# 线性表编程作业

姓名：寇一笑 18020024016 安皓源 18020022001

2020.3.23

## Contents

1. **实验目的和内容............................................................................2**
   1. 题目描述..........................................................................................2
2. **输入和输出说明............................................................................2**
   1. 输入..................................................................................................2
   2. 输出..................................................................................................2

2.3问题分析...........................................................................................2

1. **解题思路.......................................................................................3**
2. **实验代码及注释............................................................................3**
3. **运行结果图....................................................................................4**
4. **总结体会........................................................................................4**

**1.实验目的和内容**

**1.1题目描述**

祖玛是一款曾经风靡全球的游戏，其玩法是：在一条轨道上初始排列着若干个彩色珠子，其中任意三个相邻的珠子不会完全同色。此后，你可以发射珠子到轨道上并加入原有序列中。一旦有三个或更多同色的珠子变成相邻，它们就会立即消失。这类消除现象可能会连锁式发生，其间你将暂时不能发射珠子。

开发商最近准备为玩家写一个游戏过程的回放工具。他们已经在游戏内完成了过程记录的功能，而回放功能的实现则委托你来完成。游戏过程的记录中，首先是轨道上初始的珠子序列，然后是玩家接下来所做的一系列操作。你的任务是，在各次操作之后及时计算出新的珠子序列。

1. **输入和输出说明**

**2.1输入**

第一行是一个由大写字母'A'~'Z'组成的字符串，表示轨道上初始的珠子序列，不同的字母表示不同的颜色。

第二行是一个数字n，表示整个回放过程共有n次操作。

接下来的n行依次对应于各次操作。每次操作由一个数字k和一个大写字母Σ描述，以空格分隔。其中，Σ为新珠子的颜色。若插入前共有m颗珠子，则k ∈ [0, m]表示新珠子嵌入之后（尚未发生消除之前）在轨道上的位序。

ACCBA

5

1 B

0 A

2 B

4 C

0 A

**2.2输出**

输出共n行，依次给出各次操作（及可能随即发生的消除现象）之后轨道上的珠子序列。

如果轨道上已没有珠子，则以“-”表示。

ABCCBA

AABCCBA

AABBCCBA

-

A

**2.3问题分析**

根据题目，我们要解决的问题为：

1. 采用何种线性表
2. 如何执行珠子的插入操作
3. 如何检测并删除三个及三个以上相连珠子

# 3.解题思路

为操作的方便起见，我们直接通过数组的方式解决本题。首先先建立两个数组，一个用于存储珠子序列，一个用于暂存或中转。本题最核心的地方是珠子的插入，序列的检索与珠子的删除。我们灵活运用strcpy操作进行珠子的插入，并通过数组的覆盖来完成删除功能。每次插入或删除后会检索一次序列，若无法继续进行操作则进入下一循环。

1. **实验代码及注释**

#include<stdio.h>

#include<stdlib.h>

#include <string.h>

char a[20000],t[20000],c;//最多2万个珠子

int len=0,n,p;

int Zuma(int &Pointer)

{

int p1=Pointer,p2=Pointer;//定义指针

char key=a[Pointer];

while(p1>0&&a[p1]==key)p1--;//记录前面相同珠子数量

if(p1||a[p1]!=key)p1++;//直至前面不同珠子

while(p2<len&&a[p2]==key)p2++;//记录后面相同珠子数量

if(p2-p1>=3)//有3个及以上珠子相同

{

strcpy(t,a+p2);

strcpy(a+p1,t);//将从p2开始的字符串覆盖从p1开始的字符串

len-=(p2-p1);//字符串长度减去相同珠子数

Pointer=p1;//覆盖完后，下一次待比较位置在p1处

return 1;

}

return 0;

}

int main()

{

gets(a);

len=strlen(a);

scanf("%d",&n);

while(n--)

{

scanf("%d %c",&p,&c);

strcpy(t,a+p);

a[p]=c;

len++;

strcpy(a+p+1,t);//两个strcpy用于珠子的插入

while(len&&Zuma(p));//检测消除，每消除完一次再检测一次，直至Zuma返回0或长度为0

//Zuma(p);

if(len)

puts(a);

else

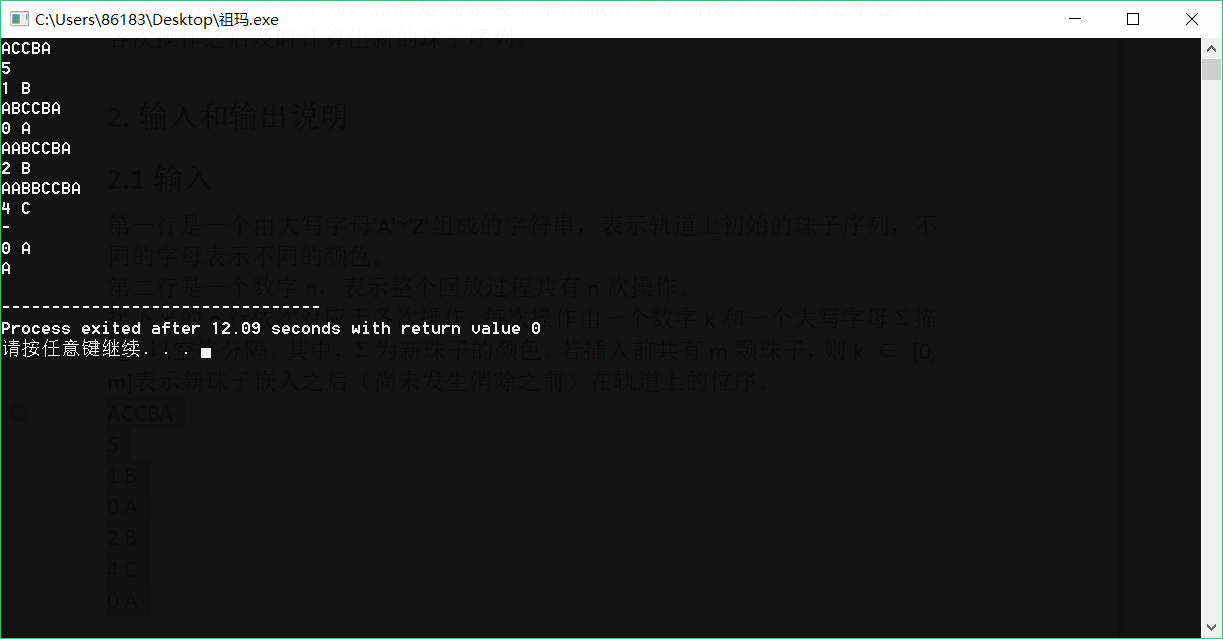
puts("-");

}

return 0;

}

1. **运行结果截图**



1. **总结体会**

本题因数组操作方便利用了数组来解决问题。对于链表我操作还不够熟练，思路比较清晰但具体的语句和相关操作不够熟练因此没有用链表来处理本问题。通过本题，我能够更加熟练通过strcpy来处理数组并完成相关操作。