**算法训练 求先序排列**

时间限制：1.0s   内存限制：256.0MB

**问题描述**  
　　给出一棵二叉树的中序与后序排列。求出它的先序排列。（约定树结点用不同的大写字母表示，长度<=8）。

输入格式

　　两行，每行一个字符串，分别表示中序和后序排列

输出格式

　　一个字符串，表示所求先序排列  
  
　　**样例**输入  
　　BADC  
　　BDCA

样例输出

ABCD

锦囊1

后序的最后一个字母为根结点。

锦囊2

先将后序的最后一个字母取出，作为根结点，以中序遍历将左右子树分开，然后分别处理。

本题的C++参考代码如下：

#include<cstdio>

#include<cstring>

char m[10000];

char h[10000];

void dfs(int ai,int aj,int bi,int bj)

{

int i;

{

if(aj<ai||bj<bi)

return ;

}

if(ai==aj)

printf("%c",m[ai]);

else

{

printf("%c",h[bj]);

for(i=ai;i<=aj;i++)

{

if(m[i]==h[bj])

{

break;

}

}

dfs(ai,i-1,bi,bi+(i-ai-1));

dfs(i+1,aj,bi+i-ai,bj-1);

}

}

int main()

{

gets(m);

gets(h);

dfs(0,strlen(m)-1,0,strlen(m)-1);

puts("");

return 0;

}

本题的C参考代码如下：

#include"stdio.h"

#include"string.h"

void dg(char z[],char h[])

{

if(strlen(h)==0)

return;

printf("%c",h[strlen(h)-1]);

if(strlen(h)==1)

return;

if(strlen(h)==2)

{

printf("%c",h[0]);

return;

}

char a[9],b[9];

int i,j;

for(i=0;z[i]!=h[strlen(h)-1];i++)

a[i]=z[i],b[i]=h[i];

a[i]='\0';

b[i++]='\0';

dg(a,b);

for(j=0;i<strlen(h);i++,j++)

a[j]=z[i],b[j]=h[i-1];

a[j]='\0';

b[j]='\0';

dg(a,b);

}

int main()

{

char h[9],z[9];

scanf("%s",z);

scanf("%s",h);

dg(z,h);

return 0;

}

本题的Java参考代码如下：

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

while (scanner.hasNext()) {

String str1 = scanner.nextLine();

String str2 = scanner.nextLine();

showResult(str1, str2);

System.out.println();

}

}

private static void showResult(String str1, String str2) {

char chl = str2.charAt(str2.length() - 1);

System.out.print(chl);

int index = str1.indexOf(chl);

if (index > 0) {

showResult(str1.substring(0, index), str2.substring(0, index));

}

if (index < str1.length() - 1) {

showResult(str1.substring(index + 1),

str2.substring(index, str2.length() - 1));

}

}

}