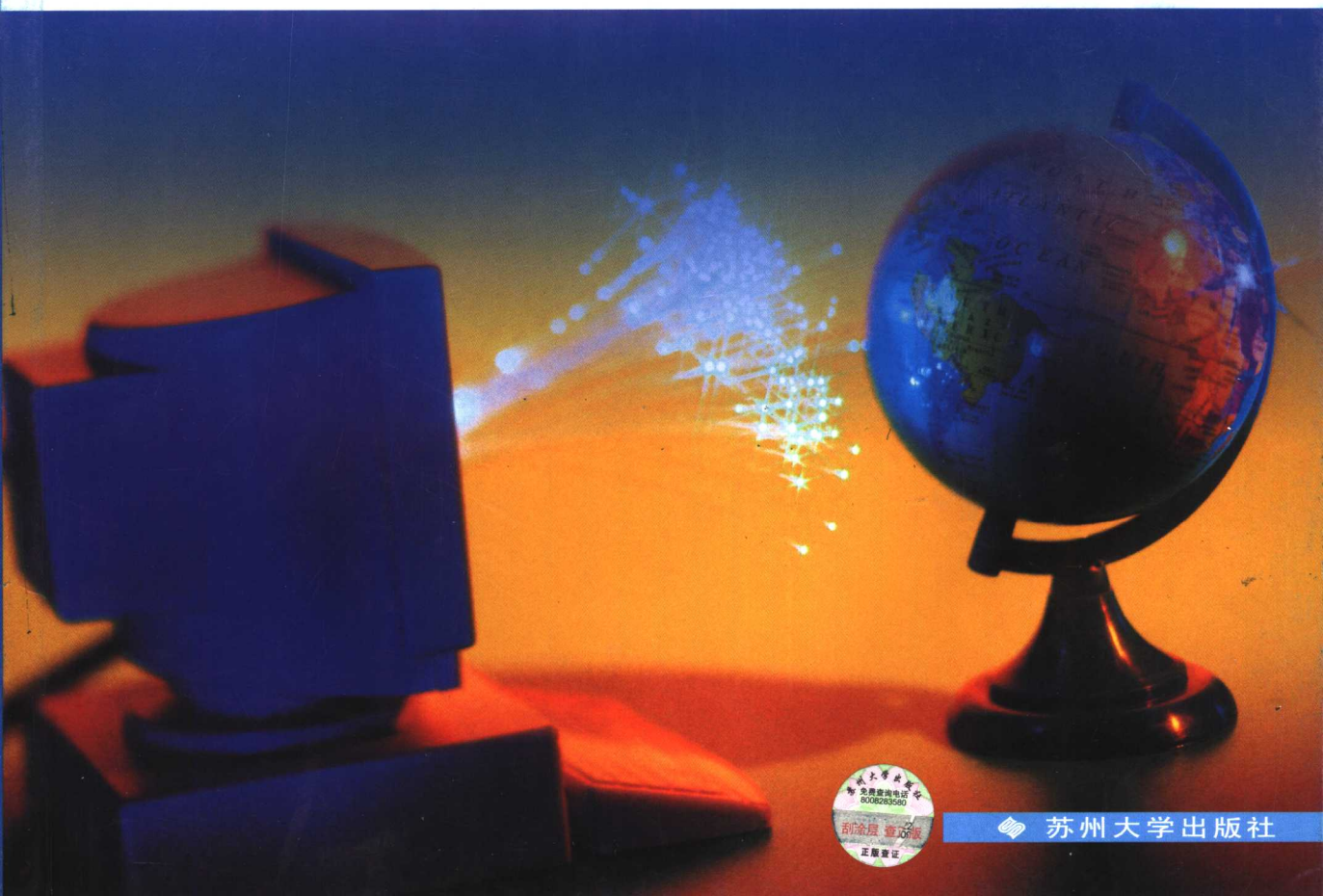


江苏省计算机等级考试辅导用书

计算机上机考试 题型归纳与解析

二级 Visual C++



苏州大学出版社

江苏省计算机等级考试辅导用书

计算机上机考试 题型归纳与解析

(二级 Visual C++)

主 编 骆 健 时兆武

主 审 李千目



苏州大学出版社

图书在版编目(CIP)数据

计算机上机考试题型归纳与解析. 二级 Visual C++/
骆健, 时兆武主编. —苏州: 苏州大学出版社, 2006. 8
江苏省计算机等级考试辅导用书
ISBN 7-81090-727-1

I. 计… II. ①骆…②时… III. ①电子计算机—
水平考试—解题②C 语言—程序设计—水平考试—解题
IV. TP3—44

中国版本图书馆 CIP 数据核字(2006)第 095842 号

计算机上机考试题型归纳与解析
(二级 Visual C++)

骆 健 时兆武 主编
责任编辑 苏 秦

苏州大学出版社出版发行
(地址: 苏州市干将东路 200 号 邮编: 215021)
常熟高专印刷有限公司印装
(地址: 常熟市元和路 98 号 邮编: 215500)

开本 787mm×1092mm 1/16 印张 41.75(共五册) 字数 1020 千
2006 年 8 月第 1 版 2006 年 8 月第 1 次印刷
ISBN 7-81090-727-1/TP·48 定价: 65.00 元
(共五册)

苏州大学版图书若有印装错误, 本社负责调换
苏州大学出版社营销部 电话: 0512-67258835

《计算机上机考试题型归纳与解析》

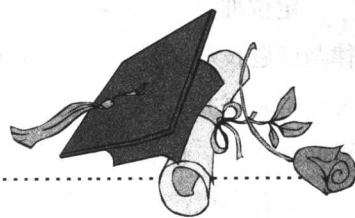
编 委 会

编 委 (按姓氏笔画为序)

王宏华	王国全	王景玉	尹 静
朱贵喜	严云洋	杨章静	李千目
李勇智	李筱松	吴 婷	时兆武
张居晓	陈 智	周 松	赵 明
姚昌顺	骆 健	黄庆宏	葛武滇

本册主编 骆 健 时兆武

主 审 李千目



前

言

“江苏省计算机等级考试”是面向省内高校非计算机专业学生的计算机水平考试,经过多年的发展,已形成了较大规模,具有较大的影响力。

考试分一级、二级、三级3个级别,一级为上机考试,二级分笔试与上机两部分,三级为笔试,笔试和上机考试均要求达到一定标准后考试才算通过。从历年的考试情况来看,上机考试的通过率总体不如笔试,从考生的应试心理分析,由于多数考生平时上机实践的机会较少,训练强度不够,因此更畏惧上机考试。针对这些情况,我们经过精心策划,组织了一批高校计算机基础教学的一线教师,通过对考试大纲和历年考题的深入调研,归纳了上机考试的各类试题题型并加以详细解析,编写了这套《计算机上机考试题型归纳与解析》,旨在帮助广大考生进行针对性的考前集训,强化训练,顺利过关。

本套书的几大特点是:

题型归纳全面,即较全面地归纳出历年的常考题型,并配以典型例题解析,通过“考点点拨”、“理论链接”等特色栏目,让读者熟悉、理解和掌握各类题型,做到心中有数。

试题解析详尽,精心设计了多套上机模拟试题供考生考前集训,所有试题均给出了详细的解答,以便于考生自学自测,做到一书在手,考试无忧。

定位准确、应试性强,在摸清考生应试心理的情况下,通过全面归纳题型来揭示命题规律与解题技巧,提供相当数量的实战训练和备考导航,从而突出针对性和实战性。

本套书共5册:一级信息技术、二级 Visual Basic、二级 Visual FoxPro、二级 C 语言、二级 Visual C++。作者均为省内多所高校长期从事相关课程教学、考试辅导的一线教师。

本套书以广大应试考生为主要读者对象,同时适于相关课程学习的各类读者参考使用。书中部分原型电子素材可到苏州大学出版社网站(www.sudapress.com)查询。

由于时间和水平的原因,书中可能仍有不当之处,欢迎读者批评指正。

《计算机上机考试题型归纳与解析》编委会

目

录

第一篇 常考题型归纳分析

一、数值类	(1)
题型1 排序问题 ★★	(1)
题型2 最值问题 ★★	(8)
题型3 素数问题 ★★	(12)
题型4 数位分离 ★	(15)
题型5 其他数学问题	(22)
二、字符串类	(32)
题型1 字符串与整数问题 ★★	(32)
题型2 出现次数 ★★	(35)
题型3 插入删除字符问题 ★★	(38)
题型4 子串问题 ★★	(42)
题型5 其他问题	(45)
三、二维数组类	(52)
题型1 最值问题 ★★	(52)
题型2 矩阵运算 ★	(55)
题型3 对角线问题 ★	(59)
题型4 行列变化问题 ★	(61)
题型5 其他问题	(64)
四、链表类	(68)
题型1 结点插入问题 ★	(68)
题型2 结点删除问题 ★	(70)

第二篇 模拟试卷精选精解

上机模拟试卷一	(74)
上机模拟试卷二	(78)
上机模拟试卷三	(81)
上机模拟试卷四	(84)
上机模拟试卷五	(87)
上机模拟试卷六	(91)

2

第一篇 常考题型归纳分析

一、数值类

题型1 排序问题 ★★

考点点拨：解决排序问题的算法有很多,例如:选择排序、插入排序、快速排序等都是经常考查的知识点。

典型题1(改错题):

void sort(int A[], int n) 函数功能是将数组 A 中所有小于等于第一个元素的元素排列到第一个元素的左端,所有大于等于第一个元素的元素排列到第一个元素的右端。

例如,初始时数组中元素为: 48,26,68,72,22,18,12,排序后为: 22,26,12,18,48,72,68。

含有错误的源程序如下:

```
1  #include <iostream. h >
2  void sort(int A[], int n)
3  {
4      int i,j,t;
5      i = 0; j = n;
6      do
7      {
8          do i ++; while(A[i] < A[0]);
9          do j ++; while (A[j] > A[0]);
10         if (i < j) {t = A[j]; A[i] = A[j]; A[j] = t; }
11         } while (i < j);
12         t = A[0]; A[0] = A[j]; A[j] = t;
13     }
14     main( )
15     {
16         int a[] = {48,26,68,72,22,18,12}, n = 7, i;
17         sort(a,n);
18         for(i = 0; i < n; i ++)
```

20 }

【要求】

1. 将上述程序录入到文件 myfa. cpp 中, 根据题目要求及程序中语句之间的逻辑关系对程序中的错误进行修改。程序中的注解可以不输入。

2. 改错时,可以修改语句中的一部分内容,调整语句次序,增加少量的变量说明、函数原型说明或编译预处理命令,但不能增加其他语句,也不能删去整条语句。

3. 改正后的源程序 myfa.cpp 必须放在考试盘的根目录下,供阅卷用。

【参考答案】

1. 第9行 `do j++;` 改为 `do j--;`

2. 第 10 行 $t = A[j];$ 改为 $t = A[i];$

【分析】

程序的基本思想是:初始时, i 指向数组的左端, j 指向数组的右端。在第一个 do-while 循环中, i 从左向右扫描数组,找到大于第一个元素的元素时停下(由第二个 do-while 循环实现)。然后, j 从右向左扫描数组,找到小于或等于第一个元素的元素时停下(由第三个 do-while 循环实现)。若此时, i 的值小于 j ,则 i, j 所指向的元素交换位置。当 i 的值小于 j 时,第一个 do-while 循环继续执行; i 在上次位置的基础上继续从左向右扫描数组, j 在上次位置的基础上继续从右向左扫描数组……当出现 i 的值大于或等于 j 时,第一个 do-while 循环结束。 i 所指的元素与数组的第一个元素交换位置。

典型题 2(编程题):

建立一个类 Array,动态生成数组,并使用选择排序算法对数组进行升序排序。具体要求如下:

1. 私有数据成员。

int * a: 指向根据 len 动态申请的数组空间;

int len: 存放数组元素的个数。

2. 公有成员函数。

Array(int b[],int length):构造函数,使用数组 b 初始化 a 所指的动态数组,length 初始化 len;

void sort(): 对动态数组中的数组进行升序排序;

void print(): 输出 a 指向的数组;

~ Array(): 析构函数, 释放动态数组空间。

3. 在主函数中定义整型数组 b, 初始数据为 {12, 45, 7, 8, 96, 4, 10, 48, 2, 46}。定义一个 Array 类的对象 test, 调用成员函数完成排序和输出。

【要求】

将源程序取名为 myfb. cpp, 并放在考试盘的根目录下, 供阅卷用。

【参考答案】

```
#include <iostream.h>
```

class Array

```

{
    private:
        int * a;
        int len;
    public:
        Array(int b[],int length)
        {
            a = new int[ length ];
            len = length;
            for( int i = 0; i < len; i ++ )
                a[ i ] = b[ i ];
        }
        void sort();
        void print();
        ~ Array()
        {
            if( a ) delete[] a;
        }
};

void Array::sort()
{
    int small,i,j,t;
    for( i = 0; i < len - 1; i ++ )
    {
        small = i;
        for( j = i + 1; j < len; j ++ )
            if( a[ j ] < a[ small ] )
                small = j;
        t = a[ i ];
        a[ i ] = a[ small ];
        a[ small ] = t;
    }
}

void Array::print()
{
    cout << "the new array:" << endl;
    for( int i = 0; i < len; i ++ )
        cout << a[ i ] << '\t';
}

void main()
{

```

```
int b[ 10 ] = { 12,45,7,8,96,4,10,48,2,46 } ;
Array test( b,10 ) ;
test. sort( ) ;
test. print( ) ;
```

【分析】

选择排序的算法思想是先从要排序的数中选择最小的数,将它放在第一个位置,然后从剩下的数中选择最小的数放在第二个位置,如此继续,直到最后从剩下的两个数中选择最小的数放在倒数第二个位置,剩下的一个数放在最后位置,排序完成。

典型题 3(改错题):

void Merge(int A[], int B[], int Temp[], int len_a, int len_b) 函数的功能是: 将两个升序序列 a 和 b 合并产生一个新的升序序列 temp。

含有错误的源程序如下:

```

1  #include <iostream. h >
2  void Merge( int A[ ],int B[ ],int Temp[ ],int len_a,int len_b)
3  {
4      int i =0,j =0,k;
5      while( ( i < len_a)&&( j < len_b) )
6      {
7          if( A[ i ] <= B[ j ] ) Temp[ k ++ ] = A[ i ++ ];
8          Temp[ k ++ ] = B[ j ++ ];
9      }
10     while( i > len_a ) Temp[ k ++ ] = A[ i ++ ];
11     while( j < len_b ) Temp[ k ++ ] = B[ j ++ ];
12 }
13 main( )
14 {
15     int a[ ] = { 8,12,45,78 } ,b[ ] = { 1,23,28,79,90 } ,temp[ 9 ] ;
16     int alen =4,blen =5,i;
17     Merge( a,b,temp,alen,blen) ;
18     for( i =0; i < alen + blen; i ++ )
19         cout << " " << temp[ i ] ;
20 }

```

【要求】

1. 将上述程序录入到文件 `myfa.cpp` 中,根据题目要求及程序中语句之间的逻辑关系对程序中的错误进行修改。程序中的注解可以不输入。
2. 改错时,可以修改语句中的一部分内容,调整语句次序,增加少量的变量说明、函数原型说明或编译预处理命令,但不能增加其他语句,也不能删去整条语句。
3. 改正后的源程序 `myfa.cpp` 必须放在考试盘的根目录下,供阅卷用。

【参考答案】

- | | | | |
|---------|-----------------------------------|----|--|
| 1. 第4行 | <code>int i=0,j=0,k;</code> | 改为 | <code>int i=0,j=0,k=0;</code> |
| 2. 第8行 | <code>Temp[k++] = B[j++]</code> ; | 改为 | <code>else Temp[k++] = B[j++]</code> ; |
| 3. 第10行 | <code>while(i > len_a)</code> | 改为 | <code>while(i < len_a)</code> |

【分析】

在初始执行 Merge 函数时,变量 *i* 指向 *a* 序列起始位置,变量 *j* 指向 *b* 序列起始位置。在第一个 while 循环中,*i* 当前所指向的 *a* 序列中的元素与 *j* 当前所指向的 *b* 序列中的元素进行比较,较小者放入 temp 序列。当 *i* 或 *j* 指向相应序列终止位置后,循环将结束,原因是 *a*、*b* 两个序列中已有一个序列的所有元素都被比较完了。所以,另一序列所剩元素可通过循环直接加入到 temp 中。

典型题 4(编程题):

建立一个类 Array,动态生成数组,并使用冒泡排序算法对数组进行升序排序。具体要求如下:

1. 私有数据成员。
`int *a`: 指向根据 `len` 动态申请的数组空间;
`int len`: 存放数组元素的个数。

2. 公有成员函数。

`Array(int b[],int length)`: 构造函数,使用数组 *b* 初始化 *a* 所指的动态数组, `length` 初始化 `len`;

`void sort()`: 对动态数组中的数组进行升序排序;

`void print()`: 输出 *a* 指向的数组;

`~Array()`: 析构函数,释放动态数组空间。

3. 在主函数中定义整型数组 *b*,初始数据为 {12,45,7,8,96,4,10,48,2,46}。定义一个 Array 类的对象 *test*,调用成员函数完成排序和输出。

【要求】

将源程序取名为 `myfb.cpp`,并放在考试盘的根目录下,供阅卷用。

【参考答案】

```
#include <iostream.h>
class Array
{
private:
    int *a;
    int len;
public:
    Array(int b[],int length)
    {
        a = new int[ length ];
        len = length;
        for( int i = 0; i < len; i ++ )
```

```

    a[i] = b[i];
}

void sort();
void print();
~Array()
{
    if(a) delete[] a;
}
;

void Array::sort()
{
    int i,j,t;
    for(i=len-1;i>0;i--)
        for(j=0;j<i;j++)
            if(a[j]>a[j+1])
                {
                    t=a[j];
                    a[j]=a[j+1];
                    a[j+1]=t;
                }
}

void Array::print()
{
    cout<<"the new array:"<<endl;
    for(int i=0;i<len;i++)
        cout<<a[i]<<'\\t';
}

void main()
{
    int b[10]={12,45,7,8,96,4,10,48,2,46};
    Array test(b,10);
    test.sort();
    test.print();
}

```

【分析】

冒泡排序基本算法思想: 相邻两个数进行比较, 若发现逆序(前一个元素大于后一个元素)则进行交换, 最后使最大者换到序列的最后面, 第一趟比较结束。重复上述过程, 从第一个数开始比较、交换, 一直比较到倒数第二个数, 第二趟比较结束……比较 $n-1$ 趟后算法结束。

典型题 5(改错题):

函数 void Bubble(int A, int n)的功能是：使用改进的冒泡排序算法对序列进行升序排

序,如果在某一趟排序中未交换元素,说明序列已经有序,则不再进行下一趟排序,整个算法结束。

含有错误的源程序如下:

```

1  #include <iostream. h >
2  void Bubble(int A, int n)
3  {
4      int i,j,last,t;
5      i = n - 1;
6      while(i > 0)
7      {
8          last = i;
9          for(j = 0; j < i; j++)
10             if(A[j + 1] > A[j])
11                 {
12                     t = A[j];
13                     A[j] = A[j + 1];
14                     A[j + 1] = t;
15                     last = j;
16                 }
17             i = last;
18         }
19     }
20     main()
21     {
22         int a[] = {90,8,12,23,28,45,79,78}, n = 8;
23         int i;
24         Bubble(a,n);
25         for(i = 0; i < n; i++)
26             cout << a[i] << '\t';
27     }

```

【要求】

1. 将上述程序录入到文件 myfa. cpp 中,根据题目要求及程序中语句之间的逻辑关系对程序中的错误进行修改。程序中的注解可以不输入。

2. 改错时,可以修改语句中的一部分内容,调整语句次序,增加少量的变量说明、函数原型说明或编译处理命令,但不能增加其他语句,也不能删去整条语句。

3. 改正后的源程序 myfa. cpp 必须放在考试盘的根目录下,供阅卷用。

【参考答案】

- 第 2 行 int A, int n 改为 int A[], int n
- 第 8 行 last = i; 改为 last = 0;
- 第 10 行 if(A[j + 1] > A[j]) 改为 if(A[j + 1] < A[j])

【分析】

for 循环控制的是一趟排序中元素的两两比较。如果在某一趟排序中不存在逆序(前一个元素比后一个元素大时称为逆序),不需要进行元素交换,if 语句将不会被执行,last 的值也不会被修改,仍然为 0,这个值赋给 i 后,整个 while 循环结束。

题型2 最值问题 ★★

考点点拨: 考查求最大值、最小值、次大值等相关问题。

典型题 6(改错题):

函数 `int fun(int *a, int n)` 的功能是：求数组 `a` 中的最大数和次大数，并将最大数与 `a[0]` 中的元素交换，次大数与 `a[1]` 中的元素交换。

含有错误的源程序如下:

```

1  #include <iostream. h >
2  #define N 10
3  void fun(int * a,int n)
4  {
5      int i,m,t,k;
6      for(i=0;i < 2;i ++ )
7      {
8          m=0;
9          for( k = i;k <= N;k ++ )
10             if( a[ k ] > a[ m ] ) m += k;
11             t = a[ i ] ; a[ i ] = a[ m ] ; a[ m ] = t;
12         }
13     }
14     main( )
15     {
16         int b(N) = { 1,5,12,0,3,6,9,7,10,8 } , i;
17         for( i=0;i < N;i ++ )
18             cout << b[ i ] << '\t' ;
19         cout << endl ;
20         fun( b,N ) ;
21         for( i=0;i < N;i ++ )
22             cout << b[ i ] << '\t' ;
23     }

```

【要求】

1. 将上述程序录入到文件 `myfa.cpp` 中,根据题目要求及程序中语句之间的逻辑关系对程序中的错误进行修改。程序中的注解可以不输入。
2. 改错时,可以修改语句中的一部分内容,调整语句次序,增加少量的变量说明、函数原型说明或编译处理命令,但不能增加其他语句,也不能删去整条语句。

3. 改正后的源程序 myfa. cpp 必须放在考试盘的根目录下,供阅卷用。

【参考答案】

- | | | | |
|-----------|---|----|--|
| 1. 第 8 行 | <code>m = 0;</code> | 改为 | <code>m = i;</code> |
| 2. 第 9 行 | <code>for(k = i; k <= N; k++)</code> | 改为 | <code>for(k = i; k < N; k++)</code> |
| 3. 第 10 行 | <code>m += k;</code> | 改为 | <code>m = k;</code> |
| 4. 第 16 行 | <code>int b(N)</code> | 改为 | <code>int b[N]</code> |

【分析】

外层 for 循环将执行两遍,执行第一遍找出整个数组的最大数,并将此数与数组的第一个元素 `a[0]` 交换。执行第二遍时,找出除 `a[0]` 外其他元素中的最大数(找出除最大数之外的最大数,即为次大数)。

【理论链接】

本题求最大值、次大值的方法还可应用于对最小值、次小值等问题进行求解。

典型题 7(编程题):

建立一个类 Array,动态生成数组,并求出数组中最大数和最小数,将最大数和最小数对调。具体要求如下:

1. 私有数据成员。

`int *a`: 指向根据 `len` 动态申请的数组空间;

`int len`: 存放数组元素的个数。

2. 公有成员函数。

`Array(int b[], int length)`: 构造函数,使用数组 `b` 初始化 `a` 所指的动态数组, `length` 初始化 `len`;

`void exchange()`: 求出数组中的最大数和最小数,将最大数和最小数交换;

`void print()`: 输出 `a` 指向的数组;

`Array& operator = (Array & r)`: 重载运算符 `&`,实现将数组 `r` 整体赋值给当前调用本函数的数组对象;

`~Array()`: 析构函数,释放动态数组空间。

3. 在主函数中使用数据 `{12,45,7,8,96,4,10,48,2,46}` 初始化类 Array 的对象 `test1`、`test2`, `test1` 调用成员函数 `exchange` 并输出结果,然后通过赋值运算符将 `test1` 的元素整体赋值给 `test2`。

【要求】

将源程序取名为 myfb. cpp,并放在考试盘的根目录下,供阅卷用。

【参考答案】

```
#include <iostream. h >
class Array
{
private:
    int *a;
    int len;
```

```

public:
    Array(int b[ ],int length)
    {
        a = new int[ length ] ;
        len = length;
        for( int i = 0; i < len; i ++ ) a[ i ] = b[ i ] ;
    }

    void exchange( ) ;
    void print( ) ;

    Array& operator = ( Array & r ) ;
    ~ Array( )
    {
        if( a ) delete[ ] a ;
    }
};

void Array::exchange( )
{
    int k,max,min,t;
    max = min = 0;
    for( k = 1; k < len; k ++ )
    {
        if( a[ k ] > a[ max ] )
        {
            max = k;
        }
        if( a[ k ] < a[ min ] )
        {
            min = k;
        }
    }
    t = a[ max ] ; a[ max ] = a[ min ] ; a[ min ] = t ;
}

void Array::print( )
{
    cout << "the new array: " << endl;
    for( int i = 0; i < len; i ++ ) cout << a[ i ] << '\t' ;
}

Array& Array::operator = ( Array & r )
{
    if( this != &r )
    {
        len = r. len ;
        delete[ ] a ;
    }
}

```