



学年学期: 24251 主考教师: 曾华琳, 林贤明 等

一、写出下列程序段的运行结果 (40 分)

1. (4 分)

```
int w=3,z=7,x=10,y=20;

printf("%d", x>y>z);
printf("%d", x>10?x+100:x-10);
printf("%d", !(w=z) || 0 && (w=x));
printf("%d\n", y-=x*=w);
```

2. (4 分)

```
int x=3,y=4,z=5,a=7,
    m=011,n=0x5a;
float b=2.5,c=4.7,t;
char c1='\101',c2='\x42';

t=b+a%3*(int)(b+c)%2/4;
if(!(x<y))
    printf("%c%c\n",c1,c2);
if(y+z)
    printf("%f\n",t);
else
    printf("%d%d\n",m,n);
printf("%d\n",n%=(m/2));
```

3. (4 分)

```
int x[3][3]={ {1},{1,2},{2}}, i;
for(i=0;i<3;i++)
{
    x[i][i]*=2;
    if (x[i][2-i]!=0) break;
    x[i][2-i]=x[2-i][i];
}

printf("%d,%d,%d,%d\n",
    x[0][0],x[1][1],x[2][0],x[2][2]);
```

4. (4 分)

```
int i=0,k=0,
    p[7]={11,13,14,15,16,17,18};
while (p[i]%2 && i<7)
    k=k+p[++i];
printf("%d,%d\n", k, i);
```

5. (4 分)

```
int a=0,b=0,s=0,t=0;
char c;
for(;(c=getchar())!='\n';)
    switch(c)
    {
        case '0': case '1': a++;
        case '2': case '3': b++;
        default:
            c>='A' && c<='Z' ?s++:t++;
    }
printf("%d %d %d,%d",
    a, b, s, t);
运行时输入: 20241108Xmu
```

6. (4 分)

```
char a[11]={'\0'};
int sum=0, i=0, j=0;

for (; i<10; i++)
    a[i]='a'+i;
do
{
    a[--i]=a[j++];
    sum+=1;
}while(i>j);
printf("%s\nsum=%d,i=%d,
    j=%d",a,sum,i,j);
```

7. (4分)

```
int a=-2;
char b=128;
while(a++)
    b+=1;
printf("%d, %d",a,b);
```

8. (4分)

```
int a[10]={3,4,6,1,3,7,6,2,5,4};
int i,j,k=0,e=0;
for(i=0;i<=9;i++)
    for(j=0;j<=9;j++) {
        if((a[i]==a[j])&&(i!=j)){
            k++;
            break;
        }
    }
if(j==9){
    if(e==0)
        printf("%d",a[i]);
    else
        printf(" %d",a[i]);
    e++;
}
```

9. (4分)

```
char str[10] = "2024_exam";
int i, j, k;
for (i = 0; str[i] != '\0'; i+=2) {
    char tempstr=str[i];
    str[i]=str[i+1];
    str[i+1]=tempstr;
}
printf("%d\n%s",
        strlen(str),str);
```

10. (4分)

```
int i,j,t,sum=0;
for(t=i=1;j<=6;i++)
{
    sum+=t;
    if(i%3==0) continue;
    else t=i;
}
printf("sum=%d,i=%d",sum,i);
```

二、改错题 (共 20 分)

1. 给定一行句子，句中每个词之间用一个空格隔开，句中的单词可能是全小写英文单词、全大写英文单词或自然数。现要求将句子以单词为基本单位倒序输出，并将英文单词进行大小写转换；将自然数转换为逆序数。(提示：句子长度 ≤ 1000 ，单词数 ≤ 100 ，单词长度 ≤ 15)

● 输入格式

仅一行，即需要反转的句子。

● 输出格式

仅一行，对句子的处理结果。

输入样例：

WE choose TO go 2 the 123 moon

输出样例：

MOON 321 THE 2 GO to CHOOSE we

本程序包含了 10 个错误，请指出错误所在的行，并进行改正。

```
#include <stdio.h> //第 1 行
#include<string.h> //第 2 行
void main() //第 3 行
{
    char s[1000]; //第 4 行
    char words[100][16]; //第 5 行
    int i, j, len; //第 6 行
    int wcount=0, wlen, index=0, inaword=0; //第 7 行
    char c; //第 8 行
    scanf("%s", s); //第 9 行
```

```

len=strlen(s);
for(i=0;i<len;i++)
{
    if(sentence[i]!=" ")
    {
        if(inaword == 0 )
        {
            wcount++;
            inaword = 1;
        }
        words[wcount-1][index]=s[i];
    }
    else
    {
        words [wcount-1][index]='\0';
        inaword = 0;
    }
}
words[wcount-1][index]='\0';
for(i=0;i<wcount; i++)
{
    wlen = strlen(words[i]);
    if(words[i][0]>=0 && words[i][0]<= 9 )
    {
        for(j=0;j<wlen;j++)
        {
            c = words[i][j];
            words[i][j] = words[i][wlen-j-1];
            words[i][wlen-j-1] = c;
        }
    }
    else
    {
        for(j=0 ; j<wlen ; j++)
        {
            words[i][j] = words[i][j] + (words[i][0]>='A'
            &&words[i][0])<='Z'?-32,32;
        }
    }
}
for(i=0; i<wcount ;i++)
{
    if(i>0) printf("%s ",words[i]);
    else printf("%s\n",words[i]);
}
}

```

//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
//第 31 行
//第 32 行
//第 33 行

三、编程题（40 分=12+14+14 注意：程序中请添加必要的注释）

1. 在一片茂密的深林里散落着 $N(N \leq 1000)$ 块磁石，每个磁石的性质可以用一个整数五元组 (x, y, m, p, r) 描述。其中 x, y 表示其坐标， m 是磁石的质量， p 是磁力， r 是吸引半径。若磁石 B 到磁石 A 的距离不大于磁石 A 的吸引半径，并且磁石 B 的质量不大于磁石 A 的磁力，那么 A 就可以吸引到 B 。探险家张三有一块磁石 L ，请问他在 (x_0, y_0) 处用 L 能吸引到几块磁石？

● 输入格式：

第一行五个整数 $x_0 y_0 p_0 r_0 N$ ，分别表示：张三所在的位置 (x_0, y_0) 磁石 L 的磁力 p_0 、吸引半径 r_0 和森林中散落的磁石个数 N 。接下来 N 行为 N 块磁石的性质，每行五个整数 $x y m p r$ 。

● 输出格式：

输出一个整数，表示他用 L 吸引到的散落磁石个数（不含自己携带的磁石 L ）。

输入样例：	能否吸引得到
2 3 4 5 4	不可以
2 1 5 4 4	可以
1 2 3 7 6	可以
6 6 3 4 4	不可以
3 3 9 8 8	
输出样例：	

2. 机场的廊桥停机位不需乘坐摆渡车，对乘客更方便，但租金费用更高。某航空公司近期为了提升用户的满意度，拟通过租用更多廊桥停机位，以确保每个航班都可以停靠在廊桥停机位上。已知航空公司第二天的航班停靠时刻信息，请编写程序计算至少需要租用多少廊桥停机位，才能满足航班停靠需求。假设飞机都是整点停靠和降落，机场的廊桥停机位采用“先到先得”的原则，一个廊桥停机位只能停放一架飞机，而且不会有两个航班在同一时刻起飞或者降落。

● 输入格式：

输入的第 1 行 n 为第二天航班数量 ($1 \leq n \leq 24$)，接下来 n 行是航空公司的航班信息，第 i 行包含两个正整数 a_i, b_i ，分别表示航班 i 的抵达和离开时刻。

● 输出格式：

输出 1 个正整数，表示航空公司至少需要租用的廊桥停机位数量。

输入样例 1:

4

4 7

5 6

8 14

9 11

输出样例:

2

输入样例 2:

4

2 15

4 11

7 17

12 16

输出样例:

3

3. 某单位充分发扬民主精神，全体员工参与工会主席的选举投票。假设员工总共有 n 人，竞选候选人有 m 位，编号分别为：1, 2, ..., m 。每位员工只投一票，选票上写明他所支持的获选人编号。请帮忙编写程序，完成投票结果统计。具体输入输出格式如下：

● 输入格式：

第一行两个整数： n (≤ 10000 ，员工总人数) 和 m ($m < 10$ ，候选人数)；第二行给出 n 个整数，分别代表每个员工的投票情况，数值在 $[1, m]$ 的票为有效投票，否则为无效票。

● 输出格式：

如果有效票数量小于总人数 n 的三分之二，则输出“无效投票！”；否则，按照得票数递减的顺序输出每个候选人的编号和对应的票数，格式为：“No i : x 票”，得票数一样的候选人则按照编号的递增顺序排序。每个候选人的得票情况输出一行，得票数为 0 的获选人不输出。

输入样例:

8, 4

1 2 1 2 4 2 4 2

输出样例:

No 2 : 4 票

No 1 : 2 票

No 4 : 2 票