

《数据结构》课程实践报告

院、系	计算机学院	年级专业	21 计算机科学与技术	姓名	赵鹏	学号	2127405037
实验布置日期	2022.10.25		提交日期	2022.12.5		成绩	

课程实践实验 9: Family Tree

一、问题描述及要求

A professor of anthropology was interested in people living in isolated islands and their history. He collected their family trees to conduct some anthropological experiment. For the experiment, he needed to process the family trees with a computer. For that purpose he translated them into text files. The following is an example of a text file representing a family tree.

```
John
Robert
  Frank
  Andrew
Nancy
  David
```

Each line contains the given name of a person. The name in the first line is the oldest ancestor in this family tree. The family tree contains only the descendants of the oldest ancestor. Their husbands and wives are not shown in the family tree. The children of a person are indented with one more space than the parent. For example, Robert and Nancy are the children of John, and Frank and Andrew are the children of Robert. David is indented with one more space than Robert, but he is not a child of Robert, but of Nancy. To represent a family tree in this way, the professor excluded some people from the family trees so that no one had both parents in a family tree.

For the experiment, the professor also collected documents of the families and extracted the set of statements about relations of two persons in each family tree. The following are some examples of statements about the family above.

```
John is the parent of Robert.
Robert is a sibling of Nancy.
David is a descendant of Robert.
```

For the experiment, he needs to check whether each statement is true or not. For example, the first two statements above are true and the last statement is false. Since this task is tedious, he would like to check it by a computer program.

Input

The input contains several data sets. Each data set consists of a family tree and a set of statements. The first line of each data set contains two integers n ($0 < n < 1000$) and m ($0 < m < 1000$) which represent the number of names in the family tree and the number of statements, respectively. Each line of the input has less than 70 characters.

As a name, we consider any character string consisting of only alphabetic characters. The names in a family tree have less than 20 characters. The name in the first line of the family tree has no leading spaces. The other names in the family tree are indented with at least one space, i.e., they are descendants of the person in the first line. You can assume that if a name in the family tree is indented with k spaces, the name in the next line is indented with at most $k + 1$ spaces. This guarantees that each person except the oldest ancestor has his or her parent in the family tree. No name appears twice in the same family tree. Each line of the family tree contains no redundant spaces at the end.

Each statement occupies one line and is written in one of the following formats, where X and Y are different names in the family tree.

```
X is a child of Y.
X is the parent of Y.
X is a sibling of Y.
X is a descendant of Y.
X is an ancestor of Y.
```

Names not appearing in the family tree are never used in the statements. Consecutive words in a statement are separated by a single space. Each statement contains no redundant spaces at the beginning and at the end of the line.

The end of the input is indicated by two zeros.

Output

For each statement in a data set, your program should output one line containing True or False.

The first letter of 'True' or 'False' in the output must be a capital. The output for each data set should be followed by an empty line.

Sample Input

```
6 5
John
  Robert
    Frank
    Andrew
  Nancy
  David
Robert is a child of John.
Robert is an ancestor of Andrew.
Robert is a sibling of Nancy.
Nancy is the parent of Frank.
John is a descendant of Andrew.
2 1
abc
  xyz
xyz is a child of abc.
0 0
```

Sample Output

```
True
True
True
False
False

True
```

问题分析

本次实验的内容是一个关于树的实际应用问题。通过名称和缩进给定一棵家谱树。给定对于书上两个人之间的辈分关系的信息，判断信息的正误。这是一个考察树相关知识点的问題，可以将问题拆分成两部分 1. 根据给定信息建立出家谱树 2. 利用建成的树对给定的信息进行判断。

输入：本题目有多组测试数据，对于每组测试数据，第一行给出两个整数 n ($0 < n < 1000$) 和 m ($0 < m < 1000$)，接下来给出 n 行信息，表示同一家族的内的关系。再给出 m 行表示一些判断。当 n 和 m 等于 0 时代表输入结束。

输出：对于每组数据内给定的每个判断，输出判断的正确与否。

二. 问题解决

由题意容易得到我们需要利用给定的信息对于 m 次询问进行判断。但若不借助数据结构的知识直接处理，问题便比较棘手。通过阅读题目得知，同一家族内的子孙关系可以构成一棵树，所以我们可以考虑使用树相关的知识来解决这个问题。

一、建树

由于要使用树形结构，那么第一步便是建树。很容易想到我们可以把家族中的人当作树的结点，利用结点之间的边来表示结点之间的关系。在建出树后便可以利用树上结点的关系对于询问中的子孙关系进行判断。建树部分可以分为以下几个部分：

1. 由于题目中给定的信息是整体从上到下给出，所以我们可以使用双亲表示法来存储这棵树，即记录树中每个结点的双亲结点，初始化时需要将所以结点的双亲置为-1，代表其暂时没有双亲。

2. 由于给定的人名均为字符串，不能够直接使用。将人名与数字之间建立映射，可以使用 C++ 标准模板库中基于红黑树实现的 map 模板来对人名进行编号，即使用 map<string,int>VertexNum 进行实现。

3. 由于输入数据的子孙关系由缩进形式给出，而树上的子孙关系通常与结点在书上的深度相关联，因此需要找到缩进数与深度之间的关系，通过观察可以发现这样一个奇妙的性质：缩进数与深度一一对应！

4. 最后只剩双亲结点的寻找这一步了，对于给定的一个结点，我们可以很容易发现，他的双亲一定是在他之前出现并且深度刚好与他相差 1 的第一个结点，于是我们在给新结点编号后倒序寻找，找到深度刚好与他相差 1 的结点并记录下来即可。

至此建树部分完成。

代码实现如下：

```
01. void Tree::SaveInfo(int i)
02. {
03.     std::string s1;
04.     getline(std::cin, s1);
05.     int temp = s1.find_last_of(' ') - s1.find_first_of(' ') + 1; // 计算空格数
06.     if (i)
07.         s1.erase(0, temp); // 删除空格
08.     vertexNum[s1] = num++; // 给名字编号
09.     depth[num - 1] = temp; // 记录层数=空格数
10.     if (i == 0)
11.         depth[num - 1] = 0;
12.     int pos = num - 2;
13.     while (depth[num - 1] <= depth[pos] && pos >= 0)
14.         pos--;
15.     parent[num - 1] = pos;
16. }
```

二、判断

对于题目中的判断处理有以下几种情况，可以使用分类讨论的方式解决：

由于我们对人名进行了编号，故对于给定的人名 person1 和 person2。我们需要对其进行转化。a=VertexNum[person1], b=VertexNum[person2]。即令 a 和 b 分别为两人在树上的结点编号。

1. a 是 b 的孩子

由于存储树时使用的是双亲表示法，所以之间判断 parent[a]是否等于 b 即可。

2. b 是 a 的孩子

与上面一种情况同理，判断 parent[b]是否等于 a 即可

3. a 是 b 的兄弟

若 a 与 b 为兄弟关系，即 a 和 b 在树上有着相同的双亲，判断 parent[a]是否等于 parent[b]即可。

4. a 是 b 的祖先

转化题意，则 a 可能是 b 的双亲的双亲...的双亲，即沿着 b 的双亲开始不断向上找，

当 b 的某一级双亲等于 a 则说明 b 是 a 的祖先。可以很容易发现这是一个类似递归的过程，但可以将尾递归改为循环来加速这一过程，需要注意的时，整个过程中都需要确保不断寻找双亲过程中结点编号大于等于 0，若找到根节点还没有找到则编号将变为-1，代表查找失败，即 a 不是 b 的祖先。

5. b 是 a 的祖先

解决方法与上一种情况相对称，在此不再赘述。

代码实现如下：

```
01. bool Tree::judge(std::string s1, std::string s2, std::string s3)
02. {
03.     int num1 = vertexNum[s1], num2 = vertexNum[s2];
04.     if (s3 == "child")
05.         return parent[num1] == num2 ? true : false;
06.     else if (s3 == "parent")
07.         return parent[num2] == num1 ? true : false;
08.     else if (s3 == "sibling")
09.         return parent[num1] == parent[num2] ? true : false;
10.     else if (s3 == "descendant")
11.     {
12.         while (num1 >= 0)
13.         {
14.             num1 = parent[num1];
15.             if (num1 == num2) break;
16.         }
17.         if (num1 >= 0) return true;
18.         return false;
19.     }
20.     else
21.     {
22.         while (num2 >= 0)
23.         {
24.             num2 = parent[num2];
25.             if (num1 == num2) break;
26.         }
27.         if (num2 >= 0) return true;
28.         return false;
29.     }
30. }
```

三. 实验结果测试

使用题目中给定的数据进行测试，运行结果为

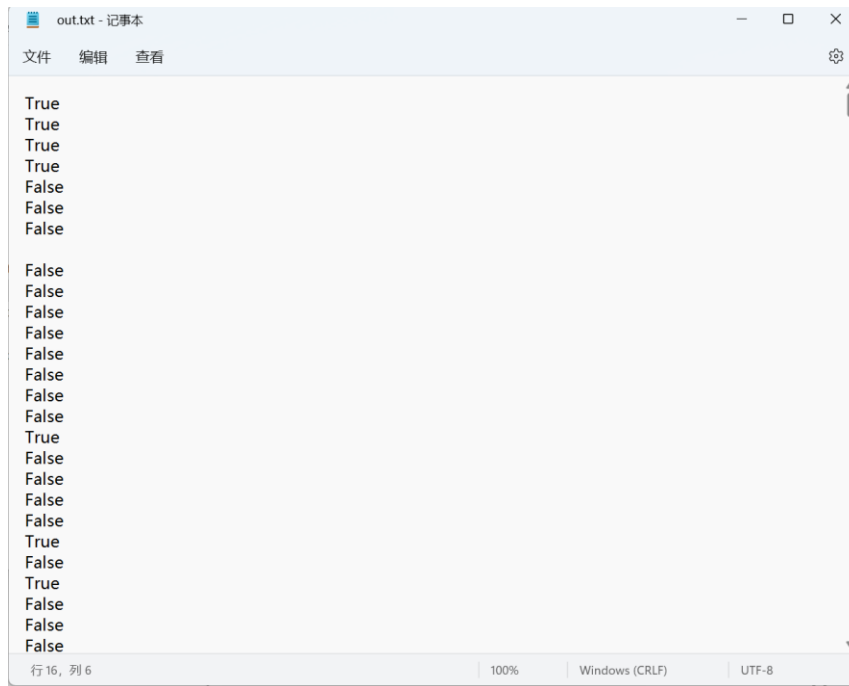
```
6 5
John
  Robert
    Frank
      Andrew
        Nancy
          David
Robert is a child of John.
True
Robert is an ancestor of Andrew.
True
Robert is a sibling of Nancy.
True
Nancy is the parent of Frank.
False
John is a descendant of Andrew.
False

2 1
abc
  xyz
xyz is a child of abc.
True

0 0
```

结论：通过

通过以 family.txt 中的数据为输入，运行结果如下：



```
out.txt - 记事本
文件  编辑  查看
True
True
True
True
False
False
False
False
False
False
False
False
False
True
False
False
False
False
True
False
False
False
False
False
行 16, 列 6 | 100% | Windows (CRLF) | UTF-8
```

详细输出文件 out.txt 已打包在附件中。

四、小结

通过此次实验，我进一步加深了对树相关知识的掌握程度，并使用有关知识解决了一个算法题目。