# 车厢调度实验报告

电信1809蔡俊哲 学号U201813485 2019/4/14

1. 需求分析
2. 建立栈，让车厢从1到n入栈或输出，求得所有可能的排列
3. 当一个数进站时要么下一个数进站，要么栈顶出站
4. 当一个数出站时要么接着出站要么下一个数入栈
5. 概要设计

用数组建立栈

struct zhan

{

int a[N];

int top;

};

主要程序：1、入栈函数

push(struct zhan \*x,int y)

{

if(x->top==N-1)

{

printf("the stack is full");

return;

}

x->a[++x->top]=y;

}

1. 出栈函数

pop(struct zhan \*x)

{

if(x->top==-1)

{

printf("the stack is empty");

return;

}

return(x->a[x->top--]);

}

1. 判断空栈

isempty(struct zhan x)

{

if(x.top==-1)

{

return 0;

}

return 1;

}

4、分情况讨论，如果出列已满车箱数则输出，若车厢全进入则只出站，站内不为空则有两个分支，入站和出站，若为空则只有入站操作

1. 详细设计

#include <stdio.h>

#define N 100

struct zhan

{

int a[N];

int top;

};

void push(struct zhan \*x,int y)

{

if(x->top==N-1)

{

printf("the stack is full");

return;

}

x->a[++x->top]=y;

}

int pop(struct zhan \*x)

{

if(x->top==-1)

{

printf("the stack is empty");

return;

}

return(x->a[x->top--]);

}

int isempty(struct zhan x)

{

if(x.top==-1)

{

return 0;

}

return 1;

}

void paixu(struct zhan x,struct zhan y,int n,int j)

{

struct zhan b=x,c=x,d=y,e=y;

int i,k=j;

if(y.top==n-1)

{

for(i=0;i<=n-1;i++)

{

printf("%d",y.a[i]);

}

printf("\n");

}

else if(k>n)

{

push(&d,pop(&b));

paixu(b,d,n,k);

}

else if(isempty(x))

{

push(&d,pop(&b));

push(&c,k);

paixu(b,d,n,k);

k++;

paixu(c,e,n,k);

}

else if(!isempty(x))

{

push(&c,k);

k++;

paixu(c,e,n,k);

}

}

void main()

{

struct zhan x,y;

int i,n,k=1;

y.top=-1;

x.top=-1;

printf("please input n\n");

scanf("%d",&n);

paixu(x,y,n,k);

}

## 4、调试分析

问题：建栈时对栈的基础函数掌握不够清楚，对于调度问题考虑不够全面

时间复杂度：O（n）

## 测试结果

当n=3时，调度为123、231、321、213、132

当n=4时，调度为 当n=5时，调度为

1234

1243

1324

1342

1432

2134

2143

2314

2341

2431

3214

3241

3421

4321