**实验报告**

|  |  |
| --- | --- |
| **课程名称** | **程序设计基础（C语言）** |
| **实验项目** | **实验三数组程序设计** |
|  |  |

|  |  |  |
| --- | --- | --- |
| **学院** |  | **计算机学院** |
| **系别** |  | **计算机大类** |
| **班级/学号** |  | **计类1607/2016011353** |
| **学生姓名** |  | **郭泽晨** |
| **实验日期** |  |  |
| **成绩** |  |  |
| **指导教师** |  | **蒋玉茹** |

## 实验三 数组（一）

1. **实验目的**
2. **掌握一维数组的定义及初始化。**
3. **掌握一维数组元素的引用及其应用。**
4. **实验内容**

**注：本次实验完成前三个题目或者仅完成第四个题目最高分为90分，完成全部四个题目最高分100分。**

**题目1：用选择法和冒泡法对数组进行从小到大的排序**

提示（步骤）：

1. 定义一个一维数组
2. 从键盘输入数组的值
3. 输出数组的初始内容
4. 排序
5. 输出排序后的结果

1.流程图

2.程序

#include<stdio.h>

int main(){

int i,j=0,t=0,x,y,s=0,k;

int a[10];

for(i=0;i<10;i++)

{

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

}

for(y=0;y<10;y++)

{

while(j<10)

{

if(a[j]>=t)

{

t=a[j];

x=j;

}

j=j+1;

}

s=a[y];

a[y]=a[x];

a[x]=s;

j=y+1;

t=0;

}

printf("\n");

for(k=0;k<10;k++)

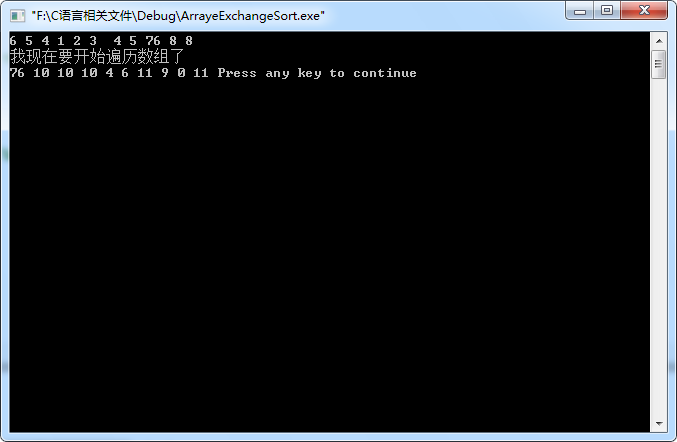
{

printf("%d ",a[k]);

}

return 0;

}3.结果



**题目2：将一个数组中的值按照原始顺序的逆序重新存放，并输出逆序后的数组。**

要求：

【输入】第1行为一个正整数n，表示原始的整数序列长度为n（n不超过50）。

【输入】第2行为n个整数，相邻两数字用一个空格分隔，表示原始的整数序列。

【输出】第3行输出逆序的整数序列。每个整数后输出一个空格。请注意行尾输出换行。

可用多种方法：

方法1：利用下标控制法输出逆序的数组

方法2：将原始序列逆序存放到另一个数组，再输出

方法3：将原始数组中数组元素逆序存放后输出

**请实现以上三种方法中的两种。**

１．流程图

2.程序

#include<stdio.h>

#define len 5

int main(){

int i,a[len],t;

for(i=0;i<len;i++)

{

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

}

for(i=0;i<len/2;i++)

{

t=a[i];

a[i]=a[len-i-1];

a[len-1-i]=t;

}

for(i=0;i<len;i++)

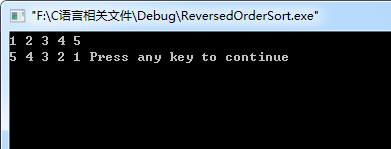
{

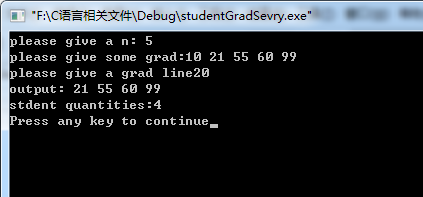
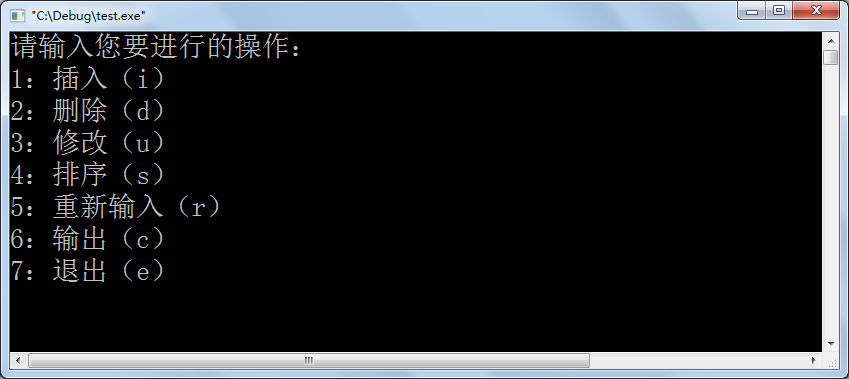
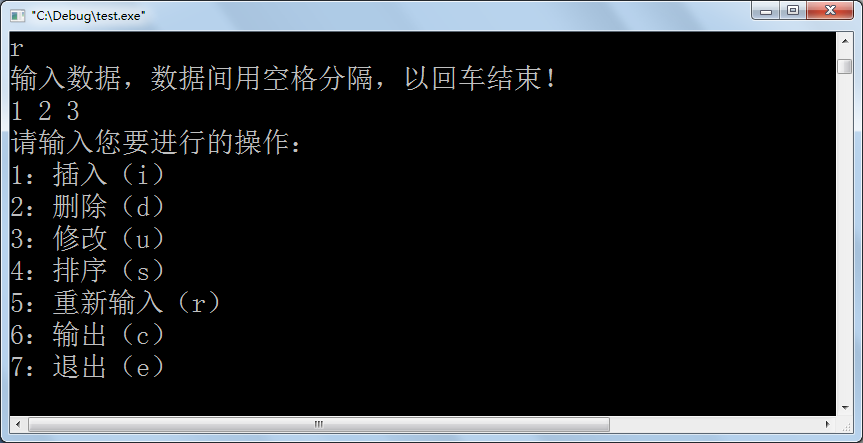
printf("%d ",a[i]);

}

return 0;

}3.结果



1. **题目3：读入n名学生的成绩，将某一给定分数的学生人数输出。**
2. 【输入】第1行为一个正整数n（不超过100）
3. 【输入】第2行为n名学生的成绩，相邻两数字用一个空格分隔。
4. 【输入】第3行为一个分数。
5. 【输出】第4行为具有给定分数的学生人数，如果没有该分数则输出0。
   1. 流程图
   2. 程序
   3. #include<stdio.h>
   4. int main(){
   5. int a[100];
   6. int n,gl,i,t=0;
   7. printf("please give a n: ");
   8. scanf("%d",&n);
   9. printf("please give some grad:");
   10. for(i=0;i<n;i++)
   11. {
   12. scanf("%d",&a[i]);
   13. }
   15. printf("please give a grad line");
   16. scanf("%d",&gl);
   17. printf("output: ");
   18. for(i=0;i<n;i++)
   19. {
   20. if(a[i]>gl)
   21. {
   22. t=t+1;
   23. printf("%d ",a[i]);
   24. }
   26. }
   27. printf("\n");
   28. printf("stdent quantities:%d",t);
   29. printf("\n");
   30. return 0;
   31. }结果
   32. 
6. **题目4：对数组进行插入、排序、删除、替换操作。**
7. **程序启动时的提示信息：**
8. 
9. 用户输入括号中的字母控制程序的运行。
10. 比如用户输入r，则提示用户输入数据。
11. 用户输入1 2 3之后，程序再次回到初始提示菜单。
12. 上述过程操作结果界面如下：
13. 
14. 其他请自行设计。

2.程序

#include<stdio.h>

#define len 100

#define shu 5

int a[len];

int k,x,c;

char ch;

void Cut(int b,int d)

{

int t;

printf("输入您要插入的数：");

scanf("%d",&b);

printf("输入您要插入的位置（第一位从0计算）：");

scanf("%d",&d);

t=shu;

for(k=shu-d;k>0;k--)

{

a[t]=a[t-1];

t--;

}

a[d]=b;

for(k=0;k<shu+1;k++)

{

printf("%d ",a[k]);

}

printf("\n");

}

void Del()

{

int yaoshanchudeshu;

printf("请输入你想删除的位置（第一位从0计算）\n");

scanf("%d",&yaoshanchudeshu);

for(k=0;k<shu-1;k++)

{

a[yaoshanchudeshu]=a[yaoshanchudeshu+1];

yaoshanchudeshu=yaoshanchudeshu+1;

}

for(k=0;k<shu-1;k++)

{

printf("%d ",a[k]);

}

}

void Edit()

{

int yaoxiugaideshu;

int xiugaichengdeshu;

printf("你要修改哪个数？(第一位从0计算)");

scanf("%d",&yaoxiugaideshu);

printf("你要修改成什么数？");

scanf("%d",&xiugaichengdeshu);

a[yaoxiugaideshu]=xiugaichengdeshu;

for(k=0;k<shu;k++)

{

printf("%d ",a[k]);

}

}

void Sort()

{

int i,j=0,t=0,x,y,s=0,k;

for(y=0;y<shu;y++)

{

while(j<shu)

{

if(a[j]>=t)

{

t=a[j];

x=j;

}

j=j+1;

}

s=a[y];

a[y]=a[x];

a[x]=s;

j=y+1;

t=0;

}

for(k=0;k<shu;k++)

{

printf("%d ",a[k]);

}

}

void Rewirte()

{

main();

}

void Cout()

{

for(k=0;k<shu;k++)

{

printf("%d ",a[k]);

}

}

int main()

{

printf("请输入%d数据，用空格分隔开，以回车结束：\n",shu);

for(k=0;k<shu;k++)

{

scanf("%d",&a[k]);

}

printf("您要如何操作该数组？\n");

printf("请输入您需要的操作：\n");

printf("1.插入（i）\n");

printf("2.删除（d）\n");

printf("3.修改（u）\n");

printf("4.排序（s）\n");

printf("5.重新输入（r）\n");

printf("6.输出（c）\n");

printf("7.退出（e） \n");

getchar();

scanf("%c",&ch);

switch(ch)

{

case'i':

printf("输入您要插入的数：");

scanf("%d",&x);

printf("输入您要插入的位置（第一位从0计算）：");

scanf("%d",&c);

Cut(x,c);

break;

case'd':

Del();

break;

case'u':

Edit();

break;

case's':

Sort();

break;

case'r':

Rewirte();

break;

case'c':

Cout();

break;

case'e':

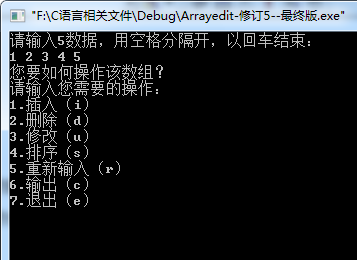
printf("谢谢您的使用！\n");

}

return 0;

}

实验结果：



1. **实验心得**

1.得好好学习才行！