

# 《C 语言程序设计》课程期中试卷 2019.10

## 一、写出下列程序段的运行结果（40 分）

1、

```
int a = 20, b = 10, c;  
c = ++a && b;  
printf("%d,%d,%d,", a,b,c);  
c = a || ++b;  
printf("%d,%d,%d", a,b,c);
```

2、

```
#define N 5  
int i,t,a[]={1,2,3,4,5};  
for(i=0;i<N/2;i++)  
{  
    t=a[i];  
    a[i]=a[N-i-1];  
    a[N-i-1]=t;  
}  
for(i=0;i<N;i++)  
{  
    printf("%d  ",a[i]);  
}
```

3、

```
float f = 41.0, c;  
double t = 3.14159, v;  
c = 5 / 9 * (f - 32);  
f *= 2.0 + 1.0;  
t += 2.0;  
v = t / 10;  
printf("f=%.2f", f);  
printf("c=%.2f", c);  
printf("t=%.2f", t);  
printf("v=%.4f", v);
```

4、

```
int a=5, b=4, c=3, k=11;  
if (a=b+c)  
    printf("Equal!\n");  
else  
    printf("Not Equal!\n");  
printf("k=%d,k=%o,k=%x\n",k,k,k);
```

5、

```
float s[6]={1, 3, 5, 7, 9};  
float x;  
int i;  
scanf("%f", &x);  
for (i=4 ; i>=0; i--)  
    if(s[i]>x)  
        s[i+1]=s[i];  
    else break;  
printf("%d \n",i+1);  
//假设输入为: 4
```

6、

```
int i;  
char str[]="university";  
for(i=0; i<=3; i++)  
    switch(i)  
    {  
        case 0: printf("%c ",str[i]);  
        case1: printf("%c\n",str[i]);  
                break;  
        case 2: printf("%s\n",str);  
        default: printf("****");  
    }
```

7、

```
char ch; int digit, i, minus, other;
digit=minus=other= 0;
for(i = 1; i <= 5; i++)
{
    ch = getchar();
    switch (ch)
    {
        case '0': case '1': case '2':
        case '3': case '4':
        case '-': minus++; break;
        case '5': case '6':
        case '7': case '8':
        case '9': digit++;break;
        default: other++; break;
    }
}
printf(“%d,%d,%d\n”, inus,digit,other);
//假设输入为: 1+4-5
```

8、

```
for(num = 99; num <= 101; num++)
{
    s = 0; /*① */
    t_num = num; /*② */
    do{
        s = s + t_num % 10;
        t_num = t_num / 10; /*③ */
    }while(t_num != 0);
    printf(“%d\n”, s); /*④ */
}
//问, 语句①②③④各执行了多少次?

```

9、

```
double eps;
scanf ("%lf", &eps);
int i = 1, flag = 1, denominator = 1;
double item = 1.0, s = 0;
while(fabs(item) >= eps)
{
    s = s + item;
    i++;
    flag = -flag;
    denominator = denominator + 2;
    item = flag * 1.0 / denominator;
}
s = s + item;
printf ("%0.2f,%d\n", s,i);
//假设输入为: 1
```

10、

```
int i=5, s=0;
while(i--)
if (i%2) continue;
else s+=i;
printf(“%d,%d\n”,i,s);
```

## 二、改错题（共 20 分）

本题中给定的程序预期实现如下功能：

从键盘输入一串字符，统计这一串字符中各个数字出现的次数，将统计结果存入数组中；然后出现的频率对按各个数进行降序排序，频率相同的数字则根据数字自身从小到大排序；最后依次输出统计结果，每个数字单独一行显示，次数为 0 的数字不显示。例如：输入 12x&34ac2345\*C(+3456，程序输出如下：

```
数字 3 出现了：3 次
数字 4 出现了：3 次
数字 2 出现了：2 次
.....
```

本程序前 10 行包含了 4 个错误，其余行包含了若干错误，请指出错的行并改正。

```

#include <stdio.h>
int main()
{
    int count[ ];
    char c,tmpd, d[]="0123456789";
    int i,j,tmp;
    while(c=getchar()!='\n')
    {
        if(0<= c <=9)
            count[c]++;
    }
    for(i =0; i<9; i++)
    {
        for(j=i; j<9; j++)
        {
            if((count[j]<count[j+1]) ||
                (count[j]==count[j+1]) &(d[j] < d[j+1]))
            {
                tmp=count[j], count[j+1]=count[j], count[j] = tmp;
                tmpd = d[j], d[j+1]= d[j] ,d[j]=tmpd;
            }
        }
    }
    for(i=0;i<=10;i++)
    {
        if(count[i]==0)
            break;
        printf("数字%d 出现了:%d 次\n",d[i],count[i]);
    }
}

```

```

/* 第 1 行 */
/* 第 2 行 */
/* 第 3 行 */
/* 第 4 行 */
/* 第 5 行 */
/* 第 6 行 */
/* 第 7 行 */
/* 第 8 行 */
/* 第 9 行 */
/* 第 10 行 */
/* 第 11 行 */
/* 第 12 行 */
/* 第 13 行 */
/* 第 14 行 */
/* 第 15 行 */
/* 第 16 行 */
/* 第 17 行 */
/* 第 18 行 */
/* 第 19 行 */
/* 第 20 行 */
/* 第 21 行 */
/* 第 22 行 */
/* 第 23 行 */
/* 第 24 行 */
/* 第 25 行 */
/* 第 26 行 */
/* 第 27 行 */
/* 第 28 行 */
/* 第 29 行 */
/* 第 30 行 */

```

### 三、编程题（40 分=12+14 +14 注意：程序中请添加必要的注释）

1. （12 分）请编写一个程序，对给定的一个字符串（仅含字母，长度不超过 100），统计其中连续出现的相同字符个数，并转换为“字母+数量”格式输出。例如，输入“AAABBBCCCab”，输出“A3 B3 C3 a1 b1”。

2. （14 分）请编写一个程序，对给定的若干个 0~9 数字，进行排列使得到的正整数尽可能小（0 不能做首位数字）。

输入格式：第一行给出数字个数  $n$  ( $n \leq 100$ )；第二行给出 0~9 范围内的  $n$  个数字（至少含 1 个非 0 数字），中间用一个空格分隔。

输出格式：给出可构成的最小正整数。

输入样例：

10

2 4 0 5 0 3 0 2 1 0

输出样例：

1000022345

3. （14 分）假定四则运算表达式由操作数（数字及至多 1 个小数点）、运算符（仅+、-、\*、/）以及小括号（(和)）组成。请编写一个程序，对于给定的一个四则运算表达式，识别出其中的操作数，顺序存储在二维字符数组 Ops[10][10] 中（每个一行，不会越界），并依次输出在屏幕上。

输入格式：在一行中给出长度不超过 100 个字符的四则运算表达式，仅由上述字符组成无空格，无需做正确性检查。

输出格式：依次输出式中各操作数，每个一行。

输入样例：

32\*(62-5)

输出样例：

32

62

5