1. 字体字号不限，编辑部会重新排版
2. 字数：内文3500字以上，除去标题、单位、摘要、关键词、文献不低于3000字
3. 建议用Word的样式：标题、标题1、标题2、正文
4. 示意图/流程图：请用Visio绘制，并提供 .vsd 文件
5. 公式：在Word2010版以上自带公式编辑中重新输入
6. 投稿邮箱： tougao@moderncomputer.cn ， 邮件主题注明：投稿，投稿后1周内发通知
7. 发表费、发票、排查、修稿事项联系编辑小月 QQ 2676300211，或电话：020-34476261

不改变数据位置的排序算法及动态演示

宁宁1，张霞2

（1. 潍坊教育学院信息工程系，青州 262500；

2. 潍坊教育学院数学系，青州 262500)

**摘要**：实际应用中经常遇到要求不改变原始数据的顺序而按关键字的大小对数据进行排序的情况，原有的一些经典排序算法不能直接用于解决该类问题。经过对选择排序算法进行研究，给出基于选择思想的不改变数据位置而对数据进行排序的算法，并利用C#语言编程对该算法的实现过程进行动态演示。

**关键词**：排序；关键字；选择；定时器

**基金项目：**潍坊教育学院青年计划（No.201102js）

Sorting Algorithm Without Changing the Data Position & Dynamic Demonstration

**FU Ning1, ZHANG Dong-xia2**

(1. Department of Information Engineering, Weifang College of Education, Qingzhou 262500;

2. Department of Information Engineering, Weifang College of Education, Qingzhou 262500)

**Abstract:** In practical application, the situation, in which it requires a listing of the data in the order of the size of the keywords without changing the order of the original data, is an often-met case. The original classic sorting algorithm cannot be used directed to solve this kind of problem. This paper, by researching into the selective sorting algorithm, puts forward an algorithm on the basis of sorting the data without changing the positions of the data. It also gives a dynamic demonstration of the realization procedure of this algorithm by applying the C language programming.

**Keywords:** Sorting; Key Word; Select; Timer

# 0 引言

排序是计算机程序设计中一项基本的操作，在实际应用中，有很多情况下需要对数据按照某种方式进行排序后才能达到某种要求，因此，学习和研究各种排序方法是计算机工作者的重要课题之一。

我们已经熟知的、比较成熟的排序算法有很多，比如冒泡排序、选择排序、插入排序、快速排序等，利用这些排序算法都能够使一组数据序列按照某个关键字排成需要的顺序[1]。但这些经典的排序算法在对数据序列排序时，都要改变数据的原始顺序，也就是说，在一般情况下，排序问题的输入是n个数a1，a2，a3，……，an的一个序列，按照某个关键字对初始序列进行重新排序后产生初始输入序列的一个重新排列：a11， a21， a31， ……，an1，使得a11< a21< a31<……< an1。经过排序后的结果序列一般和初始的输入序列是不一样的。而在实际应用中，我们可能只需要对数据按照某个关键字进行排序，记下该关键字对应的记录在数据序列中的位次或顺序，而并不要求原始数据的顺序发生改变。例如，通常我们对学生按总成绩进行排名，学生名单都是按照学号排好顺序的，对学生按成绩排名时要求学生名单的顺序不变，即仍然是按照学号进行排列的，但还要得到每个学生对应的名次，如下例所示：

待排序数据：（ 596 560 480 616 560 580 498 500 540 610）

对应的名次：（ 3 5 10 1 5 4 9 8 7 2 ）

对于实际工作中类似的排序要求，原有的一些基础排序算法就不能够直接应用，在具体的编程应用中可以在一些经典排序方法的基础上适当做一些改进，来解决具体的问题。本文基于选择排序算法的基本思想，给出了对不改变数据位置而对数据进行排序的有效算法，并利用C#语言编程给出了该算法实现的动态演示[2]。

# 1 算法实现

## 1.1 算法描述

假设要处理的数据存放在A数组中，为了我们算法实现的需要，我们定义A数组为A[2][n]，其中A[0][0..n-1]存放需要确定名次的数据，A[1][0..n-1]存放排序后各数据对应的顺序，关键字值相同的数据的顺序是相同的。首先我们假定所有数据的原始顺序名次都是0，然后利用选择排序的思想一一确定每个数据对应的顺序，假定m为下一个待确定数据的顺序值，m的初值为1，其过程为：从所有还未确定顺序的数据中查找关键字值最大的数据，确定其顺序为m，同时查找与其关键字值相同的数据，并使其顺序与已经确定的数据相同，并记录关键字值相同的数据个数p，下一个待确定的数据的顺序即为m+p。重复上述过程，直至所有数据的顺序都确定完毕[3]。

## 1.2 实现代码

**输入**：输入数组A[0..1][0..n-1]，其中A[0][0..n-1]中是要排序的数据，A[1][0..n-1]为排序后各数据的顺序

**输出**：确定了数据顺序的数组A[0..1][0..n-1]

A[1][0..n-1]=0; //假定所有数据的初始顺序为0

m=1; //下一个待确定的数据的顺序为m，初始值1

while(m<=n)

{

for i←0 to n-1 //查找下一个还没有确定顺序的数据

{

如果数据a[0][i]的顺序a[1][i]还未确定

{

k=i; //假设i位置上的数据为下一个待确定顺序的数据

for j←i to n-1 //在其余的数据中查找关键字值最大的数据

{

如果第j个数据的顺序还未确定，且其关键字值比前面的大

k=j; //确定该位置上的数据为下一个要确定顺序的数据

}

a[1][k]=m; //确定第k个数据的顺序

p=1; //假设同数据的个数为1个

for j←k+1 to n-1 //查找关键字值相同的数据并进行处理

{

如果第j个数据的顺序还没有确定且其关键字值与第k个数据相同

{ a[1][j]=a[1][k]; //两个数据的位次相同

p++; //关键字值相同的数据个数+1

}

}

m+=p; //设置下一个数据的顺序

break;

}

}

}

输出数组A[0..1][0..n-1]；

# 2 动态演示程序

在.NET开发环境中生成一个Windows应用程序，在窗体的界面上放置一个面板控件Panel，作为展示动态排序过程的演示窗口，添加四个Timer控件，适当设置Timer控件的Interval属性，分别用来代替算法中的四个循环，以Timer控件的定时特点来达到动态演示的效果。程序中的关键代码如下：

//类中定义的有关变量：

private const int MAXNUMBERS = 10;//可以设置的最大数组容量

private int[,] numbers=new int[2,MAXNUMBERS]; //存放待处理的数据及位次顺序

private System.Windows.Forms.Label[] lblMess = new Label[2];

//显示数据用的标签控件数组

private System.Windows.Forms.Label[,] lblNumbers = new Label[2,MAXNUMBERS];

private int mOuter; //控制外层循环

private int iInner; //控制第二层内层循环

private void tmrSame\_Tick(object sender, EventArgs e)

{ if (pInner < 10)

{ if (numbers[1, pInner] == 0 && numbers[0, pInner] == numbers[0, kSearch])

{ numbers[1, pInner] = numbers[1, kSearch];

lblNumbers[1, pInner].Text = lblNumbers[1, kSearch].Text;

lblNumbers[0, pInner].ForeColor = Color.Black;

lblNumbers[0, pInner].BackColor = SystemColors.Control;

lblNumbers[1, pInner].ForeColor = Color.Black;

lblNumbers[1, pInner].BackColor = SystemColors.Control;

nCount++;

}

pInner++;

}

else{ mOuter += nCount;

tmrSame.Enabled = false;

tmrOuter.Enabled = true;

}

}

# 3 程序运行界面效果

该程序运行的界面效果如图1所示，已经确定了位次的数据和还没有确定好的数据以不同的颜色进行了区分，非常直观，见图1：



图1 动态演示程序界面

# 4 结语

本文给出了基于选择思想的不改变数据的原始位置而对数据进行排序的算法并利用C#语言编程实现了该算法的动态演示，该算法可用于解决实际工作中的一些相关问题，具有一定的实际意义。用C#语言实现的动态演示程序则有助于读者更好地理解和把握该算法的基本思想和实现过程。

# 参考文献：

[1]严蔚敏,吴伟民. 数据结构(C语言版)[M]. 北京:清华大学出版社,1997.

[2]M.H.Alsuwaiyel,著,吴伟,方世昌等,译. 算法设计技巧与分析. 北京:电子工业出版社,2004,8

[3]李继武. Visual C#.NET项目开发实战[M]. 北京:清华大学出版社,2007.

[4]林邦杰. 深入浅出C#程序设计[M]. 北京：中国铁道出版社,2005.

# 作者简介：

宁宁（1982-），男，山东青州人，硕士，讲师，研究方向为最优化理论、算法设计与分析

通信作者：张霞（1986-），女，河南郑州人，本科，工程师，研究方向为卷烟机主机、卷烟机电控、工业智能化，E-mail: zhangxia@xx.com

\*通信作者非必须。

非公开发表部分：

姓名：宁宁

**联系地址**：山东（青州市）潍坊教育学院信息工程系

**邮编**：262500

**电子邮箱：**[funing@163.com](mailto:funing@163.com)

**联系电话：**0536-3287222 13863888888

**QQ：**123456 //请务必留QQ号，以便编辑通知

投稿咨询 现代计算机编辑小月 QQ 2676300211