车厢调度

电信1808 连琛 U201813455 4月8日

问题描述：

假设停在铁路调度站入口处的车厢序列的编号依次为1,2,3….n。设计一个程序，求出所有可能由此输出的长度为n的车厢序列。

1. 需求分析

程序要求出所有可能的车厢的序列，其中0<n<5。

测试数据

4

4 3 2 1

3 4 2 1

3 2 4 1

3 2 1 4

2 4 3 1

2 3 4 1

2 3 1 4

2 1 4 3

2 1 3 4

1 4 3 2

1 3 4 2

1 3 2 4

1 2 4 3

1 2 3 4

1. 概要设计

主程序流程为读入n的值，定义两个数组栈并初始化，再应用递推函数求解即可。

读入n的值

定义两个数组栈并初始化

应用递推函数求解

1. 详细设计

main()

{ int a[],b[],n;

scanf(n);

a[0]=b[0]=0;

gan(a,b,1,n);

}

equ(a,b) {b=a;}

push(a,e) {a[0]++; a[a[0]]=e;}

gan(a,b,I,n) // a表示待定要进入的b的车 b表示已经出来的车 i表示下一个将要出来// 的车的编号 n为总数

{int aa[100],bb[100],j; //用 aa和bb 储存a和b当前的状态

if (i>n) //说明所有车全部进入，只需将ab中的数据按顺序输出

{for (j=1;j<=b[0];j++) printf(b[j]);

for (j=a[0];j>0;j--) printf(a[j]);

}

else

{//这时i<n 需要分类讨论i进入a和a出一辆车进入b的情况

equ(a,aa); equ(b,bb); // 之后运算会改变ab的值，因此用aabb保存ab的当前状态

push(a,i); // i进入a

gan(a,b,i+1,n);

equ(aa,a); equ(bb,b); // 还原ab的状态

if (a[0]!=0) // a非空则计算a进入b的情况

{

b[0]++; b[b[0]]=a[a[0]]; a[0]--;

gan(a,b,i,n);

}

equ

push

main

gan

1. 调试分析

时间复杂度为o(2^n) 空间复杂度为 o(n\*2^n);

1. 用户使用说明

打开codeblocks 直接使用，输入n的值即可

1. 测试结果

4

4 3 2 1

3 4 2 1

3 2 4 1

3 2 1 4

2 4 3 1

2 3 4 1

2 3 1 4

2 1 4 3

2 1 3 4

1 4 3 2

1 3 4 2

1 3 2 4

1 2 4 3

1 2 3 4

3

3 2 1

2 3 1

2 1 3

1 3 2

1 2 3