

全国计算机等级考试



历届上机真题详解

三级 C 语言程序设计

(2008版)

全国计算机等级考试命题研究组 编

南开大学出版社



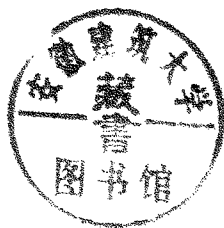
全国计算机等级考试

历届上机真题详解

三级 C 语言程序设计

(2008 版)

全国计算机等级考试命题研究组 编



南开大学出版社

天津

图书在版编目(CIP)数据

全国计算机等级考试历届上机真题详解: 2008 版. 三级 C 语言程序设计 / 全国计算机等级考试命题研究组编. 4 版. —天津: 南开大学出版社, 2007. 11

ISBN 978-7-310-02285-4

I. 全… II. 全… III. ①电子计算机—水平考试—试题
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(2007)第 154575 号

版权所有 侵权必究

南开大学出版社出版发行

出版人: 肖占鹏

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

河北昌黎太阳红彩色印刷有限责任公司印刷

全国各地新华书店经销

*

2007 年 11 月第 4 版 2007 年 11 月第 4 次印刷

787×1092 毫米 16 开本 14.25 印张 352 千字

定价: 27.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

内容提要

本书提供了全国计算机等级考试三级 C 语言程序设计机试题库，共 100 道 C 语言程序设计考试题的分析和答案，并指出考核的知识点、重点难点、解题思路、程序流程，另外，针对许多题目提出了更为有效便捷的解法，使读者可以开阔思路，学习编程思想与技巧。本书配套光盘主要包含两部分内容：（1）C 语言上机考试的全真模拟环境，本书所有试题均可在该环境中进行答题和评分，以此进行考前强化训练；（2）考试过程的录像动画演示，从登录、答题到交卷，均有指导教师的全程语音讲解。

本书针对参加全国计算机等级考试三级网络技术、三级信息技术、三级数据库技术科目机试的考生，同时也可以作为普通高校、大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

前 言

全国计算机等级考试 (National Computer Rank Examination, NCRE) 是由教育部考试中心主办, 用于考查应试人员的计算机应用知识与能力的考试。本考试的证书已经成为许多单位招聘员工的一个必要条件, 具有相当的“含金量”。

为了帮助考生更顺利地通过计算机等级考试, 我们做了大量市场调查工作, 根据考生的备考体会, 以及培训教师的授课经验, 推出了《历届上机真题详解——三级 C 语言程序设计》一书。本书主要由如下部分组成。

一、历届真题及详尽解析

对于备战等级考试而言, 做题, 是进行考前冲刺的最佳方式。这是因为它的针对性相当强, 考生可以通过做题的实际练习, 来检验自己是否真正掌握了相关知识点, 了解考试重点, 并且根据需要再对知识结构的薄弱环节进行强化。

本书提供了三级 C 语言程序设计的历届真题, 并在每道题的后面, 提供了参考答案和详细分析。在解析中, 精解考点, 分析题眼, 详解重点难点, 并给出应试技巧。

二、上机考试必备知识

在这一部分中, 概括了三级 C 语言程序设计上机考试的技巧与策略, 内容涉及如何准备上机考试, 上机考试的注意事项以及上机考试的过程。

三、机试全真模拟环境配套光盘

本书配套光盘物超所值, 主要包含两部分内容: (1) 上机考试的全真模拟环境, 本书所有试题均可在该环境中进行答题和评分, 以此进行考前强化训练; (2) 考试过程的录像动画演示, 从登录、答题到交卷, 均有指导教师的全程语音讲解。

本书针对参加全国计算机等级考试三级网络技术、三级信息技术和三级数据库技术上机考试的考生, 同时也可以作为普通高校、大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

为了保证本书及时面市和内容准确, 很多朋友做出了贡献, 陈河南、贺民、许伟、侯佳宜、贺军、于樊鹏、戴文雅、戴军、李志云、陈安南、李晓春、王春桥、王雷、韦笑、龚亚萍、冯哲、邓卫、唐玮、魏宇、李强等老师付出了很多辛苦, 在此一并表示感谢!

在学习的过程中, 您如有问题或建议, 请使用电子邮件与我们联系。或登录百分网, 在“书友论坛”与我们共同探讨。

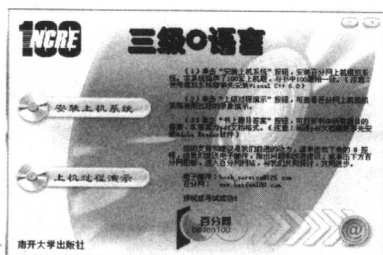
电子邮件: book_service@126.com

百分网: www.baifen100.com

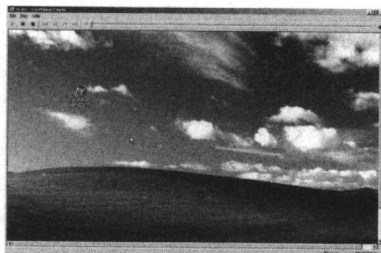
全国计算机等级考试命题研究组

2007 年 10 月

配套光盘说明



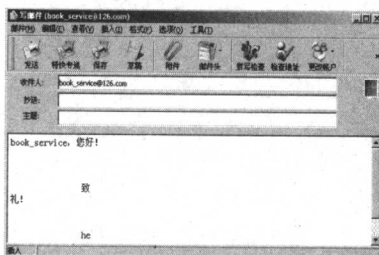
光盘初始启动界面, 可选择安装上机系统、查看上机操作过程



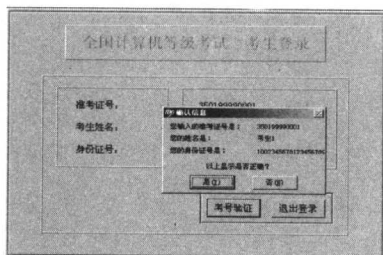
上机操作过程的录像演示, 有指导教师的全程语音讲解



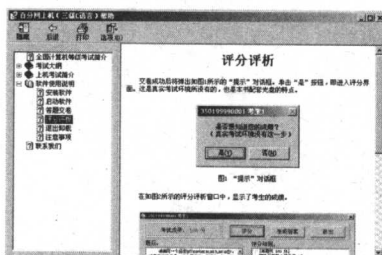
单击光盘初始界面的“百分网”图标, 可进入百分网, 您可以在这里与我们共同探讨问题



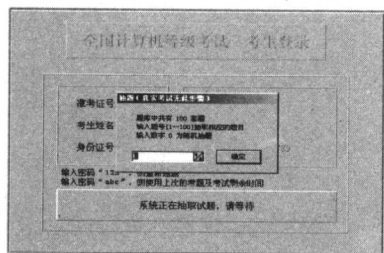
单击光盘初始界面左下角的“邮件”图标, 您可以给我们发送邮件, 提出您的建议和意见



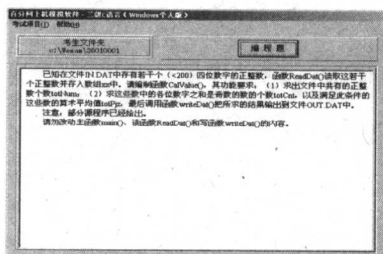
考生登录界面: 在此输入准考证号等考生信息



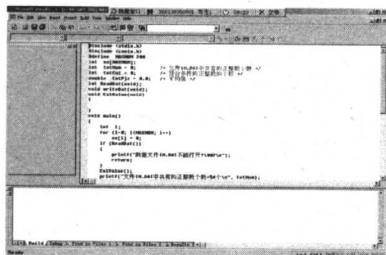
从“开始”菜单可启动帮助系统, 在这里可看到考试简介、考试大纲以及详细的软件使用说明



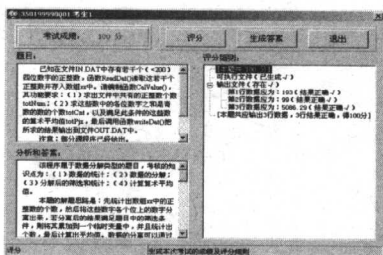
您可以随机抽题, 也可以指定固定的题目



浏览题目界面, 查看考试题目, 单击“考试项目”开始答题



在实际环境中答题, 完成后单击工具栏中的“交卷”按钮



答案和分析界面, 查看所考核题目的答案和分析

目 录

第一部分 历届真题及解析	1
第二部分 上机考试必备知识	212
2.1 上机考试注意事项	212
2.1.1 复习要点	212
2.1.2 考试要点	212
2.2 上机考题分类	212
2.3 上机考试过程	213
2.3.1 登录过程	214
2.3.2 考试过程	215
2.3.3 交卷评分	217

第一部分 历届真题及解析

★★

第1题

请编写一个函数 num(int m,int k,int xx[]), 该函数的功能是: 将大于整数 m 且紧靠 m 的 k 个素数存入数组 xx 传回。

最后调用函数 readwriteDat()读取 10 组数据, 分别得出结果且把结果输出到文件 out.dat 中。

例如: 若输入 17 5, 则应输出: 19 23 29 31 37。

部分程序已经给出。

请勿改动主函数 main()和输入输出函数 readwriteDat()的内容。

```
#include <conio.h>
#include <stdio.h>
void readwriteDat();
void num(int m,int k,int xx[])
{
}

main()
{
    int m,n,xx[1000];
    printf("\nPlease enter two integers:");
    scanf("%d%d",&m,&n);
    num(m,n,xx);
    for(m=0;m<n;m++)
        printf("%d ",xx[m]);
    printf("\n");
    readwriteDat();
}

void readwriteDat()
{
    int m,n,xx[1000], i;
    FILE *rf,*wf;
    rf=fopen("in.dat","r");
    wf=fopen("out.dat","w");
    for(i=0;i<10;i++)
    {
        fscanf(rf,"%d %d",&m,&n);
```



```

        num(m,n,xx);
        for (m=0;m<n;m++)
            fprintf(wf,"%d ",xx[m]);
        fprintf(wf,"\n");
    }
    fclose(rf);
    fclose(wf);
}

```

【答案】

```

void num(int m, int k, int xx[])
{
    /* 定义整型变量 data, 并初始化为 m+1 (即大于整数 m 且紧靠 m 的第 1 个整数) */
    int data = m+1;
    int half, n = 0, i;
    while(1)
    {
        /* 判断 data 是否为素数 */
        half = data/2;
        for (i=2; i<=half; i++)
            if (data%i == 0)
                break;
        if (i > half)          /* 如果此时 i>half 说明 data 是一个素数 */
        {
            xx[n] = data;      /* 将 data 存入数组 xx */
            n++;              /* 让计数变量 n 增 1 */
        }
        /* 如果已经找足 k 个素数了, 就通过 break 语句跳出循环 */
        if (n >= k)
            break;
        data++;               /* 继续找下一个素数 */
    }
}

```

【解析】

该程序属于按条件查找类型的题目, 考核的知识点主要为:

- (1) 判断一个数字是否为素数。
- (2) 统计素数的个数。

本题的解题思路为: 从 $m+1$ 开始逐个数字判断是否为素数 (即只能被 1 和它本身整除的整数)。若是素数则存入数组 xx 中, 一直寻找连续的 k 个素数为止。判断是否为素数, 可以通过除以 2 到它一半之间所有的数。比如, 判断 17 是否为素数, 则需除以 2、3、4、5、6、7、8, 若都不能整除, 则该数字就一定是素数。寻找连续的 k 个素数; 可以设置一个记录个数的变量, 每找到一个素数时该记录变量加 1。

程序的流程为: 首先从键盘输入两个十进制整数, 以第一个整数作为基础开始寻找素数,

第2个整数作为需要寻找素数的个数，然后通过 readwriteDat()函数从 in.dat 中读入 10 组数据，通过 num()函数的计算，把结果写回文件 out.dat 中。

在 num()函数中，使用一个 while 循环查找 k 个素数，每次查找成功后 n 值将加 1。因为不能确定循环执行的具体次数，所以设定 while 循环的条件恒为真（条件为“1”），并加入跳出循环的控制语句，即当 $n \geq k$ （超过所需的素数个数）时循环结束。

进入 while 循环体后，首先使用 for 循环，确定当前数字是否为素数，即判断当前数字 data 是否能被 2 到它一半之间的任意数整除，如果能被整除，则说明此数不是素数，程序通过 break 语句跳出 for 循环（此时 $i \leq \text{half}$ ），否则循环执行到 $i > \text{half}$ 时，程序正常退出 for 循环体，说明当前数字 data 是素数，这里的 i 值既作为 for 循环计数器，又作为当前数字是否为素数的标志。

if 语句将根据此时 i 值的大小判断 data 是否为素数，如果是素数，则将 data 存入数组 xx 中，同时记录素数个数的变量 n 加 1，接着判断 n 的值是否达到了题目要求的个数，如果达到了，则跳出 while 循环，num()函数结束，否则 data 加 1 后重新进入 while 循环体。直到找到满足个数的素数，num()函数结束。

★★

第2题

已知数据文件 IN.DAT 中存有 200 个四位数，并已调用读函数 readDat()把这些数存入数组 a 中，请考生编制一函数 jsVal()，其功能是：如果四位数各位上的数字均是 0 或 2 或 4 或 6 或 8，则统计出满足此条件的个数 cnt，并把这些四位数按从大到小的顺序存入数组 b 中。最后 main()函数调用写函数 writeDat()把结果 cnt 以及数组 b 中符合条件的四位数输出到 OUT.DAT 文件中。

注意：部分程序已经给出。

程序中已定义数组：a[200]，b[200]，已定义变量：cnt。

请勿改动数据文件 IN.DAT 中的任何数据、主函数 main()、读函数 readDat()和写函数 writeDat()的内容。

```
#include <stdio.h>
#define MAX 200
int a[MAX], b[MAX], cnt = 0 ;
void jsVal()
{
}
void readDat()
{
    int i ;
    FILE *fp ;
    fp = fopen("in.dat", "r") ;
    for(i = 0 ; i < MAX ; i++)
        fscanf(fp, "%d,", &a[i]) ;
    fclose(fp) ;
```

```

    }
    void writeDat()
    {
        FILE *fp ;
        int i ;
        fp = fopen("out.dat", "w") ;
        fprintf(fp, "%d\n", cnt) ;
        for(i = 0 ; i < cnt ; i++)
            fprintf(fp, "%d\n", b[i]) ;
        fclose(fp) ;
    }
    void main()
    {
        int i ;
        readDat() ;
        jsVal() ;
        printf("满足条件的数=%d\n", cnt) ;
        for(i = 0 ; i < cnt ; i++)
            printf("%d ", b[i]) ;
        printf("\n") ;
        writeDat() ;
    }

```

【答案】

```

void jsVal()
{
    int bb[4];
    int i, j, k, flag;
    for (i=0; i<200; i++) /* 循环以遍历数组 a 中的每一个四位数 */
    {
        bb[0] = a[i]/1000; /* 将 a[i] 的千位数字存入 bb[0] */
        bb[1] = a[i]%1000/100; /* 将 a[i] 的百位数字存入 bb[1] */
        bb[2] = a[i]%100/10; /* 将 a[i] 的十位数字存入 bb[2] */
        bb[3] = a[i]%10; /* 将 a[i] 的个位数字存入 bb[3] */
        for (j=0; j<4; j++) /* 循环以遍历数组 bb, 即 a[i] 的每一位 */
        {
            /* 根据当前位是否能被 2 整除, 来设置旗标变量 flag 为 1 还是 0 */
            if (bb[j]%2 == 0)
            {
                flag = 1;
            }
            else
            {
                flag = 0;
            }
        }
    }
}

```



```

/* 如果有 1 位不能被 2 整除, 后面的位就不必判断了, 跳出该循环 */
break;
}
}
if (flag == 1) /* 如果此时 flag 的值为 1 说明所有位均能被 2 整除 */
{
    b[cnt] = a[i]; /* 将此数存入数组 b */
    cnt++; /* 计数值 cnt 增 1 */
}
}
/* 对数组 b 的前 cnt 个数进行选择排序 */
for (i=0; i<cnt-1; i++)
    for (j=i+1; j<cnt; j++)
        /* 如果 b[i] 因小于它后面的数 b[j] 而交换两者, 说明此排序为降序 */
        if (b[i] < b[j])
        {
            k = b[i];
            b[i] = b[j];
            b[j] = k;
        }
}

```

【解析】

该程序属于数据分解类型的题目, 考核的知识点主要为: (1) 数据的分解; (2) 分解后的筛选和统计; (3) 筛选后的排序。

本题的解题思路是: 首先应该将四位数各个位上的数字分离出来, 然后按题目中的要求进行筛选, 并统计出满足条件的个数, 最后将筛选所得的数据排序。分离数字可以通过“/”和“%”的组合来实现。筛选通过 if 语句实现, 实现时设置一个标记变量 flag, 当条件成立时, 将标记赋值为 1; 统计个数时利用题目中给出的变量 cnt, 每当条件成立时 (标记为 1) 变量 cnt 加 1。最后将筛选后的元素排序, 可以通过双重 for 循环内嵌 if 语句的方式来实现, 排序算法为每次选定一个位置上的元素与它后面的所有元素相比较, 如果比后面的元素小, 则两者交换, 完成比较一趟后, 此位置的元素就比其后面所有的元素都大。比较一直进行到所有元素均被访问后结束, 一共需要比较 cnt 趟, 每趟需要比较 cnt-i 次。

程序的流程为: 先调用了 readDat() 函数读入数据, 并存放在数组 a 中, 然后调用 jsVal() 函数对数据处理后, 由 writeDat() 函数写回文件 OUT.DAT 中。

在 jsVal() 函数中, 通过一个 for 循环遍历整个数组 a, 循环中首先把当前四位数各个位上的数字分离出来。a[i]/1000 得到 a[i] 千位上的数字, a[i]%1000/100 得到百位上的数字, a[i]%100/10 得到十位上的数字, a[i]%10 得到个位上的数字, 以上过程考生应该熟练掌握。

得到各个位上的数字后, 程序进入到下一个内嵌 for 循环体中, 本层 for 循环的功能是通过内嵌 if 语句, 判断分离后的数据是否满足条件, 即 a[i] 各位上的数字 bb[0], bb[1], bb[2], bb[3] 能否均被 2 整除 (也就是题目中要求的各个位上的数字都是 0 或 2 或 4 或 6 或 8), 如能整除, 则把 flag 设置为 1 (flag 为能否整除的标记), 否则将 flag 置为 0, 并跳出本层 for 循环体, 随后 if

语句的功能是将满足条件的数字 $a[i]$ 存入数组 b 中, 并且使记录个数的变量 cnt 加 1, 当外层 for 循环结束后, 即统计出了共有多少个满足条件的数字。

最后对数组 b 中的数据降序排序, 通过双重 for 循环和内嵌 if 语句来实现, 循环结束后, $jsVal()$ 函数结束。

★★

第 3 题

函数 $ReadDat()$ 实现从文件 $IN.DAT$ 中读取一篇英文文章存入字符串数组 xx 中; 请编制函数 $StrOR()$, 其函数的功能是: 以行为单位依次把字符串中所有小写字母 o 左边的字符串内容移到该串的右边存放, 然后把小写字母 o 删除, 余下的字符串内容移到已处理字符串的左边存放, 之后把已处理的字符串仍按行重新存入字符串数组 xx 中。最后 $main()$ 函数调用函数 $WriteDat()$ 把结果 xx 输出到文件 $OUT.DAT$ 中。

例如: 原文: n any field. Yu can create an index
you have the correct record.

结果: n any field. Yu can create an index
rd. yu have the crrect rec

原始数据文件存放的格式是: 每行的宽度均小于 80 个字符, 含标点符号和空格。

注意: 部分源程序已经给出。

请勿改动主函数 $main()$ 、读数据函数 $ReadDat()$ 和输出数据函数 $WriteDat()$ 的内容。

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
char xx[50][80] ;
int maxline = 0 ; /* 文章的总行数 */
int ReadDat(void) ;
void WriteDat(void) ;
void StrOR(void)
{
}

void main()
{
    if(ReadDat())
    {
        printf("数据文件 IN.DAT 不能打开!\n\007") ;
        return ;
    }
    StrOR() ;
    WriteDat() ;
}

int ReadDat(void)
```

```

{
    FILE *fp ;
    int i = 0 ;
    char *p ;
    if((fp = fopen("IN.DAT", "r")) == NULL)
        return 1 ;
    while(fgets(xx[i], 80, fp) != NULL)
    {
        p = strchr(xx[i], '\n') ;
        if(p) *p = 0 ;
        i++ ;
    }
    maxline = i ;
    fclose(fp) ;
    return 0 ;
}

```

```
void WriteDat(void)
```

```

{
    FILE *fp ;
    int i ;
    fp = fopen("OUT.DAT", "w") ;
    for(i = 0 ; i < maxline ; i++)
    {
        printf("%s\n", xx[i]) ;
        fprintf(fp, "%s\n", xx[i]) ;
    }
    fclose(fp) ;
}

```

【答案】

```
void StrOR(void)
```

```

{
    int i, j, k;
    char buf[80];
    for (i=0; i<maxline; i++) /* 循环以遍历英文文章的每一行 */
    {
        j = strlen(xx[i])-1; /* 求出当前行最后一个字符的位置 */
        /* 从最后一个字符往前循环遍历当前行，直到碰到小写字母'o'或到行首 */
        while (xx[i][j]!='o' && j>=0)
            j--;
        if (j < 0) /* 如果 j<0 说明找到行首也没发现一个'o' */
            continue; /* 则当前行不用操作，去下一行 */
        /* 找到字母'o'，用字符串结束标志 0（或写成'\0'）替换之，使该行一分为二 */
        xx[i][j] = 0;
    }
}

```



```

        strcpy(buf, &xx[i][j+1]); /* 将后半截字符串复制到缓存行 buf 中 */
        /* 从第 1 个字符开始循环遍历前半截字符串 */
        k = j = 0;
        while (xx[i][j])
        {
            /* 循环往后遍历, 找出第 1 个小写字母 'o' 的位置或到达字符串末尾 */
            while (xx[i][j]!='o' && xx[i][j])
                j++;
            if (!xx[i][j]) /* 如果 !xx[i][j] 为真, 说明已到达字符串末尾 */
            {
                /* 将找过的这段字符串连接到 buf 中已存字符串的末尾并跳出循环 */
                strcat(buf, &xx[i][k]);
                break;
            }
            /* 如果找到了字母 'o', 用 0 替换之以分割出找过的这段字符串 */
            xx[i][j] = 0;
            /* 将找过的这段字符串连接到 buf 中已存字符串的末尾 */
            strcat(buf, &xx[i][k]);
            /* 从下一个位置开始继续循环遍历剩下的字符串 */
            j++;
            k = j;
        }
        /* 执行到这里, 当前行已遍历完毕, 用操作结果覆盖掉原来内容 */
        strcpy(xx[i], buf);
    }
}
    
```

【解析】

该程序属于字符串处理类型的题目, 考核的知识点主要为: (1) 查找并删除字符串中的指定字符; (2) 对字符串中的元素进行移位。

本题的解题思路是: 以行为单位从后往前查找字母 'o', 每找到一个 'o', 就先将 'o' 替换成字符串结束标志 '\0', 再将后面找过的部分字符串从前往后存入缓存数组 buf 中。这样遍历完成后, 既实现了删除字符 'o', 又实现了字符串的互换。

程序的流程为: 首先调用 ReadDat() 函数读入数据并存放于数组 xx 中, 通过 StrOR() 函数处理后, 由 WriteDat() 函数写回文件 OUT.DAT 种。

在 StrOR() 函数中, 最外层 for 循环的作用是使下面的操作逐行进行, 首先通过库函数 strlen() 得到每行字符串的实际长度并将长度值减 1 换算成最后一个字符的下标值赋给 j, 然后根据 j 值从后往前遍历当前行, 实现上面所述步骤。直到 i=maxline 所有行都处理完毕, 函数结束。

★★

第 4 题

函数 ReadDat() 实现从文件 IN.DAT 中读取一篇英文文章并存入到字符串数组 xx 中, 请编制函数 StrOL(), 其功能是: 以行为单位对行中以空格或标点符号为分隔的所有单词进

行倒排。然后把已处理的字符串（应不含标点符号）仍按行重新存入字符串数组 xx 中，最后调用函数 WriteDat()把结果 xx 输出到文件 OUT.DAT 中。

例如：原文：You He Me

I am a student.

结果：Me He You

student a am I

原始数据文件存放的格式是：每行的宽度均小于 80 个字符，含标点符号和空格。

部分程序已经给出。

请勿改动主函数 main()、读数据函数 ReadDat()和输出数据函数 WriteDat()的内容。

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <ctype.h>
char xx[50][80];
int maxline=0; /*文章的总行数*/
int ReadDat(void);
void WriteDat(void);
void StrOL(void)
{
}
void main()
{
    if(ReadDat())
    {
        printf("数据文件 IN.DAT 不能打开! \n\007");
        return;
    }
    StrOL();
    WriteDat();
}
int ReadDat(void)
{
    FILE *fp;
    int i=0;
    char *p;
    if((fp=fopen("IN.DAT", "r"))==NULL)
        return 1;
    while(fgets(xx[i], 80, fp)!=NULL)
    {
        p=strchr(xx[i], '\n');
        if(p) *p=0;
```

```

        i++;
    }
    maxline=i;
    fclose(fp);
    return 0;
}

void WriteDat(void)
{
    FILE *fp;
    int i;
    fp=fopen("OUT.DAT","w");
    for(i=0;i<maxline;i++)
    {
        printf("%s\n",xx[i]);
        fprintf(fp,"%s\n",xx[i]);
    }
    fclose(fp);
}

```

【答案】

```

void StrOL(void)
{
    int i, j, k, strl, l;
    /* 循环遍历英文文章的全部字符, 将既不是字母也不是空格的字符替换成空格 */
    for (i=0; i<maxline; i++)
        for (j=0; j<(int)strlen(xx[i]); j++)
            if (!isalpha(xx[i][j]) && xx[i][j]!=' ')
                xx[i][j] = ' ';
    for (l=0; l<maxline; l++) /* 循环以遍历文章的每一行 */
    {
        char ch[80] = {0};
        char pp[80] = {0};
        strl = strlen(xx[l]);
        i = strl-1; /* 求出当前行最后一个字符的位置 */
        k = 0;
        while(l)
        {
            /* 从最后一个字符往前循环遍历当前行, 直到碰到非字母字符或到行首 */
            /* (即以从后往前的顺序查找单词) */
            while (isalpha(xx[l][i]) && i>=0)
            {
                /* 将字符数组 pp 中的前 k+1 个字符往后移动一位 */
                for (j=k; j>=0; j--)
                    pp[j+1] = pp[j];
            }
        }
    }
}

```