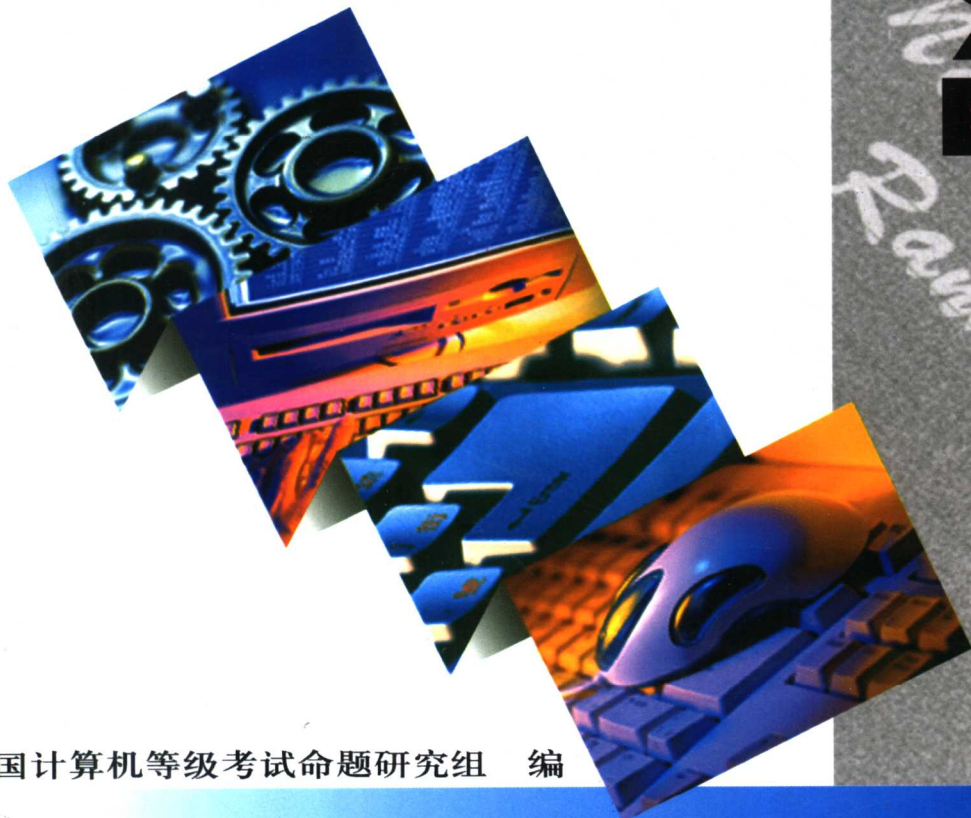


韬略
BESTBOOK

2006

9月考试专用



全国计算机等级考试命题研究组 编

全国计算机等级考试

历届上机试题 · 笔试真题汇编

—— 三级网络技术



上机考试练习系统

- 习题练习，提高上机能力
- 模拟考试，完全贴近真实考试环境
- 大量上机试题均可按真实考试上机操作：评分

笔试练习系统

- 仿真练习，模拟考试
- 归纳总结知识点，详细解析，力求举一反三，触类旁通
- 包括近几年笔试题及参考答案，熟悉考题并掌握知识点

中国大地出版社

全国计算机等级考试辅导丛书

历届上机试题

(2002 年 - 2006 年)

· 笔试真题汇编

三级网络技术

全国计算机等级考试命题研究组 编

中国大地出版社

内 容 简 介

本书是由全国计算机等级考试命题研究组专家编写。全书共有40道历届上机考试真题(每道均配有详细的解析)及2002年9月—2006年4月的历届笔试真题(每套均配有详细的解析),以供考生考前使用。对于备战等级考试而言,做真题是进行考前冲刺的最佳方式。应试导向准确,针对性强,考生只需用少量时间,通过对真题的实际练习,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。考生就能在较短时间内巩固所学知识,掌握要点、突破难点、把握考点、熟练掌握答题方法及技巧,适应考试氛围,顺利通过考试。

图书在版编目(CIP)数据

三级网络技术历届上机试题·笔试真题汇编/全国计算机等级考试命题研究组编. —北京:中国大地出版社,2003.5

(全国计算机等级考试辅导丛书)

ISBN 7-80097-564-9

I. 三... II. 全... III. 计算机网络-水平考试-习题 IV. TP393-44

中国版本图书馆 CIP 数据核字(2003)第 030005 号

丛 书 名: 全国计算机等级考试汇编系列

书 名: 三级网络技术历届上机试题·笔试真题汇编

出版发行: 中国大地出版社

(北京市海淀区大柳树路19号 100081)

责任编辑: 张 雄

经 销: 全国各地新华书店

印 刷: 铁十六局印刷厂

版 次: 2006年5月第1版

印 次: 2006年5月第1次印刷

开 本: 787×1092 1/16 字数: 1900千字

印 张: 150

书 号: ISBN 7-80097-564-9/TP·8

定 价: 300.00元(全套)

(凡购买中国大地出版社的图书,如发现印装质量问题,本社发行部负责调换)



前言

在信息时代,计算机与软件技术日新月异,发展迅猛,渗透到了经济、文化和社会的各个领域,迅速地改变着人们的观念、生活和社会结构。因此,计算机知识的掌握及应用毋庸置疑成了培养新型人才的一个重要环节。

国家教育部考试中心顺应社会发展的需要,于1994年推出“全国计算机等级考试”(简称NCRE),其目的是以考促学,向社会推广普及计算机知识,为选拔人才提供统一、公正、客观和科学的标准。该项考试适应了国家信息化发展的迫切需要,对计算机应用知识与技能的普及起到了有力的促进作用,成为了面向未来、面向新世纪培养人才、继续教育的一种有效途径。

参加NCRE的许多人都普遍感到这种考试与传统考试不同,除指定的教材外,缺少关于上机指导、笔试指导以及模拟试题方面的资料,因此,为配合社会各类人员参加考试,能顺利通过“全国计算机等级考试”,我们组织多年从事辅导计算机等级考试的专家在对近几年的考试深刻分析、研究基础上,并依据教育部考试中心最新考试大纲的要求,编写出这套指导应考者参加考试的备考辅导资料,本套丛书具有以下特点:

一、本套丛书自2000年在中国大地出版社出版以来,其后是不断修订再版,无论是内容还是题型,均以**教育部考试中心最新考试大纲**为纲,围绕**考生需求**为领,不断的作出修订和改进,力求把**韬略图书**做到最好。

二、在图书内容上,每本书分为两大部分:第一部分是**历届上机试题汇编**(其中包括了近年来上机考试真题以及详细的解析);第二部分是**历届笔试真题汇编**(其中包括了近年来笔试考试真题以及详细的解析),以供考生参考,达到事半功倍的效果。

三、本书采用小5号字紧缩式排版,每一页比同类其他书内容更充实、丰富,目的是让考生在同等硬件条件下汲取更多营养。

四、参与本书的编写者都为北京大学、清华大学等计算机专业人才,均是具有丰富教学和研究经验的专家、教授。另外,在此书的出版过程中,曾得到**全国计算机等级考试委员会顾问组组长罗晓沛教授**的悉心指导和热情支持,在此表示特别感谢。

五、本系列丛书和全国计算机等级考试(<http://www.ncre.cn>,该网站是隶属于教育部考试中心的官方网站,是全国计算机等级考试惟一权威信息发布网站)合作,只要您花5元钱就可以得到面值30元的上机考试卡。您可以凭借该卡登录全国计算机等级考试网,注册成为该网会员,学习全国计算机等级考试**网上课程**,该课程提供全真上机考试模拟环境,汇集正式考试的各种试题、答案及答题技巧,练习、自测模式任选,随机抽题,熟悉上机考试环境,轻松过级不再是梦。

六、凡购买本套丛书的读者,均可免费成为“韬略读者俱乐部”的会员。并享受购书带来的诸多实惠,欢迎读者积极参与。

七、由于本套丛书修订出版时间仓促,谬误之处在所难免,恳请广大读者能及时给予批评指正,以促进本套丛书质量的不断提高,谢谢!

全国计算机等级考试命题研究组

2006.北京



二级网络技术考试大纲

一、基本要求

1. 具有计算机软件及应用的基本知识。
2. 掌握操作系统的基本知识。
3. 掌握计算机网络的基本概念与基本工作原理。
4. 掌握 Internet 的基本应用知识。
5. 掌握组网、网络管理与网络安全等计算机网络应用的基本知识。
6. 了解网络技术的发展。
7. 掌握计算机操作并具有 C 语言编程(含上机调试)的能力。

二、考试内容

(一) 基础知识

1. 计算机系统组成。
2. 计算机软件的基础知识。
3. 多媒体的基本概念。
4. 计算机应用领域。

(二) 计算机网络基本概念

1. 计算机网络的定义与分类。
2. 数据通信技术基础。
3. 网络体系结构与协议的基本概念。
4. 广域网、局域网与城域网的分类、特点与典型系统。
5. 网络互联技术与互联设备。

(三) 局域网应用技术

1. 局域网分类与基本工作原理。
2. 高速局域网。
3. 局域网组网方法。
4. 结构化布线技术。

(四) 网络操作系统

1. 操作系统的基本功能。
2. 网络操作系统的基本功能。
3. 了解当前流行的网络操作系统的概况。

(五) Internet 基础

1. Internet 的基本结构与主要服务。
2. Internet 通信协议—TCP/IP。
3. Internet 接入方法。
4. 超文本、超媒体与 Web 浏览器。





(六) 网络安全技术

1. 信息安全的基本概念。
2. 网络管理的基本概念。
3. 网络安全策略。
4. 加密与认证技术。
5. 防火墙技术的基本概念。

(七) 网络应用: 电子商务与电子政务

1. 电子商务基本概念与系统结构。
2. 电子政务基本概念与系统结构。
3. 浏览器、电子邮件及 Web 服务器的安全特性。
4. Web 站点内容的策划与推广。
5. 使用 Internet 进行网上购物与访问政府网站。

(八) 网络技术发展

1. 网络应用技术的发展。
2. 宽带网络技术。
3. 网络新技术。

(九) 上机操作

1. 掌握计算机基本操作。
2. 熟练掌握 C 语言程序设计基本技术、编程和调试。
3. 掌握与考试内容相关的上机应用。

三、考试方式

(一) 笔试: 120 分钟, 满分 100 分

(二) 上机考试: 60 分钟, 满分 100 分





目 录

第1部分 历届上机试题汇编	1
第2部分 历届笔试真题汇编	72
2002年9月全国计算机等级考试三级网络技术笔试试卷	72
2002年9月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	78
2003年4月全国计算机等级考试三级网络技术笔试试卷	86
2003年4月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	91
2003年9月全国计算机等级考试三级网络技术笔试试卷	99
2003年9月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	104
2004年4月全国计算机等级考试三级网络技术笔试试卷	112
2004年4月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	117
2004年9月全国计算机等级考试三级网络技术笔试试卷	125
2004年9月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	130
2005年4月全国计算机等级考试三级网络技术笔试试卷	138
2005年4月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	143
2005年9月全国计算机等级考试三级网络技术笔试试卷	149
2005年9月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	154
2006年4月全国计算机等级考试三级网络技术笔试试卷	163
2006年4月全国计算机等级考试三级网络技术笔试试卷参考答案及解析	168



第 1 部分 历届上机试题汇编

1. 请编写一个函数 jsValue(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 out1.dat 中。部分程序已经给出。

例如:若输入 17,5,则应输出:19,23,29,31,37。

请勿改动主函数 main() 和输出数据函数 readwriteDat() 的内容。

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

main()
{
    int m,n,xx[1000];
    clrscr();
    printf(" \nPlease enter two integers:");
    scanf("%d,%d",&m,&n);
    jsValue (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("in1.dat","r");
    wf=fopen("out1.dat","w");
    for(i=0;i<10;i++)
    {
        fscanf(rf,"%d %d",&m,&n);
        jsValue (m,n,xx);
        for(m=0;m<n;m++)
            fprintf(wf,"%d",xx[m]);
        fprintf(wf,"\n");
    }
    fclose(rf);
    fclose(wf);
}
```





【答案】

```
void jsValue (int m,int k,int xx[])
{
    int data = m + 1;
    int half,n = 0,i;
    while(1)
    {
        half = data/2;
        for(i = 2;i <= half;i++)
            if(data%i == 0)
                break;
        if(i > half)
        {
            xx[n] = data;
            n++;
        }
        if(n >= k)
            break;
        data++;
    }
}
```

【解析】

考点:(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到它的一半之间所有的数,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,然后把 $jsValue()$ 函数的处理结果,通过 $readwriteDat()$ 函数写回文件 $out1.dat$ 中。

在 $jsValue()$ 函数中,使用一个 $while$ 循环查找 n 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 $i \leq half$),否则循环执行到 $i > half$ 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,又作为是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, $jsValue()$ 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, $jsValue()$ 函数结束。

2. 在文件 $in2.dat$ 中有200个正整数,且每个数均在1000至9999之间。函数 $ReadDat()$ 读取这200个数字存放到数组 aa 中。请编制函数 $jsSort()$,其函数的功能是:要求按每个数的后三位的大小进行升序排列,然后取出满足此条件的前10个数依次存入数组 bb 中,如果后三位的数值相等,则按原先的数值进行降序排列。最后调用函数 $WriteDat()$ 把结果 bb 输出到文件 $out2.dat$ 中。

例:处理前 6012 5099 9012 7025 8088

处理后 9012 6012 7025 8088 5099





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 >
int aa[ 200 ], bb[ 10 ];
void jsSort( )
{

}

void main( )
{

    ReadDat( );
    jsSort( );
    WriteDat( );

}

ReadDat( )
{

    FILE * in;
    int i;
    in = fopen( " in2. dat", " r" );
    for( i = 0; i < 200; i ++ )
        fscanf( in, "% d", &aa[ i ] );
    fclose( in );

}

WriteDat( )
{

    FILE * out;
    int i;
    clrscr( );
    out = fopen( " out2. dat", " w" );
    for( i = 0; i < 10; i ++ )
    {

        printf( "% d ", bb[ i ] );
        fprintf( out, "% d \n ", bb[ i ] );

    }

    fclose( out );

}
```

【答案】

```
void jsSort( )
{

    int i, j, data;
    for( i = 0; i < 199; i ++ )
        for( j = i + 1; j < 200; j ++ )

            if ( aa[ i ] % 1000 > aa[ j ] % 1000 )

}
```





```

    data = aa[i];
    aa[i] = aa[j];
    aa[j] = data;
}
else if( aa[i] % 1000 == aa[j] % 1000)
    if( aa[i] < aa[j])
    {
        data = aa[i];
        aa[i] = aa[j];
        aa[j] = data;
    }
}

for(i=0;i<10;i++)
    bb[i] = aa[i];
}

```

【解析】

考点:(1)数据的分解;
 (2)分解后的筛选;
 (3)筛选后的排序。

本题的解题思路是:对每个四位数的后三位数字进行比较排序,若后三位相等则按原始值进行排序。可以通过“%”来取得当前数的后三位数字,比较排序的算法是每次选定一个位置的元素和它后面的所有元素相比较,如果比后面的元素大则两者交换,若相等则按实际大小排序,比较一趟完成后,此位置上元素的后三位数字,都不大于其后面所有元素的后三位数字。比较一直进行到所有元素均被访问后结束,一共需要比较199趟,每趟需要比较 $199-j$ 次。

程序的流程是:首先调用了 ReadDat() 函数读入数据并且存放在数组 aa 中,然后调用 jsSort() 函数对数据进行操作,最后由 WriteDat() 函数将结果写回文件 out2. dat 中去。

在 jsSort() 函数中,前面两重 for 循环的功能是对数组 aa 中的元素进行排序,其排序的过程是根据每个元素的后三位数字大小进行升序排列。其中 $aa[i] \% 1000$ 操作是舍去千位上的数字,得到百位、十位和个位数字,然后通过下面 else if 语句和内嵌 if 语句进行判断,如果元素的后三位相等,则按实际大小进行降序排列。最后执行一个 for 循环,将排序好的前十位数字存入数组 bb 中,jsSort() 函数结束。

3. 已知数据文件 in3. 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:依次从数组 a 中取出一个四位数,如果该四位数连续大于该四位数以后的 5 个数且该数是奇数,则统计出满足此条件的个数 cnt 并把这些四位数按从小到大的顺序存入数组 b 中,最后调用写函数 writeDat() 把结果 cnt 以及数组 b 中符合条件的四位数输出到 out3. 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 MAX200
int a[ MAX ], b[ MAX ], cnt = 0;
void jsVal()
{
    ...
}

void readDat()
{
    int i;

```





```
FILE *fp;
fp = fopen("in3.dat", "r");
for(i = 0; i < MAX; i++) fscanf(fp, "%d", &a[i]);
fclose(fp);
}

void main()
{
    int i;
    readDat();
    jsVal();
    printf(" Satisfies the condition number: %d\n", cnt);
    for(i = 0; i < cnt; i++) printf("%d ", b[i]);
    printf("\n");
    writeDat();
}

writeDat()
{
    FILE *fp;
    int i;
    fp = fopen("out3.dat", "w");
    fprintf(fp, "%d\n", cnt);
    for(i = 0; i < cnt; i++) fprintf(fp, "%d\n", b[i]);
    fclose(fp);
}
```

【答案】

```
void jsVal()
{
    int i, j, flag = 0;
    int k;
    for(i = 0; i < MAX - 5; i++)
    {
        for(j = i + 1; j <= i + 5; j++)
        {
            if(a[i] > a[j]) flag = 1;
            else { flag = 0; break; }
            if(a[i] % 2 == 0)
                flag = 0;
        }

        if(flag == 1)
            { b[cnt] = a[i]; cnt++; }
    }

    for(i = 0; i < cnt - 1; i++)
        for(j = i + 1; j < cnt; j++)
            if(b[i] > b[j])
                { k = b[i]; b[i] = b[j]; b[j] = k; }
}
```



【解析】

考点:(1)按条件筛选;

(2)统计满足条件的个数;

(3)排序算法。

本题的解题思路是:对数组 a 中的元素逐个进行判断,若其能连续大于后面的 5 个数(此处可以通过设置一个标志变量来标识 $a[i]$ 能否连续大于其后面的 5 个数字),且为奇数时,把它存入 b 数组中,并对数组 b 进行排序。此排序算法为每次选定一个位置的元素和它后面的所有元素相比较,如果比后面的元素大,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小。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 $\text{cnt} - j$ 次。

程序的流程是:先调用 $\text{readDat}()$ 函数,读入数据到数组 a 中,然后调用 $\text{jsVal}()$ 函数,对数据进行处理再由 $\text{writeDat}()$ 函数写回文件 out3.dat 中去。

在 $\text{jsVal}()$ 函数中,通过双重 for 循环和内嵌 if 语句对数组 a 中的元素逐个判断,每当条件满足时就将标记变量 flag 设置为 1,否则将 flag 设置为 0。外层 for 循环结束的条件是 $i = \text{MAX} - 5$ (即一直比较到数组当中倒数第 6 个元素),在 for 循环中,当 $a[i]$ 连续大于其后面的 5 个元素时,标志变量 flag 被置为 1,否则标志 flag 被置为 0,并且跳出内层 for 循环。下面的 if 语句的作用是判断 $a[i]$ 是否为奇数,若是偶数则将 flag 置为 0。 for 循环结束后的 if 语句作用是,通过标志变量 flag 来判断 $a[i]$ 是否满足题目中的条件,即 $a[i]$ 即连续大于后面 5 个数字,又是奇数时,若满足则将 $a[i]$ 存入 b 数组中。最后对 b 数组中的元素进行升序排列,排序完成之后,函数结束。

4. 编写一个函数 $\text{findStr}()$,该函数统计一个长度为 2 的子字符串在另一个字符串中出现的次数。例如,假定输入的字符串为 "asdasasdfgasdaszx67asdmklo",子字符串为 "as",函数返回值是 6。

函数 $\text{ReadWrite}()$ 实现从文件 in4.dat 中读取两个字符串,并调用函数 $\text{findStr}()$,最后把结果输出到文件 out4.dat 中。

注意:部分程序已经给出。

请勿改动主函数 $\text{main}()$ 和其他函数中的任何内容,仅在函数 $\text{findStr}()$ 的花括号中填入你编写的若干语句。

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int findStr(char *str, char *substr)
```

```
{
```

```
}
```

```
main()
```

```
{
```

```
    char str[81], substr[3];
```

```
    int n;
```

```
    clrscr();
```

```
    printf("enter the original string:");
```

```
    gets(str);
```

```
    printf("enter an substring:");
```

```
    gets(substr);
```

```
    puts(str);
```

```
    puts(substr);
```

```
    n = findStr(str, substr);
```

```
    printf("n = %d\n", n);
```

```
    ReadWrite();
```

```
ReadWrite()
```

```
}
```





```

char str[81], substr[3], ch;
int n, len, i = 0;
FILE * rf, * wf;
rf = fopen("in4.dat", "r");
wf = fopen("out4.dat", "w");
while(i < 5)
{
    fgets(str, 80, rf);
    fgets(substr, 10, rf);
    len = strlen(substr) - 1;
    ch = substr[len];
    if(ch == '\n') substr[len] = 0;
    n = findStr(str, substr);
    fprintf(wf, "%d\n", n);
    i++;
}
fclose(rf);
fclose(wf);

```

【答案】

```

int findStr(char * str, char * substr)
{
    int n;
    char * p, * r;
    n = 0;
    while(* str)
    {
        p = str;
        r = substr;
        while(* r)
            if(* r == * p)
            {
                r++;
                p++;
            }
        else
            break;
        if(* r == '\0')
            n++;
        str++;
    }
    return n;
}

```

【解析】

考点:(1)在给定的字符串中查找指定字符;

(2)统计查找后满足条件的个数。

本题的解题思路是:对原字符串中的字符进行扫描,若原字符串中含有子字符串,即原字符串的其中一部分与子字符串





完全相同时,统计出个数。判断原字符串是否含有子字符串,可以将原字符串中的元素逐个与子字符串相比较,当两字符串中各个元素都相同且子字符串结束时,则证明原字符串包含一个子字符串。在统计个数的时候需要设置一个记录变量,每当原字符串包含子字符串一次的时候,该记录变量自动加1。

程序的流程是:首先通过键盘接收两个字符串,其中一个作为原字符串,另一个作为子字符串,然后调用 findStr() 函数对两字符串进行比较,处理后的结果由 ReadWrite() 函数写回文件 out4. dat 中。

在 findStr() 函数中,当原字符串不为空,即 *str 不为空时进入到外层 while 循环中,此时原字符串指针 str 和子字符串指针 substr 都指向其字符串内的第一个元素,将两字符串的指针分别赋给 p 和 r,然后再对 p 和 r 进行比较。这样做的好处是,当 p 与 r 改变的时候 str 和 substr 不会受到影响。当子字符串也不为空时即 *r 不为空时,进入内嵌的 while 循环,该 while 循环的功能是,将原字符串与子字符串逐个元素进行比较看是否相等,结束该层 while 循环的情况有两种:(1) 比较完毕,即原字符串中包含该子字符串,此时子字符串的指针指向串尾(为“\0”);(2) 未比较完毕,此时子字符串的指针不指向串尾。下面的 if 语句的功能是,通过判断子字符串的指针是否指向串尾,进而来判断内层 while 循环结束的原因,若是比较完成,即 *r == '\0',则把记录个数的变量加1,并且原字符串指针也指向下一个字符,为重新进入外层 while 循环作准备。

5. 编写函数 jsValue(), 它的功能是:求 Fibonacci 数列中大于 t 的最小一个数,结果由函数返回。其中 Fibonacci 数列 F(n) 的定义为:

$$F(0) = 0, F(1) = 1$$

$$F(n) = F(n-1) + F(n-2)$$

最后调用函数 writeDat() 读取 50 个数据 t, 分别得出结果且把结果输出到文件 out5. dat 中。

例如:当 t = 1000 时,函数值为:1597。

部分程序已经给出。

请勿改动主函数 main() 和写函数 writeDat() 的内容。

```
#include <stdio.h>
```

```
int jsValue(int t)
```

```
{
```

```
{
```

```
main()
```

```
{
```

```
int n;
```

```
n = 1000;
```

```
printf("n = %d, f = %d\n", n, jsValue(n));
```

```
writeDat();
```

```
{
```

```
writeDat()
```

```
{
```

```
FILE *in, *out;
```

```
int i, n, s;
```

```
in = fopen("in5. dat", "r");
```

```
out = fopen("out5. dat", "w");
```

```
for(i = 0; i < 50; i++)
```

```
{
```

```
fscanf(in, "%d", &n);
```

```
s = jsValue(n);
```

```
printf("%d\n", s);
```

```
fprintf(out, "%d\n", s);
```

```
}
```

```
fclose(in);
```





```
fclose(out);
```

【答案】

```
int jsValue(int t)
{
    int f1=0,f2=1,fn;
    fn=f1+f2;
    while(fn<=t)
    {
        f1=f2;
        f2=fn;
        fn=f1+f2;
    }
    return fn;
}
```

【解析】

考点:(1)求解 Fibonacci 数列的第 n 项的值;

(2)查找满足条件的 Fibonacci 数列的第 n 项值。

本题的解题思路是:从第一项开始,逐个求出 Fibonacci 数列的每一项的值并且与给定的数据相比较,若找到第一个大于给定的数据则将其返回。

程序的流程是:先调用 jsValue(n)函数处理数据,由 writeDat()函数将数据写回文件 out5.dat 中。

在 jsValue()函数中,根据题目的条件,Fibonacci 数列中每一项的值均为前两项之和,将 Fibonacci 数列的每一项都初始化,然后通过 while 循环的条件进行判断,当 $fn \leq t$ 时,即第 n 项的值小于 1000,进入 while 循环体,计算 fn 的值,每计算一个 fn 的值就与 t 相比较一次,直到 $fn > t$ 即找到大于 t 的最小的一个数,循环结束,该层 while 循环的功能就是查找比 t 大的 fn 的值,while 循环结束后,利用 return 语句返回 fn 的值,函数结束。

6. 请编写函数 countValue(),它的功能是:求 n 以内(不包括 n)同时能被 3 与 7 整除的所有自然数之和的平方根 s ,并作为函数值返回,最后结果 s 输出到文件 out6.dat 中。

例如若 n 为 1000 时,函数值应为: $s = 153.909064$ 。

部分程序已经给出。

请勿改动主函数 main()和输入输出数据函数 progReadWrite()的内容。

```
#include <conio.h>
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```
double countValue(int n)
```

```
{
```

```
}
```

```
main()
```

```
{
```

```
clrscr();
```

```
printf("the root during the total of the numbers = %f\n",countValue(1000));
```

```
progReadWrite();
```

```
}
```

```
progReadWrite()
```

```
{
```

```
FILE *fp,*wf;
```





```
int i, n;
float s;
fp = fopen("in6.dat", "r");
if (fp == NULL)
{
    printf("the datafile in6.dat is not exist!");
    return;
}

wf = fopen("out6.dat", "w");
for(i = 0; i < 10; i++)
{
    fscanf(fp, "%d\n", &n);
    s = countValue(n);
    fprintf(wf, "%f\n", s);
}

fclose(fp);
fclose(wf);
```

【答案】

```
double countValue(int n)
{
    double xy = 0.0;
    int i;
    for(i = 1; i < n; i++)
        if(i%3 == 0 && i%7 == 0)
            xy += i;
    xy = sqrt((double)xy);
    return xy;
}
```

【解析】

考点:(1)按条件查找数字;

(2)对查找结果进行再加工。

本题的解题思路是:先对1到n之间的所有数字进行筛选,若能被3和7整除,则将其累加到一个临时变量中,再通过sqrt()函数求出其平方根。判断是否能整除,可以通过判断该数字模3和模7结果来实现,若为0则能整除。为了保证结果的精确度,在调用sqrt()函数的时候,需要将sqrt()函数的参数强制转换成双精度类型。此部分考生应该仔细分析。

程序的流程是:首先调用countValue()函数,并传递给countValue()函数一个参数,然后由progReadWrite()函数将结果写回文件out6.dat中。

在countValue()函数中,首先要求传递一个整型变量作为参数,其作用是在该参数范围以内进行查找符合条件的整数,然后通过一个for循环,依次取出比n小的各个自然数进行判断,for循环的结束条件是i=n,即所有的数字都查找完成后结束,if语句的功能是判断数字是否能被3和7整除,如果if语句的条件成立,则将该数累加到临时变量xy中,最后通过求平方根函数sqrt()求得xy的平方根,并存入到xy中。在求平方根时,为保证精度,先将xy强制转化成双精度实型数据,注意强制类型转换的括号不可以省略。最后将xy返回,函数结束。

7. 下列程序的功能是:在三位整数(100至999)中寻找符合条件的整数并依次从小到大存入数组中;该整数既是完全平方数,又有两个数字相同,例如144、676等。

请编制函数jsValue()实现此功能,满足该条件的整数的个数是通过所编制的函数返回。

最后调用函数writeDat()把结果输出到文件out7.dat中。