)

Week 11 Programming Assignment 2 (/noc20_cs53/progassignment?name=230)

Week 12

Text Transcripts