**新建Microsoft Word文档**

问题描述

　　L正在出题，新建了一个word文档，想不好取什么名字，身旁一人惊问：“你出的题目叫《新建Microsoft Word文档》吗？”，L大喜，一拍桌子，说：“好，就叫这个名字了。”  
　　仔细观察，当你新建一个word文档时，会得到一个名为“新建 Microsoft Word 文档.doc”的文件，再新建一个，则名为“新建 Microsoft Word 文档(2).doc”，再新建，便是“新建 Microsoft Word 文档(3).doc”。不断新建，编号不断递增。倘若你现在新建了三个文档，然后删除了“新建 Microsoft Word 文档(2).doc”，再新建就又会得到一个“新建 Microsoft Word 文档(2).doc”。  
  
　　严格说，Windows在每次新建文档时，都会选取一个与已有文件编号不重复的最小正整数作为新文档的编号。  
  
　　请编程模拟以上过程，支持以下两种操作  
　　New：新建一个word文档，反馈新建的文档的编号  
　　Delete id：删除一个编号为id的word文档，反馈删除是否成功  
　　初始时一个文件都没有，“新建 Microsoft Word 文档.doc”的编号算作1。

输入格式

　　第一行一个正整数n表示操作次数，接下来n行，每行表示一个操作。若该行为”New”，则表示新建，为”Delete id”则表示要删除编号为id的文档，其中id为一个正整数。操作按输入顺序依次进行。

输出格式

　　对于输入的每一行，输出其反馈结果。对于新建操作，输出新建的文档的编号；对于删除操作，反馈删除是否成功：如果删除的文件存在，则删除成功，输出”Successful”，否则输出”Failed”。

样例输入

12  
New  
New  
New  
Delete 2  
New  
Delete 4  
Delete 3  
Delete 1  
New  
New  
New  
Delete 4

样例输出

1  
2  
3  
Successful  
2  
Failed  
Successful  
Successful  
1  
3  
4  
Successful

数据规模和约定

　　操作次数（即输入的行数）不超过1481  
　　删除编号的数值不超过2012

C++代码：

#include <iostream>

#include <string>

#include <cmath>

using namespace std;

#define SIZE 1481

int main(){

int arr[SIZE] = { 0 };

int table[10] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };

int n;

cin >> n;

cin.clear();

cin.get();

int length;

int temp;

string str;

string cmd[SIZE];

for (int i = 0; i < n; i++){

getline(cin, cmd[i]);

}

for (int i = 0; i < n; i++){

str = cmd[i];

if (str == "New"){

for (int j = 0; j < SIZE; j++){

if (arr[j] == 0){

arr[j] = j + 1;

cout << j + 1 << endl;

break;

}

}

}

else{

length = str.length();

temp = 0;

for (int i = 7; i < length; i++){

temp = temp + table[str[i] - 48] \* pow(10, length - 1 - i);

}

if (arr[temp - 1] == 0){

cout << "Failed" << endl;

continue;

}

else{

arr[temp - 1] = 0;

cout << "Successful" << endl;

continue;

}

}

}

return 0;

}

C代码：

#include<stdio.h>

#include<stdlib.h>

void main()

{

int a[1482]={0};

int b[1482]={0};

char str[10];

char \*p;

char (\*in)[10];

char (\*ip)[10];

int i=0;

int n;

int n1;

scanf("%d",&n);

n1=n;

in=(char \*)malloc(sizeof(char)\*10\*n);

ip=in;

scanf("%s",in);

while(--n)//

{

if(in[0][0]=='N')

{

in++;

scanf("%s",in);

if(in[0][0]!='N'&&1==n)

scanf("%d",&b[i++]);

}

else

{

in++;

scanf("%d",&b[i++]);

if(n==1)

{

scanf("%s",in);

if(in[0][0]!='N')

scanf("%d",&b[i++]);

}else if(n!=1)scanf("%s",in);

}

}

in=ip;

n=n1;

i=0;

/\*

while(i<n)

{

printf("%d\n",b[i++]);

}

\*/

while(n--)

{

if(in[0][0]=='N')

{

int t=1;

while(a[t]!=0)

{

t++;

}

a[t]=1;

printf("%d\n",t);

in++;

}

else

{

if(b[i]>1482||a[b[i]]==0)

{

printf("Failed\n");

in++;

i++;

}

else

{

a[b[i]]=0;

i++;

printf("Successful\n");

in++;

}

}

}

}

JAVA代码：

import java.io.\*;

public class Main {

public static void main(String[] args) throws NumberFormatException, IOException {

BufferedReader buf=new BufferedReader(new InputStreamReader(System.in));

int n=Integer.parseInt(buf.readLine());

int[] arr=new int[n+1];

StringBuilder strb=new StringBuilder();

for(int i=0;i<n;i++){

String str=buf.readLine();

if(str.equals("New")){

for(int j=1;j<=n;j++){

if(arr[j]==0){

arr[j]=1;

strb.append(j).append("\r\n");

break;

}

}

}else{

str=str.substring(str.indexOf(" ")+1);

int t=Integer.parseInt(str);

if(arr[t]==1){

arr[t]=0;

strb.append("Successful");

}else{

strb.append("Failed");

}

strb.append("\r\n");

}

}

System.out.println(strb);

}

}