

C 语言真题

Author:牧阳 MuYoung

From:网上总结

//有 15 个人围成一圈 ,顺序从 1 到 15 编号。从第一个人开始报数 ,凡报到 n 的人退出圈子。输入 n(n>=1) 的值 , 输出最后留在圈子里的人的编号

```
#include<stdio.h>
void main(){
    int num[50];
    int i,j,k,m,n,N;
    int *p;
    printf("请输入总人数和 n : \n");
    scanf("%d%d",&N,&n);
    p=num;
    for(i=0;i<N;i++)
        *(p+i)=i+1;
    i=0;//i 为循环变量
    k=0;//k 为报数时的计数变量
    m=0;//m 为退出人数
    while(m<N-1){
        if(*(p+i)!=0)
            k++;
        if(k==n){
            *(p+i)=0;
            k=0;
            m++;
        }
        i++;
        if(i==N)
            i=0;
    }
    while(*p==0)
        p++;
    printf("最后还在圈子里的是 : %d 号",*p);
    return;
}
```

//编写函数 , 用冒泡排序法或选择排序法对输入的 100 个整数按从小到大的顺序排序

```
#include<stdio.h>
void main(){
    int a[100];
    int i,j,t;
    printf("请输入 100 个整数 : \n");
    for(i=0;i<100;i++)
        scanf("%d",&a[i]);
    for(i=0;i<99;i++){
        for(j=0;j<99-i;j++){
            if(a[j]>a[j+1]){
                t=a[j];
```

```

                a[j]=a[j+1];
                a[j+1]=t;
            }
        }
    }
    printf("排序后的结果为：\n");
    for(i=0;i<100;i++)
        printf("%d ",a[i]);
    printf("\n");
}

```

//采用递归法，编写实现 n!的函数。

```

#include<stdio.h>
int digui(int n){
    if(n==0||n==1)
        return 1;
    else
        return n*digui(n-1);
}
void main(){
    int n,num;
    printf("请输入一个正整数：\n");
    scanf("%d",&n);
    num=digui(n);
    printf("%d ! = %d",n,num);
}

```

//b 为 6*6 的方阵，先输入 36 个数据形成该方阵，然后在完成下列操作：

//1 计算两条对角线元素的和与乘积

//2 计算方阵汇总所有不靠边元素之和

//3 计算方阵中所有靠边元素之和

//4 使两条对角线上的元素均为 1，其余元素均为 0

```

#include<stdio.h>
void main(){
    int b[6][6];
    int i,j;
    int result11=0;//主对角线元素之和
    int result12=1;//主对角线元素之积
    int result2=0;//不靠边元素之和
    int result3=0;//靠边元素之和

    printf("请输入 36 个数：\n");
    for(i=0;i<6;i++)
        for(j=0;j<6;j++)
            scanf("%d",&b[i][j]);
}

```

```

printf("组成的矩阵为 : \n");
for(i=0;i<6;i++){
    for(j=0;j<6;j++){
        printf("%2d",b[i][j]);
        printf("\n");
    }

    //对角元素之和
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            if(i==j){
                result11+=b[i][j];
                result12*=b[i][j];
            }
            if(i+j==5){
                result11+=b[i][j];
                result12*=b[i][j];
            }
        }
    }
    printf("对角线元素之和 : %d\n",result11);
    printf("对角线元素之积 : %d\n",result12);
    //不靠边元素之和
    for(i=1;i<5;i++){
        for(j=1;j<5;j++){
            result2+=b[i][j];
        }
    }
    printf("不靠边元素之和 : %d\n",result2);
    //靠边元素之和
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            if(i==0||i==5)
                result3+=b[i][j];
            else if(j==0||j==5)
                result3+=b[i][j];
        }
    }
    printf("不靠边元素之和 : %d\n",result3);
    //使两条对角线上的元素为 1 , 其余元素为 0
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            if(i==j)
                b[i][j]=1;
            else if(i+j==5)
                b[i][j]=1;
            else
                b[i][j]=0;
        }
    }
}

```

```

    }
    printf("转换后的矩阵为 : \n");
    for(i=0;i<6;i++){
        for(j=0;j<6;j++){
            printf("%2d",b[i][j]);
            printf("\n");
        }
    }
}

```

//编写一个候选人得票的统计程序，设有 m 个选举人，n 个候选人，
 //每输入一个得票候选人的名字，要求输出每个候选人的得票结果。

```

#include<stdio.h>
#include<string.h>
#define N 5
struct candidate{
    char name[20];
    int vote;
};
void main(){
    struct candidate man[N]={
        {"A",0},{ "B",0},{ "C",0},{ "D",0},{ "E",0}
    };
    char votename[20];
    int i;
    printf("候选人名字 : \n");
    for(i=0;i<N;i++){
        printf("%s,",man[i].name);
    }
    printf("\n 请根据上述名字投票(以 end 结束) : \n");
    while(1){
        scanf("%s",votename);
        if(strcmp(votename,"end")==0)
            break;
        for(i=0;i<N;i++){
            if(strcmp(votename,man[i].name)==0)
                man[i].vote++;
        }
        for(i=0;i<N;i++){
            printf("候选人 : %s 得票 : %d\n",man[i].name,man[i].vote);
        }
    }
}

```

//编写一个函数，该函数有三个参数，一个是二维数组，一个是二维数组的行数，
 //最后一个二维数组的列数，输出该二维数组两条对角线元素之和

```

#include<stdio.h>
void add(int s[100][100],int n){

```

```

int i,j,sum1=0,sum2=0;
for(i=0;i<n;i++)
    for(j=0;j<n;j++)
        if(i==j)
            sum1+=s[i][j];
for(i=0;i<n;i++)
    for(j=n-1;j>=0;j--)
        if(i+j==(n-1))
            sum2+=s[i][j];
printf("主对角线元素和为：%d\n",sum1);
printf("副队角线元素和为：%d\n",sum2);
}
void main(){
    int s[100][100];
    int i,j,n;
    printf("请输入矩阵阶数：\n");
    scanf("%d",&n);
    printf("请输入矩阵数字：\n");
    for(i=0;i<n;i++)
        for(j=0;j<n;j++)
            scanf("%d",&s[i][j]);
    add(s,n);
}

```

//编写程序，该程序的功能是确定字符串中首次出现的某字符在串中的位置（即该字符是字符串中的第几个字符）然后从字符串中删除该字符。要求：

//1 如果未找到该字符，程序给出相应信息，否则，输出该字符在字符串中（不考虑非首次出现的该字符的删除），并且显示删除前后的字符串

//2 通过键盘输入字符串以及被确定的字符

```

#include<stdio.h>
#include<string.h>
void main(){
    char s1[100],s;
    int i,j=-1,len=0;
    printf("请输入字符串：\n");
    gets(s1);
    printf("请输入字符：\n");
    scanf("%c",&s);

    len=strlen(s1);
    for(i=0;i<len;i++)
        if(s1[i]==s){
            j=i;
            break;
        }
}

```

```

    if(j!=-1){
        printf("%c 在字符串中首次出现的位置：%d\n",s,j+1);
        printf("删除前的字符串为：%s\n",s1);
        for(i=j;i<=len-1;i++){
            s1[i]=s1[i+1];
        }
        printf("删除后的字符串为：%s\n",s1);
    }else
        printf("该字符在字符串中不存在！\n");
}

```

//设计算法并编写函数 int findMax_Min(int a[],int n)，有如下功能：

//在具有 n 个元素的数组 a 中找到最大和最小元素并输出，函数返回值为最大和最小值之和

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

```

int findMax_Min(int a[],int n){
    int i,max=a[0],min=a[0];
    for(i=0;i<n;i++){
        if(a[i]>max)
            max=a[i];
        if(a[i]<min)
            min=a[i];
    }
    printf("最大值为：%d;最小值为：%d",max,min);
    return max+min;
}

```

//已知结构体，编写一个函数 int caortx(struct LNode *HL,int x)

//统计出单链表中结点值等于给定值 x 的结点个数

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

```

struct LNode{
    int data;
    struct LNode *next;
};

int caortx(LNode *HL,int x){
    int k=0;
    LNode *p = HL->next;//假设 HL 有头结点
    while(p!=NULL){
        if(p->data==x)
            k++;
        p = p->next;
    }
    return k;
}

```

//编写统计候选人得票的程序，设有十个候选人，有 100 个人参加投票，

//每次输入一个得票的候选人的名字，要求最后统计输出每个候选人的得票结果

```
#include<stdio.h>
#include<string.h>
struct candidate{
    char name[20];
    int vote;
};
void main(){
    int i;
    char votename[20];
    struct candidate man[10]={
        {"a",0}, {"b",0}, {"c",0}, {"d",0}, {"e",0},
        {"f",0}, {"g",0}, {"h",0}, {"i",0}, {"j",0}
    };
    printf("候选人名字为：\n");
    for(i=0;i<10;i++)
        printf("%s ",man[i].name);
    printf("\n 请根据以上名字投票（以 end 结束）：\n");
    while(1){
        scanf("%s",votename);
        if(strcmp(votename,"end")==0)
            break;
        for(i=0;i<10;i++){
            if(strcmp(votename,man[i].name)==0)
                man[i].vote++;
        }
    }
    for(i=0;i<10;i++)
        printf("%s 的得票数： %d\n",man[i].name,man[i].vote);
}
```

//编写一个函数，功能：从一个整数 m 中，统计其中各位上等于 n 的数字数目，并返回，其中 $0 \leq n \leq 9$ ，
//若 n 越界，则返回-1，并提示‘第二个参数越界’。例如：4500201 中有 0 共 3 个，编写主函数并调试。

```
#include<stdio.h>
void count(int m,int n){
    int i=m,j,num=0;
    while(i!=0){
        j=i%10;
        i=i/10;
        if(j==n)
            num++;
    }
    printf("数字%d 中的%d 共有： %d 个\n",m,n,num);
}
```



```

int main(){
    int m,n;
    printf("请输入一个正整数：\n");
    scanf("%d",&m);
    printf("请输入要统计的数字：\n");
    scanf("%d",&n);
    if(n>=0&&n<=9){
        count(m,n);
    }else{
        printf("第二个参数越界！\n");
        return -1;
    }
}

```

//建立一个学生在某一个课程到课情况统计程序。功能要求：

//1.可一次性输入所有学生的到课情况，输入学生总人数，该课程总课时，学生学号及到课情况，分为正常，迟到，请假，旷课

//2.可统计某一个学生的到课情况的上课率（包括正常，迟到），旷课率并输出；

//3.可统计所有学生的上课率，旷课率，并输出；

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

```
#define N 20
```

```

struct student{
    char name[20];
    int stuno;
    int zhengchang;
    int chidao;
    int qingjia;
    int kuangke;
}man[N];

```

```

int insert(int n){
    int i=0;
    while(1){
        printf("请输入：name stuno zhengchang chidao qingjia kuangke\n");

```

```

        scanf("%s%d%d%d%d%d",&man[i].name,&man[i].stuno,&man[i].zhengchang,&man[i].chidao,&man[i].qingjia,&man[i].kuangke);

```

```

        i++;
        if(i>=n)
            break;
    }
    return i;
}

```

```

void getPersonInformation(int no,int num){
    int i=0;
    for(i=0;i<num;i++){

```

```

        if(man[i].stuno==no){
            float sum=man[i].zhengchang+man[i].chidao+man[i].qingjia+man[i].kuangke;
            float shangkelv=(man[i].zhengchang+man[i].chidao+man[i].qingjia)/sum;
            float kuangkelv=man[i].kuangke/sum;
            printf("该学生的上课率是： %f,旷课率是:%f\n",shangkelv,kuangkelv);
            return;
        }
    }
}

void getAllInformation(int n){
    int i=0;
    float sum=0;
    float shangke=0,kuangke=0;
    while(i<n){
        sum+=man[i].zhengchang+man[i].chidao+man[i].qingjia+man[i].kuangke;
        shangke+=man[i].zhengchang+man[i].chidao+man[i].qingjia;
        kuangke+=man[i].kuangke;
        i++;
    }
    printf("学生总的上课率： %f,总的旷课率:%f\n",shangke/sum,kuangke/sum);
}

void main(){
    int n=0,no=0;
    printf("请输入学生人数： \n");
    scanf("%d",&n);
    int num=insert(n);
    printf("请输入查询学生的学号： \n");
    scanf("%d",&no);
    getPersonInformation(no,num);
    getAllInformation(num);
}

```

//设计算法并编写程序完成如下功能：

//1) 从键盘输入 100 个整数

//2) 统计这 100 个数中非负个数

//3) 输出这个个数，并以每行输出 10 个数据的要求输出所有的负数（包括 0）

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

```

void main(){
    int a[100];
    int i,j=0,k=0;
    for(i=0;i<100;i++)
        scanf("%d",&a[i]);
    for(i=0;i<100;i++)
        if(a[i]>=0)
            k++;
}

```

```

printf("非负数字个数： %d\n",k);
for(i=0;i<100;i++){
    if(a[i]<=0){
        printf("%d ",a[i]);
        j++;
    }
    if(j%10==0)
        printf("\n");
}
}

```

//编写一个函数，输入一个字符串，分别统计该字符串中出现大的数字字符个数，字母字符个数和其他类型字符个数

```

#include<stdio.h>
#include<string.h>
void main(){
    char str[100];
    int i,len,a=0,b=0,c=0;
    printf("请输入一个字符串： \n");
    scanf("%s",str);
    len=strlen(str);
    for(i=0;i<len;i++){
        if(str[i]>='0'&&str[i]<='9')
            a++;
        else if((str[i]>='a'&&str[i]<='z')||(str[i]>='A'&&str[i]<='Z'))
            b++;
        else
            c++;
    }
    printf("数字字符个数： %d， 字母字符个数： %d， 其他字符个数： %d\n",a,b,c);
}

```

//一个已经排好序的数组，编写程序，要求：输入一个数，按原来排序的规律将它插入合适的位置中，并且输出数组内容。

```

#include<stdio.h>
void insertSort(int a[],int len,int x){
    int i,j;
    while(a[i]<x&&i<len)i++;
    for(j=len+1;j>i;j--)
        a[j]=a[j-1];
    a[i]=x;
    len++;
    printf("插入后的数组为:\n");
    for(i=0;i<len;i++)
        printf("%d,",a[i]);
}

```

```

        printf("\n");
    }

```

//编写函数，函数首部为 void strcat(char *s1,char *s2)，实现将两个字符串合并后存到 s1

```

#include<stdio.h>
void strcat(char *s1,char *s2){
    int i,j;
    for(i=0;s1[i]!='\0';i++);
    for(j=0;s2[j]!='\0';j++)
        s1[i++]=s2[j];
    s1[i]='\0';
    printf("%s",s1);
}

void main(){
    char s1[50],s2[20];
    printf("请输入字符串 s1:\n");
    scanf("%s",s1);
    printf("请输入字符串 s2:\n");
    scanf("%s",s2);

    strcat(s1,s2);
}

```

//编写一个函数，功能是：将字符串 s 中的所有数字字符去掉，保留其余的字符，并且将形成的新字符串存储在原 s 的空间中。

```

#include<stdio.h>
void sub(char *str){
    int i,j;
    for(i=0;str[i]!='\0';i++){
        if(str[i]>='0'&&str[i]<='9'){
            for(j=i;str[j]!='\0';j++)
                str[j]=str[j+1];
            i=i-1;
        }
    }
    printf("去掉数字的字符：%s\n",str);
}

void main(){
    char str[100];
    printf("请输入一个字符串：\n");
    scanf("%s",&str);
    sub(str);
}

```