

考试时间为4小时。

本试卷包含两种题型：“代码填空”与“程序设计”。总计100分。

其中代码填空：3+4+4+5+6+9+10 = 41 分

程序设计：15+16+28 = 59分

填空题要求参赛选手在弄清给定代码工作原理的基础上填写缺失的部分，使得程序逻辑正确、完整。所填写的代码不超过一条语句（即不能出现分号）。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM**下对应题号文件夹中的“解答.txt”中即可。

编程题要求选手设计的程序对于给定的输入能给出正确的输出结果。考生的程序只有能运行出正确结果的时候才有机会得分。注意：在评卷时使用的输入数据与试卷中给出的实例数据可能是不同的。选手的程序必须是通用的，不能只对试卷中给定的数据有效。

对每个题目，要求考生把所有函数写在一个文件中。调试好后，存入**考生文件夹（C:\EXAM**下对应题号文件夹中的“解答.txt”中即可。相关的工程文件不要拷入。

对于编程题目，要求选手给出的解答完全符合ANSI C标准，不能使用c++特性；不能使用诸如绘图、中断调用等硬件相关或操作系统相关的API。

1. 代码填空（满分3分）

神秘的三位数

有这样一个3位数，组成它的3个数字阶乘之和正好等于它本身。即： $abc = a! + b! + c!$

下面的程序用于搜索这样的3位数。请补全缺失的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM**下对应题号文件夹中的“解答.txt”中即可。

```
int JC[] = {1, 1, 2, 6, 24, 120, 720, 5040, 40320, 362880};
int i;
for(i=100; i<1000; i++)
{
    int sum = 0;
    int x = i;
    while( )
    {
        sum += JC[x%10];
        x /= 10;
    }
    if(i==sum) printf("%d\n", i);
}
```

2. 代码填空（满分4分）

歌赛新规则

歌手大赛的评分规则一般是去掉一个最高分，去掉一个最低分，剩下的分数求平均。当评委较少的时候，如果我们只允许去掉一个分数，该如何设计规则呢？有人提出：应该去掉与其余的分数平均值相差最远的那个分数。即“最离群”的分数。

以下的程序用于实现这个功能。其中x存放所有评分，n表示数组中元素的个数。函数返回最“离群”的那个分数值。请补全缺失的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM__）**下对应题号文件夹中的“解答.txt”中即可。

```
double score(double x[], int n)
{
    int i, j;
    double dif = -1;
    double bad;
    for(i=0; i<n; i++)
    {
        double sum = 0;
        for(j=0; j<n; j++)
        {
            if(_____) sum += x[j];
        }
        double t = x[i] - sum / (n-1);
        if(t<0) t = -t;
        if(t>dif)
        {
            dif = t;
            bad = x[i];
            printf("%d, %f\n", i, x[i]);
        }
    }

    return bad;
}
```

3. 代码填空（满分4分）

反转串

我们把“cba”称为“abc”的反转串。

下面的代码可以把buf中的字符反转。其中n表示buf中待反转的串的长度。请补充缺少的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM\ ）**下对应题号文件夹中的“解答.txt”中即可。

```
void reverse_str(char* buf, int n)
{
    if(n<2) return;
    char tmp = buf[0];
    buf[0] = buf[n-1];
    buf[n-1] = tmp;
    _____;
}
```

4. 代码填空（满分5分）

n进制小数

将任意十进制正小数分别转换成2, 3, 4, 5, 6, 7, 8, 9进制正小数，小数点后保留8位，并输出。例如：若十进制小数为0.795，则输出：

十进制正小数 0.795000 转换成 2 进制数为：0.11001011

十进制正小数 0.795000 转换成 3 进制数为：0.21011011

十进制正小数 0.795000 转换成 4 进制数为：0.30232011

十进制正小数 0.795000 转换成 5 进制数为：0.34414141

十进制正小数 0.795000 转换成 6 进制数为：0.44341530

十进制正小数 0.795000 转换成 7 进制数为：0.53645364

十进制正小数 0.795000 转换成 8 进制数为：0.62702436

十进制正小数 0.795000 转换成 9 进制数为：0.71348853

以下代码提供了这个功能。其中，dTestNo表示待转的十进制小数。iBase表示进制数。请填写缺失的部分。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM\ 一）**下对应题号文件夹中的“解答.txt”中即可。

```
void fun(double dTestNo, int iBase)
{
    int iT[8];
    int iNo;

    printf("十进制正小数 %f 转换成 %d 进制数为：", dTestNo, iBase);

    for(iNo=0; iNo<8; iNo++)
    {
        dTestNo *= iBase;
        iT[iNo] = _____;
        if(_____) dTestNo -= iT[iNo];
    }

    printf("0. ");
    for(iNo=0; iNo<8; iNo++) printf("%d", iT[iNo]);
    printf("\n");
}

void main ( )
{
    double dTestNo= 0.795;
    int iBase;

    for(iBase=2; iBase<=9; iBase++)
        fun(dTestNo, iBase);
    printf("\n");
}
```

5. 代码填空（满分6分）

轮换

串“abcd”每个字符都向右移位，最右的移动到第一个字符的位置，就变为“dabc”。这称为对串进行位移=1的轮换。同理，“abcd”变为：“cdab”则称为位移=2的轮换。下面的代码实现了对串s进行位移为n的轮换。请补全缺失的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM\...**下对应题号文件夹中的“解答.txt”中即可。

```
void shift(char* s, int n)
{
    char* p;
    char* q;
    int len = strlen(s);
    if(len==0) return;
    if(n<=0 || n>=len) return;

    char* s2 = (char*)malloc(_____);
    p = s;
    q = s2 + n % len;
    while(*p)
    {
        *q++ = *p++;
        if(q-s2>=len)
        {
            *q = _____;
            q = s2;
        }
    }
    strcpy(s, s2);
    free(s2);
}
```

6. 代码填空（满分9分）

中奖计算

某抽奖活动的规则是：每位参与者在纸上写下一个8位数的号码。最后通过摇奖的办法随机产生一个8位数字。参与者写下的数字中最多有多少个连续位与开奖号码中的相同，则称为中了几个号。

例如：小张写的数字是：12345678，而开奖号码是：42347856。则称小张中了3个号，因为其中最长的相同连续位是：“234”。如果小张写的是：87654321，则他只中了一个号。

下面的代码根据传入的参数，返回中了几个号。其中：a表示被评价的号码，b表示摇号产生的数字。请填写缺少的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM** **）**下对应题号文件夹中的“解答.txt”中即可。

```
int g(int a, int b)
{
    char sa[]="00000000";
    char sb[]="00000000";
    int n = 0;
    int i, j;

    sprintf(sa, "%8d", a);
    sprintf(sb, "%8d", b);
    for (i=0; i<8; i++)
    {
        for (j=1; j<=8-i; j++)
        {
            char t = _____;
            sa[i+j] = 0;
            if (strstr(sb, sa+i))
            {
                if (j>n) _____;
            }
            sa[i+j] = t;
        }
    }

    return n;
}
```

7. 代码填空（满分10分）

概率问题

某个袋子中有红球 m 个，白球 n 个。现在要从中取出 x 个球。那么红球数目多于白球的概率是多少呢？

下面的代码解决了这个问题。其中的 y 表示红球至少出现的次数。

这与前文的问题是等价的。因为如果取30个球，要求红球数大于白球数，则等价于至少取出16个红球。请根据仅存的线索，判断程序逻辑，并补全缺少的代码。

把填空的答案（仅填空处的答案，不包括题面）存入**考生文件夹（C:\EXAM**___下对应题号文件夹中的“解答.txt”中即可。

```
/*
    m: 袋中红球的数目
    n: 袋中白球的数目
    x: 需要取出的数目
    y: 红球至少出现的次数
*/
double pro(int m, int n, int x, int y)
{
    if(y>x) return 0;
    if(y==0) return 1;
    if(y>m) return 0;
    if(x-n>y) return 1;
    double p1 = _____;
    double p2 = _____;
    return (double)m/(m+n) * p1 + (double)n/(m+n) * p2;
}
```


8. 程序设计（满分15分）

方阵的主对角线之上称为“上三角”。

请你设计一个用于填充 n 阶方阵的上三角区域的程序。填充的规则是：使用1, 2, 3...的自然数列，从左上角开始，按照顺时针方向螺旋填充。

例如：当 $n=3$ 时，输出：

1 2 3

6 4

5

当 $n=4$ 时，输出：

1 2 3 4

9 10 5

8 6

7

当 $n=5$ 时，输出：

1 2 3 4 5

12 13 14 6

11 15 7

10 8

9

程序运行时，要求用户输入整数 n （ $3 \sim 20$ ）

程序输出：方阵的上三角部分。

要求格式：每个数据宽度为4，右对齐。

要求考生把所有函数写在一个文件中。调试好后，存入与**考生文件夹（C:\EXAM\）**下对应题号文件夹中的“解答.txt”中即可。相关的工程文件不要拷入。

对于编程题目，要求选手给出的解答完全符合ANSI C标准，不能使用c++特性；不能使用诸如绘图、中断调用等硬件相关或操作系统相关的API。

9. 程序设计（满分16分）

公司发了某商店的购物券1000元，限定只能购买店中的 m 种商品。每种商品的价格分别为 m_1, m_2, \dots ，要求程序列出所有的正好能消费完该购物券的不同购物方法。

程序输入：

第一行是一个整数 m ，代表可购买的商品的种类数。

接下来是 m 个整数，每个1行，分别代表这 m 种商品的单价。

程序输出：

第一行是一个整数，表示共有多少种方案

第二行开始，每种方案占1行，表示对每种商品购买的数量，中间用空格分隔。

例如：

输入：

2
200
300

则应输出：

2
2 2
5 0

输入：

2
500
800

则应输出：

1
2 0

要求考生把所有函数写在一个文件中。调试好后，存入与**考生文件夹（C:\EXAM** **）**下对应题号文件夹中的“解答.txt”中即可。相关的工程文件不要拷入。对于编程题目，要求选手给出的解答完全符合ANSI C标准，不能使用c++特性；不能使用诸如绘图、中断调用等硬件相关或操作系统相关的API。

10程序设计（满分28分）

一种Playfair密码变种加密方法如下：首先选择一个密钥单词（称为pair）（字母不重复，且都为小写字母），然后与字母表中其他字母一起填入至一个5x5的方阵中，填入方法如下：

1. 首先按行填入密钥串。
2. 紧接其后，按字母序按行填入不在密钥串中的字母。
3. 由于方阵中只有25个位置，最后剩下的那个字母则不需变换。

如果密钥为youandme，则该方阵如下：

```
y o u a n
d m e b c
f g h i j
k l p q r
s t v w x
```

在加密一对字母时，如am，在方阵中找到以这两个字母为顶点的矩形：

```
o u a
m e b
```

这对字母的加密字母为该矩形的另一对顶点，如本例中为ob。

请设计程序，使用上述方法对输入串进行加密，并输出加密后的串。

另外有如下规定：

- 1、一对一对取字母，如果最后只剩下一个字母，则不变换，直接放入加密串中；
- 2、如果一对字母中的两个字母相同，则不变换，直接放入加密串中；
- 3、如果一对字母中有一个字母不在正方形中，则不变换，直接放入加密串中；
- 4、如果字母对出现在方阵中的同一行或同一列，如df或hi，则只需简单调换这两个字母，即变换为fd或ih；
- 5、如果在正方形中能够找到以字母对为顶点的矩形，假如字母对为am，则该矩形的另一对顶点字母中，与a同行的字母应在前面，在上例中应是ob；同样若待变换的字母对为ta，则变换后的字母对应为wo；
- 6、本程序中输入串均为小写字母，并不含标点、空格或其它字符。

解密方法与加密相同，即对加密后的字符串再加密，将得到原始串。

要求输入形式如下：

从控制台输入两行字符串，第一行为密钥单词（长度小于等于25），第二行为待加密字符串（长度小于等于50），两行字符串末尾都有一个回车换行符，并且两行字符串均为小写字母，不含其它字符。

在标准输出上输出加密后的字符串。

例如，若输入：

```
youandme
welcometohangzhou
```

则表示输入的密钥单词为youandme，形成的正方形如上所示；待加密字符串为welcometohangzhou。在正方形中可以找到以第一对字母we为顶点的矩形，对应另一对顶点字母为vb，因此加密后为vb，同理可找到与字母对lc, et, oh, ho对应的顶点字母对。而字母对om位于上述正方形中的同一列，所以直接以颠倒这两个字母来加密，即为mo，字母对an同理。字母对gz中的z不在上述正方形中，因此原样放到加密串中。最后剩一个字母u也原样输出。

因此输出的结果为：

```
vbrmmomvugnagzguu
```

要求考生把所有函数写在一个文件中。调试好后，存入与**考生文件夹（C:\EXAM\ 下）**对应题号文件夹中的“解答.txt”中即可。相关的工程文件不要拷入。

对于编程题目，要求选手给出的解答完全符合ANSI C标准，不能使用c++特性；不能使用诸如绘图、中断调用等硬件相关或操作系统相关的API。