

希赛网, 专注于软考、PMP、通信考试的专业 IT 知识库和在线教育平台。希赛网在线题库, 提供历年考试真题、模拟试题、章节练习、知识点练习、错题本练习等在线做题服务, 更有能力评估报告, 让你告别盲目做题, 针对性地攻破自己的薄弱点, 更高效的备考。

希赛网官网: <http://www.educity.cn/>

希赛网软件水平考试网: <http://www.educity.cn/rk/>

希赛网在线题库: <http://www.educity.cn/tiku/>

2010 上半年程序员案例分析真题答案与解析: <http://www.educity.cn/tiku/tp1382.html>

2010 年上半年程序员考试下午真题 (参考答案)

- (共 15 分)

阅读以下说明和流程图, 将应填入 (n) 处的字句写在答题纸的对应栏内。

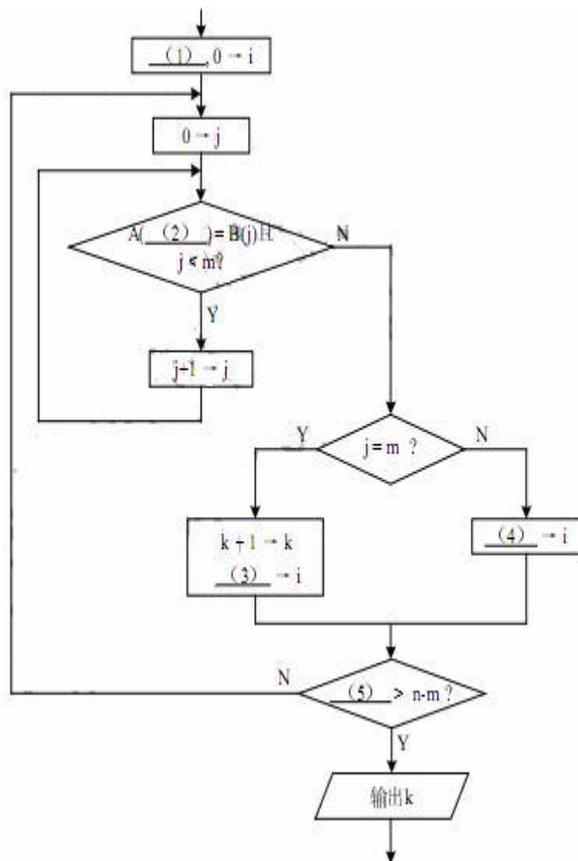
【说明】

下面的流程图旨在统计指定关键词在某一篇文章中出现的次数。

设这篇文章由字符 $A(0), \dots, A(n-1)$ 依次组成, 指定关键词由字符 $B(0), \dots, B(m-1)$ 依次组成, 其中 $n > m \geq 1$ 。注意, 关键词的各次出现不允许有交叉重叠。例如, 在“aaaa”中只出现两次“aa”。

该流程图采用的算法是: 在字符串 A 中, 从左到右寻找与字符串 B 相匹配的并且没有交叉重叠的所有子串。流程图中, i 为字符串 A 中当前正在进行比较的动态子串首字符的下标, j 为字符串 B 的下标, k 为指定关键词出现的次数。

【流程图】



• (共 15 分)

阅读以下问题说明、C 程序和函数，将解答填入答题纸的对应栏内。

【问题 1】

分析下面的 C 程序，指出错误代码（或运行异常代码）所在的行号。

【C 程序】

行号	代码
1	#include <stdio.h>
2	#include <string.h>
3	int main()
4	{ char *p, arrChar[100] = "testing";
5	arrChar = "test";
6	p = "testing";
7	*p = '0';
8	p = arrChar + 1;
9	printf("%d\t%d\n", sizeof(arrChar), strlen(arrChar));
10	printf("%d\t%d\n", sizeof(char *), strlen(p));
11	return 0;
12	}

【问题 2】

函数 `inputArr(int a[], int n)` 的功能是输入一组整数 (输入 0 或输入的整数个数达到 `n` 时结束) 存入数组 `a`, 并返回实际输入的整数个数。函数 `inputArr` 可以成功编译。但测试函数调用 `inputArr` 后, 发现运行结果不正确。

请指出错误所在的代码行号, 并在不增加和删除代码行的情况下进行修改, 写出修改正确后的完整代码行, 使之符合上述设计意图。

【C 函数】

行号	代码
1	<code>int inputArr (int a[], int n) {</code>
2	<code>int i, k = 0;</code>
3	<code>do {</code>
4	<code>scanf("%d", a[k]);</code>
5	<code>k++;</code>
6	<code>if (k=n) break;</code>
7	<code>}while (a[k]==0);</code>
8	<code>return k;</code>
9	<code>}</code>

• (共 15 分)

阅读以下说明和 C 函数, 将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

基于管理的需要, 每本正式出版的图书都有一个 ISBN 号。例如, 某图书的 ISBN 号为“978-7-5606-2348-1”。

ISBN 号由 13 位数字组成: 前三位数字代表该出版物是图书 (前缀号), 中间的 9 个数字分为三组, 分别表示组号、出版者号和书名号, 最后一个数字是校验码。其中, 前缀号由国际 EAN 提供, 已经采用的前缀号为 978 和 979; 组号用以区别出版者国家、地区或者语言区, 其长度可为 1~5 位; 出版者号为各出版者的代码, 其长度与出版者的计划出书量直接相关; 书名号代表该出版者该出版物的特定版次; 校验码采用模 10 加权的算法计算得出。

校验码的计算方法如下:

第一步: 前 12 位数字中的奇数位数字用 1 相乘, 偶数位数字用 3 相乘 (位编号从左到右依次为 13 到 2);

第二步: 将各乘积相加, 求出总和 `S`;

第三步: 将总和 `S` 除以 10, 得出余数 `R`;

第四步: 将 10 减去余数 `R` 后即校验码 `V`。若相减后的数值为 10, 则校验码为 0。

例如, 对于 ISBN 号“978-7-5606-2348-1”, 其校验码为 1, 计算过程为:

$$S = 9 \times 1 + 7 \times 3 + 8 \times 1 + 7 \times 3 + 5 \times 1 + 6 \times 3 + 0 \times 1 + 6 \times 3 + 2 \times 1 + 3 \times 3 + 4 \times 1 + 8 \times 3 = 139$$

$$R = 139 \bmod 10 = 9$$

$$V = 10 - 9 = 1$$

函数 `check(char code[])` 用来检查保存在 `code` 中的一个 ISBN 号的校验码是否正确, 若正确则返回 `true`, 否则返回 `false`。例如, ISBN 号“978-7-5606-2348-1”在 `code` 中的存储布局如表 3-1 所示 (书号的各组成部分之间用“-”分隔):

表 3-1 数组 code 的内容示例

下标	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
字符	9	7	8	.	7	.	5	6	0	6	.	2	3	4	8	.	1	0

在函数 check(char code[])中, 先将 13 位 ISBN 号放在整型数组元素 tarr[0]~tarr[12]中 (如表 3-2 所示, 对应 ISBN 号的位 13~位 1), 由 tarr[0]~tarr[11]计算出校验码放入变量 V, 再进行判断。

表 3-2 数组 tarr 的内容示例

下标	0	1	2	3	4	5	6	7	8	9	10	11	12
字符	9	7	8	7	5	6	0	6	2	3	4	8	1

【C 函数】

```

bool check(char code[])
{
    int i, k = 0;
    int S = 0, temp = 0;
    int V;
    int tarr[13] = {0};

    if (strlen(code) < 17) return false;
    for( i=0; i<17; i++ ) /* 将 13 位 ISBN 号
        if ( code[i] != '-' )
            tarr[ __ (1) __ ] = code[i] - '0' ;

    for( i=0; __ (2) __; i++ ) {
        if ( i%2 )
            S += __ (3) __;
        else
            S += __ (4) __;
    }

    V = ( __ (5) __ == 0 )? 0 : 10 - S %10;
    if ( tarr[12] == V)
        return true;

    return false;
}
    
```

• (共 15 分)

阅读以下说明和 C 程序, 将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

某旅游服务应用程序运行时, 根据输入的两个城市名查找其间的距离。各城市间的距离如表 4-1 所示。表格中的第一行和第一列表示城市名, 表中的每个元素是一个整数, 代表该元素所在行和列对应的城市之间的距离 (单位: km)。

表 4-1

	Beijing	Chengdu	Chongqing	Dalian	Guiyang	Lanzhou	Nanjing	Sanya
Beijing		1697	2695	937	1784	1356	926	2543
Chengdu	1697		313	1840	533	940	1409	1505
Chongqing	2695	313		1734	343	1117	1206	1306
Dalian	937	1840	1734		1995	1594	818	2602
Guiyang	1784	533	343	1995		1113	1346	976
Lanzhou	1356	940	1117	1594	1113		1654	2075
Nanjing	926	1409	1206	818	1346	1654		1806
Sanya	2543	1505	1306	2602	976	2075	1806	

在程序中, 城市名用一维全局数组 cityTable 存储, 城市之间的距离矩阵用二维全局数组 kmTable 表示, 并用相应的值对这两个数组进行初始化。

```
#define NCities 8 /* 城市个数 */
#define TRUE 1
static char * cityTable[NCities] = { /* 城市名按字典序升序排列 */
    "Beijing",
    ....., /* 其他城市名略去 */
    "Sanya",
};
static int kmTable[NCities][NCities] = {
    {0, 1697, 2695, 937, 1784, 1356, 926, 2543},
    {1697, 0, 313, 1840, 533, 940, 1409, 1505},
    ....., /* 剩余元素的初始值略去 */
};
```

程序执行时, 首先按提示输入两个城市名, 然后在 cityTable 中查找与城市名对应的下标, 最后用该下标在 kmTable 中找到这两个城市之间的距离。

程序中定义的函数 FindCityInSortedArray 和 GetCity 说明如下:

- (1) 函数 FindCityInSortedArray 的功能是用二分查找法在全局数组 cityTable 中查找城市名所对应的下标值。
- (2) 函数 GetCity 的功能是读入城市名, 调用函数 FindCityInSortedArray 来获取城市所对应的下标值。如果该城市名不存在, 则提示用户重新输入。

【C 程序】

```
int main() {
    int city1, city2;
    city1 = GetCity("输入第 1 个城市名: ");
    city2 = GetCity("输入第 2 个城市名: ");
    printf("%s 和 %s 之间的距离为: %d km.\n", cityTable[city1],
cityTable[city2],
    kmTable[city1][city2]);
    return 0;
}

static int GetCity(char * prompt) {
    char * cityName;
    int index;
    cityName = (char *)malloc(20*sizeof(char));
    while ( TRUE ) {
        printf("%s", prompt);
        gets(cityName);          /* 获取输入字符串 */
        index = FindCityInSortedArray(cityName);
        if ( __ (1) __ ) break;
        printf("城市名不存在, 请重新输入. \n");
    }
    free(cityName);
    return __ (2) __;
}

static int FindCityInSortedArray(char * key) {
    int lh, rh, mid, cmp;
    lh = 0;
    rh = NCities - 1;
    while ( __ (3) __ ) {
        mid = (lh + rh) / 2;
        cmp = strcmp( __ (4) __ );          /* 比较两个城市名是否相同 */
        if (cmp == 0) return __ (5) __;    /* 两个城市名相同 */
        if (cmp < 0) { rh = mid - 1; }
        else { lh = mid + 1; }
    }
    return (-1);          /* 城市名不存在时返回-1 */
}
```

• (共 15 分)

阅读以下说明和 C++ 代码, 将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

现需要统计某企业员工的月平均工资, 即该企业本月发给员工的工资总和除以员工数。假设企业本月发给员工的工资总和为 sumSalary, 该企业的员工总数为 employeeNumber, 下面的程序

代码计算该企业员工本月的平均工资，其中需要处理 employNumber 为 0 的情况。

【C++代码】

```
#include <iostream>
using namespace std;
class Department{
protected:
float average(float x, int y){
    if (y ==0 ) throw (1);
    return x/y;
}
public:
void caculate(void){
    float sumSalary;
    int employeeNumber;
    try{
        cout << "请输入当月工资总和与员工数:" << endl;
        cin >> sumSalary >> employeeNumber;
        float k = average(sumSalary, employeeNumber);
        cout << "平均工资: " << k << endl;
    }
    (2) (int e){
        if(e == 0){
            cout << "请重新输入当月工资总和与员工数:" << endl;
            cin >> sumSalary >> employeeNumber;
            float k = average(sumSalary, employeeNumber);
            cout << "平均工资: " << k << endl;
        }
    }
};
void main(){
    try {
        (3);
        d.caculate();
    }
    (4) (int e){
        if ( e == 0)
            cout << "程序未正确计算平均工资!" << endl;
    }
}
```

【问题 1】

程序运行时，若输入的员工工资总和为 6000，员工数为 5，则屏幕输出为：

请输入当月工资总和与员工数：

6000 5

(5)

【问题 2】

若程序运行时，第一次输入的员工工资总和为 6000，员工数为 0，第二次输入的员工工资总和为 0，员工数为 0，则屏幕输出为：

请输入当月工资总和与员工数：

6000 0

(6)

0 0

(7)

• (共 15 分)

阅读以下说明和 Java 代码，将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

现需要统计某企业员工的月平均工资，即该企业本月发给员工的工资总和除以员工人数。假设企业本月发给员工的工资总和为 sumSalary，该企业的员工总数为 employeeNumber，下面的程序代码计算该企业员工本月的平均工资，其中需要处理 employeeNumber 为 0 的情况。

【Java 代码】

```
import java.util.Scanner;

public class JavaMain {
    static float average(float x, int y) throws Exception{
        if (y == 0 ) throw new Exception(__(1)__);
        return x/y;
    }
    static void caculate() throws Exception{
        float sumSalary;
        int employeeNumber;
        Scanner sc = new Scanner(__(2)__);
        try{
            System.out.println("请输入当月工资总和与员工数: ");
            sumSalary = sc.nextFloat();    //从标准输入获得工资总和
            employeeNumber = sc.nextInt();  //从标准输入获得员工数
            float k = average(sumSalary, employeeNumber);
            System.out.println("平均工资: " + k);
        }
        __(3)__(Exception e){
            if (e.getMessage().equalsIgnoreCase("zero")){
                System.out.println("请重新输入当月工资总和与员工数: ");
                sumSalary = sc.nextFloat();
                employeeNumber = sc.nextInt();
                float k = average(sumSalary, employeeNumber);
                System.out.println("平均工资: " + k);
            }
        }
    }
    public static void main(String[] args) {
        try {
            caculate();
        }
        __(4)__(Exception e){
            if ( e.getMessage().equalsIgnoreCase("zero"))
                System.out.println("程序未正确计算平均工资! ");
        }
    }
}
```

【问题 1】

程序运行时, 若输入的员工工资总和为 6000, 员工数为 5, 则屏幕输出为:
请输入当月工资总和与员工数:

6000 5

(5)

【问题 2】

若程序运行时，第一次输入的员工工资总和为 6000，员工数为 0，第二次输入的员工工资总和为 0，员工数为 0，则屏幕输出为：

请输入当月工资总和与员工数：

6000 0

(6)

0 0

(7)

希赛网在线题库