



NCRE

全国计算机等级考试真题(笔试+上机)详解与样题精选

全国计算机等级考试 **真题**(上机专辑)

详解与样题精选 (二级 C 语言程序设计)

钱 隆 薛慧明 申家军 钱 艳 编著

研究真题是考试过关的捷径

实战样题是加分致胜的法宝

“巧记、巧练、巧过关”

根据最新大纲编写



清华大学出版社

全国计算机等级考试真题 **上机** 详解与样题精选
专 辑

(二级 C 语言程序设计)

钱 隆 薛慧明 申家军 钱 艳 编著

清华大学出版社

北 京

内 容 简 介

本书针对教育部考试中心全国计算机等级考试二级 C 语言上机试题进行了全面分析,在此基础上进行了详细分类与透彻解析,并对题中涉及到的众多知识点做了相应的理论链接,以达到“以点带面,以面带全”的效果。目的是使考生能够抓住考试的重点与难点,掌握考试中经常出现的题型和每种题型的解法,同时也使考生熟悉专家们的出题思路、命题规律,从而提高应试复习的效率和命中率。本书主要内容包括:第 1 章为程序填空题,这是新增的题型,本书对此进行了重点讲解;第 2 章对程序修改试题中的错误进行分类介绍,重点讲解排错方法;第 3 章程序设计试题,根据常见求解问题分类,重点讲解解题思路和方法;第 4 章提供了 20 套上机样题和参考答案,样题的命题形式、考点分布、难易程度等均与等级考试的真实试卷相当,便于考生考前实战冲刺,体验真实训练。

本书针对性强,特别适合作为参加全国二级 C 语言上机考试的备考用书,也可作为各类大中专院校学生学习 C 语言程序设计的参考书使用。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用清华大学核研院专有核径迹膜防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

全国计算机等级考试真题(上机专辑)详解与样题精选(二级 C 语言程序设计)/钱隆,薛慧明,申家军,钱艳编著. —北京:清华大学出版社,2005.3

ISBN 7-302-10516-2

I.全… II.①钱… ②薛… ③申… ④钱… 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(2005)第 012382 号

出 版 者:清华大学出版社

地 址:北京清华大学学研大厦

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

客户服务:010-62776969

组稿编辑:章忆文

文稿编辑:李春明

封面设计:陈刘源

印 装 者:三河市春园印刷有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:21.5 字数:511 千字

版 次:2005 年 3 月第 1 版 2005 年 3 月第 1 次印刷

书 号:ISBN 7-302-10516-2/TP·7140

印 数:1~5000

定 价:32.00 元(含 1 张光盘)

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

丛 书 序

计算机知识是当代人类文化的重要组成部分, 计算机应用能力是跨世纪人才不可缺少的素质。有鉴于社会对计算机技术的客观需求, 教育部考试中心推出了全国计算机等级考试, 其目的是以考促学, 全面提高社会计算机应用水平, 并为用人单位提供统一、客观、公正的评价标准。全国计算机等级考试自 1994 年开考以来, 参考人数逐年递增, 现已成为国内影响最大、参加人数最多的计算机类水平考试。

为适应当前信息技术的飞速发展, 国家教育部考试中心对全国计算机等级考试的考试科目及内容进行了重大调整, 对考试大纲进行了全面修订。为了更好地服务于考生, 引导考生尽快掌握计算机的先进技术, 并顺利通过计算机等级考试, 我们在深入剖析最新考试大纲和历年考题的基础上, 特别编写了这套“全国计算机等级考试真题(笔试+上机)详解与样题精选”丛书。

已出书目

1. 《全国计算机等级考试真题(上机考试)详解与样题精选(一级 MS Office)》
2. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(二级公共基础知识+C 语言)》
3. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(二级公共基础知识+Visual Basic)》
4. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(二级公共基础知识+Visual FoxPro)》
5. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(三级 PC 技术)》
6. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(三级网络技术)》
7. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(三级信息管理技术)》
8. 《全国计算机等级考试真题(笔试+上机)详解与样题精选(三级数据库技术)》
9. 《全国计算机等级考试真题(上机专辑)详解与样题精选(二级 C 语言程序设计)》
10. 《全国计算机等级考试真题(上机专辑)详解与样题精选(三级 C 语言程序设计)》

丛书特色

- 以考题带动考点的学习与复习。与其他同类图书不同之处是: 本丛书的结构不是传统的“考点→例题→习题”, 而是在听取大量专家及考生意见的基础上, 采用“真题→分析→考点”的方式。实践证明, 这种“将考点融入考题、以考

题学习考点”的方式应试针对性极强，特别适合考生在短时间内突破过关。

- ❑ 真题分类解析。丛书将近几年考题及大纲样题进行深度剖析，然后按教育部考试中心指定教材的章节分类编排，从而利于考生分类复习，专项攻克，同时也便于考生更好地理解 and 掌握等级考试的内容、范围及难度，便于考生把握命题规律，快速提升应试能力。
- ❑ 题型分析透彻。将历年考题及典型例题进行分类解析，覆盖全部考试要点，讲解深入、全面，能让读者达到触类旁通、举一反三之功效。
- ❑ 全真模拟实战。丛书提供数套全真样题，样题是由经验丰富的等级考试辅导老师经过精心设计和锤炼的。全面模拟考试真题，预测考点，应试导向准确。
- ❑ 书盘结合。每本书都配有一套上机考试模拟盘，其考试界面、题型和考试环境与真实考场完全相同，便于考生熟悉上机考试。

📖 读者对象

本套丛书特别适合参加全国计算机等级考试的考生使用，也可作为各类全国计算机等级考试培训班的教材，以及大、中专院校师生的教学参考书。

丛书编委会

前 言

随着科学技术的不断发展,计算机在国民经济和社会生活各个领域都发挥着越来越重要的作用。同时,学习计算机、普及计算机基础知识也已经成为人们自觉自愿的行为。为了更好地促进人们对计算机的学习使用,教育部考试中心推出了全国计算机等级考试制度。近年来,教育部考试中心对计算机等级考试科目及内容做出重大调整,在加大试题库容量的同时,将上机题库有关内容在考前向社会公开。

全国计算机等级考试内容分笔试和上机考试两个部分。其中,上机考试部分因为很多考生平时缺乏训练,所以成绩不是很理想。针对这个问题,我们编写了这本辅导教材,以帮助考生迅速提高实践能力,掌握上机题型的特点和答题技巧,从而顺利通过考试。

本书针对教育部考试中心全国计算机等级考试二级 C 语言上机试题进行了认真的分析,并在此基础上进行了详细分类与透彻解析,并对题中涉及到的众多知识点做了相应的理论链接,以达到“以点带面,以面带全”的效果。目的是使考生能够抓住考试的重点与难点,掌握考试中经常出现的题型和每种题型的解法,同时也使考生熟悉专家们的出题思路、命题规律,从而提高应试复习的效率和命中率。

全书分为 4 章。第 1 章为程序填空题,这是新增的题型,本书对此进行了重点讲解;第 2 章对程序修改试题中的错误进行分类介绍,重点讲解排错方法;第 3 章程序设计试题,根据常见求解问题分类,重点讲解解题思路和方法;第 4 章提供了 20 套上机样题和参考答案,样题的命题形式、考点分布、难易程度等均与等级考试的真实试卷相当,便于考生考前实战冲刺,体验真实训练。

本书针对性强,特别适合作为参加全国二级 C 语言上机考试的备考用书,也可作为各类大中专院校学生学习 C 语言程序设计的参考书使用。

本书配有上机盘,目的是为了广大等级考试考生提供一个实战训练的上机环境。整个模拟环境与实际上机考试相同。本书配书盘的安装密码是 VCBSJZJ2c。

本书由钱隆主编与统稿,薛慧明、申家军和钱艳参与了编写工作,韦华云、唐德勇参与了校对和排版工作。本书配书光盘由刘菁、赵传申、杨明、杨萍等完成。另外,参与资料收集及整理工作的还有丁为民、许明亚、陈智、付淑慧、吴婷、谢波等,在此一并致谢!

需要指出的是,书中很多题目的解法并不是惟一的,读者完全可以举一反三,自己总结出更多更好的算法来。

由于时间和作者水平有限,书中难免有一些疏漏之处,敬请广大读者批评指正。

编者

2005 年 2 月

目 录

第 1 章 程序填空题.....	1
考点 1: 字符串操作 ★★★★★.....	1
考点 2: 公式计算 ★★★★★.....	12
考点 3: 算法实现 ★★★★★.....	19
考点 4: 文件操作 ★★★.....	26
考点 5: 数组操作 ★★★.....	31
考点 6: 结构体操作 ★★★.....	33
考点 7: 字母转换 ★★.....	39
第 2 章 程序修改题.....	41
考点 1: 字母大小写转换 ★★★.....	41
考点 2: 填空 ★★★.....	45
考点 3: 标点符号、字符或运算符 ★★★★★.....	51
考点 4: 丢括号或语句结束符 ★★★★★.....	75
考点 5: 变量初始化 ★★★★★.....	82
考点 6: 数据类型匹配 ★★★★★.....	89
考点 7: 限定条件 ★★★★★.....	99
考点 8: 数组元素引用 ★★★.....	102
考点 9: 指针使用 ★★★★★.....	103
考点 10: 逻辑关系 ★★★★★.....	114
考点 11: 算法实现 ★★★★★.....	127
考点 12: 函数类型说明 ★★★.....	133
考点 13: 关键字 ★★★.....	137
第 3 章 程序设计题.....	140
考点 1: 统计字符串中单词或字母的个数 ★★.....	140
考点 2: 大小写转换 ★★.....	143
考点 3: 删除或修改指定元素 ★★★★★.....	146
考点 4: 字符串连接 ★★.....	160
考点 5: 字符串比较 ★★.....	164
考点 6: 字符串化为数字 ★★.....	168
考点 7: 指定下标或 ASCII 码的字符操作 ★★★★★.....	170
考点 8: 素数的计算 ★★★.....	177
考点 9: 方程求根 ★★.....	181
考点 10: 级数与多项式 ★★★★★.....	183
考点 11: 数字重组 ★★★★★.....	195

考点 12: 一维数组运算 ★★★★★	205
考点 13: 二维数组和矩阵运算 ★★★★★	213
考点 14: 排序与查找 ★★★★★	227
考点 15: 结构体数据处理 ★★★★★	234
考点 16: 链表结构处理 ★★★	244
第 4 章 上机样题精选	249
上机样题一	249
上机样题二	251
上机样题三	254
上机样题四	258
上机样题五	260
上机样题六	264
上机样题七	266
上机样题八	269
上机样题九	271
上机样题十	274
上机样题十一	277
上机样题十二	280
上机样题十三	282
上机样题十四	286
上机样题十五	289
上机样题十六	292
上机样题十七	295
上机样题十八	298
上机样题十九	300
上机样题二十	303
附录 A 上机样题精选参考答案与分析	307
上机样题一参考答案与分析	307
上机样题二参考答案与分析	308
上机样题三参考答案与分析	309
上机样题四参考答案与分析	310
上机样题五参考答案与分析	311
上机样题六参考答案与分析	312
上机样题七参考答案与分析	313
上机样题八参考答案与分析	314
上机样题九参考答案与分析	315
上机样题十参考答案与分析	316
上机样题十一参考答案与分析	317

上机样题十二参考答案与分析	318
上机样题十三参考答案与分析	319
上机样题十四参考答案与分析	320
上机样题十五参考答案与分析	321
上机样题十六参考答案与分析	322
上机样题十七参考答案与分析	323
上机样题十八参考答案与分析	324
上机样题十九参考答案与分析	325
上机样题二十参考答案与分析	326
附录 B 二级 C 语言考试大纲 (2004 年版)	328
公共基础知识	328
基本要求	328
考试内容	328
考试方式	329
C 语言程序设计	329
基本要求	329
考试内容	329
考试方式	331
参考文献	332

第 1 章 程序填空题

考点 1: 字符串操作 ★★★★★

考点点拨: 考查 C 语言中字符串操作的方法。

【试题 1】

给定程序如下, 要求使未在字符串 s 中出现, 而在字符串 t 中出现的字符, 形成一个新的字符串放在 u 中, u 中字符按原字符串中字符顺序排列, 但去掉重复字符。

例如: 当 s="abcdef", t="acbdghfh" 时, u 中的字符为: "gh"。

请在程序的下划线处填入正确的内容并把下划线删除, 使程序得出正确的结果。

 **注意:** 源程序如下。不得增行或删行, 也不得更改程序的结构!

【试题程序】

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

```
#include <string.h>
```

```
void fun (char *s, char *t, char *u)
```

```
{    int    i, j, sl, tl, k, ul=0;
```

```
    sl = strlen(s);    tl = strlen(t);
```

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

```
    {    for (j=0; j<sl; j++)
```

```
        if (t[i] == s[j]) break;
```

```
    if (j>=sl)
```

```
    {    for (k=0; k<ul; k++)
```

```
        if (t[i] == u[k]) 1 ;
```

```
    if (k>=ul)
```

```
        u[ul++] = 2 ;
```

```
    }
```

```
}
```

```
    u[ul] = 3 ;
```

```
}
```

```

main()
{   char    s[100], t[100], u[100];
    printf("\nPlease enter string s:"); scanf("%s", s);
    printf("\nPlease enter string t:"); scanf("%s", t);
    fun(s, t, u);
    printf("The result is: %s\n", u);
}

```

答案:

第1空填“break;”。

第2空填“t[i]”。

第3空填“\0”。

分析:

题中第1空“break;”为跳出循环的语句。第2空“u[ul++] = t[i];”是将t[i]赋给u[ul], 然后ul加1。第3空填“u[ul]=\0;”, 字符串的最后一个字符应该是'\0'。

【试题2】

给定程序如下, 要求使在字符串 s 中出现、而未在字符串 t 中出现的字符形成一个新的字符串放在 u 中, u中字符按原字符串中字符顺序排列, 不去掉重复字符。

例如: 当s="acbdghfh", t="abcdef"时, u中的字符为"ghh"。

请在程序的下划线处填入正确的内容并把下划线删除, 使程序得出正确的结果。

 **注意:** 源程序如下。不得增行或删行, 也不得更改程序的结构!

【试题程序】

```

#include <stdio.h>
#include <string.h>

void fun (char *s, char *t, char *u)
{   int    i, j, sl, tl;
    sl = strlen(s);   tl = strlen(t);
    for (i=0; i<sl; i++)
    {   for (j=0; j<tl; j++)
        {   if (s[i] == t[j])   1 ;
            if (j>=tl)
                2 = s[i];
        }
        3 = '\0';
    }
}

```

```

main()
{
    char    s[100], t[100], u[100];
    printf("\nPlease enter string s:"); scanf("%s", s);
    printf("\nPlease enter string t:"); scanf("%s", t);
    fun(s, t, u);
    printf("the result is: %s\n", u);
}

```

答案:

第1空填“break;”。

第2空填“*u++”。

第3空填“*u”。

分析:

第1空“break;”为跳出循环的语句。第2空填“*u++= s[i];”是将s[i]赋给*u, 然后u加1。第3空填“*u='\0';”, 字符串的最后一个字符应该是'\0’。

【试题3】

给定程序如下, 要求使在字符串s中下标为奇数位置上的字符, 紧随其后重复出现一次, 放在一个新串t中, t中字符按原字符串中字符出现的逆序排列(注意0为偶数)。

例如: 当s中的字符串为“abcdef”时, 则t中的字符串应为“ffddbb”。

请在程序的下划线处填入正确的内容并把下划线删除, 使程序得出正确的结果。

 注意: 源程序如下。不得增行或删行, 也不得更改程序的结构!

【试题程序】

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

```
#include <string.h>
```

```

void fun (char  *s, char  *t)
{
    int  i, j, sl;
    sl = strlen(s);
    if(sl%2) sl-=2; 1 sl--;
    for (i=sl, j=2; i>=0; i-=2)
    {
        t[2*j] = s[i];
        3 = s[i];
        j++;
    }
    t[2*j]='\0';
}

```

```

main()
{  char s[100], t[100];
    printf("\nPlease enter string s:"); scanf("%s", s);
    fun(s, t);
    printf("The result is: %s\n", t);
}

```

答案:

第1个空填“else”。

第2个空填“0”。

第3个空填“ t[2*j+1]”。

分析:

第1个空所在语句“if(s1%2) s1-=2; else s1--;”, 当s1为奇数时, s1-=2; 当s1为偶数时, s1--。第2个空填“0”是用于初始化循环变量。由题意奇数下标的字符连续写两次, 故在第3个空填“t[2*j+1]”。

【试题4】

给定程序如下, 要求使既在字符串s中出现、又在字符串t中出现的字符形成一个新的字符串放在u中, u中字符按原字符串中字符顺序排列, 但去掉重复字符。

例如, 当s="abcdefg", t="ackblm"时, u中的字符串为"acb"。

请在程序的下划线处填入正确的内容并把下划线删除, 使程序得出正确的结果。

💡 注意: 源程序如下。不得增行或删行, 也不得更改程序的结构!

【试题程序】

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

```
#include <string.h>
```

```

void fun (char *s, char *t, char *u)
{  int  i, j, sl, tl, k, ul=0;
    sl = strlen(s);    tl = strlen(t);
    for (i=0; i<sl; i++)
    {  for (j=0; j<tl; j++)
        if (s[i] == t[j]) break;
        if(j<tl)
        {  for (k=0; k<ul; k++)
            if (s[i] 1 u[k]) break;
            if (k==ul)
                2 = s[i];
        }
    }
}

```

```

    }
}
u[ul] = 3;
}

main()
{
    char s[100], t[100], u[100];
    printf("\nPlease enter string s:"); scanf("%s", s);
    printf("\nPlease enter string t:"); scanf("%s", t);
    fun(s, t, u);
    printf("The result is: %s\n", u);
}

```

答案:

第1个空填“==”。

第2个空填“u[ul++]”。

第3个空填“\0”。

分析:

“==”为比较运算符。“if (s[i] == u[k]) break;”语句是指当s[i]==u[k], 即u[k]中的字符已经在字符指针s做指向的字符数组中存在时, 循环就结束。所以第1个空填“==”。第2个空填“u[ul++] = s[i];”是将s[i]赋给u[ul], 然后ul加1。第3个空“u[ul]=\0;”, 字符串应该以\0'结束。

理论链接:

串是用一维字符数组来实现的, 而描述串的字符型指针是基于又高于一维字符数组的。

例如:

```

char string[100];    /* 定义串 */
char *s;             /* 定义指针变量 */
s = string;          /* 赋值 s */

```

(1)s 既可以是整个串的地址, 也可以是该串中某一字符的地址, 而*s 只能表示某一相应字符的值。

(2)s 是地址变量, 所以可以进行加减等地址运算, 其结果是串中某字符的首地址; 而string 是地址常量, 故不能进行地址性的加减运算。例如, s++是合法的, 而 string++就不合法了。

【试题5】

给定程序如下, 要求从字符串s尾部开始, 按逆序把相邻的两个字符交换位置, 并依次把每个字符紧随其后重复出现一次, 放在一个新串t中。

例如: 当s中的字符串为"abcde"时, 则t中的字符串应为"ddeebbccaa"。

请在程序的下划线处填入正确的内容并把下划线删除, 使程序得出正确的结果。

 注意: 源程序如下。不得增行或删行, 也不得更改程序的结构!

【试题程序】

```

#include <stdio.h>
#include <string.h>

void fun (char *s, char *t)
{
    int i, j, sl;
    sl = 1;
    for (i=sl-1, j=0; i>=0; i-=2)
    {
        if (i-1 >= 0) t[j++] = s[i-1];
        if (i-1 >= 0) t[j++] = s[i-1];
        t[j++] = s[i]; t[j++] = s[i];
    }
    2;
}

main()
{
    char s[100], t[100];
    printf("\nPlease enter string s:");
    scanf("%s", s);
    fun(3);
    printf("The result is: %s\n", t);
}

```

答案:

第1个空填“strlen(s)”。

第2个空填“t[j]='\0'”或者“t[j]=0”。

第3个空填“s,t”。

分析:

第1个空填“strlen(s)”。strlen()是测试字符串长度的函数,函数的值为字符串的实际长度,不包括'\0'在内。

第2个空填“t[j]='\0'”或者“t[j]=0”。字符串是以'\0'作为结束标志的,其ASCII的值是0。

第3个空填“s,t”。“fun(s,t);”用于调用fun函数。

【试题6】

给定程序如下,要求将仅在字符串s中出现而不在字符串t中出现的字符,和仅在字符串t中出现而不在字符串s中出现的字符,构成一个新字符串并放在u中,u中的字符按原字符串中字符顺序排列,不去掉重复字符。

例如:当s="aabcde", t="bdfgg"时, u中的字符串为"aacefgg"。

请在程序的下划线处填入正确的内容并把下划线删除，使程序得出正确的结果。



注意：源程序如下。不得增行或删行，也不得更改程序的结构！

【试题程序】

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

void fun (char *s, char *t, char *u)
{   int i, j, sl, tl;
    sl = strlen(s);    tl = strlen(t);
    for (i=0; i<sl; i++)
    {   for (j=0; j<tl; j++)
        if (s[i] == t[j]) break;
        if (j >= tl)
            ___1___ = s[i];
    }
    for (i=0; i<tl; i++)
    {   for (j=0; j<sl; j++)
        if (t[i] == s[j]) break;
        if (j ___2___ sl)
            *u++ = t[i];
    }
    ___3___ = '\0';
}

main()
{   char s[100], t[100], u[100];
    printf("\nPlease enter string s:"); scanf("%s", s);
    printf("\nPlease enter string t:"); scanf("%s", t);
    fun(s, t, u);
    printf("The result is: %s\n", u);
}
```

答案：

第1个空填“*u++”。

第2个空填“>=”。

第3个空填“*u”。

分析：

本题的考核点是字符指针和判断语句。

第1个空填“*u++”，“if (j >= tl)*u++ = s[i];”是和上面循环语句“for (j=0; j<tl; j++) if (s[i]

== t[j]) break;”结合起来的。如果循环一直进行下去,则最后j和t1是相等的。同理,第2个空填">="。

第3空个填"*u"。字符数组以'\0'标志结束。

理论链接:

C语言中有两个很有用的运算符,通常在其他计算机语言中是找不到它们的——自增和自减运算符,++和--。运算符“++”是操作数加1,而“--”是操作数减1,即 $x = x + 1$ 同 $++x$, $x = x - 1$ 同 $--x$ 。

自增和自减运算符可用在操作数之前,也可放在其后,例如 $x = x + 1$ 可写成 $++x$ 或 $x++$,但在表达式中这两种用法是有区别的。自增或自减运算符在操作数之前,C语言在引用操作数之前就先执行加1或减1操作;运算符在操作数之后,C语言就先引用操作数的值,而后再进行加1或减1操作。

例如,“ $x = 1; y = ++x;$ ”的结果为 $y = 2$ 。如果程序改为“ $x = 1; y = x++;$ ”则 $y = 1$ 。在这两种情况下,x都等于2,但区别在于设置的时刻,这种对自增和自减发生时刻的控制是非常有用的。

【试题7】

给定程序如下,要求将在字符串s中出现、而未在字符串t中出现的字符,构成一个新的字符串并放在u中,u中字符按原字符串中字符顺序的逆序排列,不去掉重复字符。

例如,当 $s = \text{"abcde"}$, $t = \text{"bdfg"}$ 时,u中的字符是:“ecaa”。

请在程序的下划线处填入正确的内容并把下划线删除,使程序得出正确的结果。

 注意:源程序如下。不得增行或删行,也不得更改程序的结构!

【试题程序】

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

void fun (char *s, char *t, char *u)
{   int    i, j, sl, tl, ul;
    char   r, *up=u;
    sl = strlen(s);    tl = strlen(t);
    for (i=0; i<sl; i++)
    {   for (j=0; j<__1__; j++)
        if (s[i] == t[j]) break ;
        if(j __2__ tl)
            *u++ = s[i];
    }
    *u = '\0';
    ul = strlen(up);
```