

# C语言第四周作业

## 程序题

### 闰年判断

【问题描述】3.1.3 从键盘输入年份，如果是闰年输出“XXX年是闰年”，否则输出“XXX年不是闰年”。

【样例输入1】 请输入年份: 2019

【样例输出1】 2019年不是闰年

【样例输入2】 请输入年份: 2016

【样例输出2】 2016年是闰年

【样例说明】输入提示语句中的冒号是中文字符。本题输出结束后没有换行符。

判断任意年份是否为闰年，需要满足以下条件中的任意一个：

- (1) 该年份能被4整除同时不能被 100 整除；
- (2) 该年份能被400整除。

- 这道题并没有更多的细节就是在 `if` 语句条件里面熟练使用 `||`(或者) 和 `&&`(并且)
- 注意在C语言中，表示判断等于的符号是 `==`

### 示例代码

```
#include<stdio.h>
int main()
{
    int year;
    printf("请输入年份:");
    scanf("%d",&year);
    if(year%4==0&&year%100!=0||year%400==0)//一次性写出判断闰年的条件
        printf("%d年是闰年",year);
    else
        printf("%d年不是闰年",year);
    return 0;
}
```

## 分段函数

【问题描述】3.1.1 实现从键盘输入实数x的值，计算y的值。x是实数，y是整数。

$$y = \begin{cases} 1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$$

【输入形式】从键盘输入一个float型实数。

【输出形式】从屏幕上显示运算结果，运算结果是整数。

【样例输入1】 Enter a number:3.5

【样例输出1】 y=1

【样例输入2】 Enter a number:0

【样例输出2】 y=0

【样例输入3】 Enter a number:-5.6

【样例输出3】 y=-1

【样例说明】

- (1) 输入为float型实数，输出为整数，本题输出结束后没有换行符。
- (2) 注意输入提示语句的大小写，冒号为英文符号，后面无空格。
- (3) 等号两边无空格。

- 这里使用3个 `if` 做判断就行了没有必要去纠结是不是使用 `if嵌套`

## 示例代码

```
#include<stdio.h>
int main()
{
    float x;
    printf("Enter a number:");
    scanf("%f",&x);
    if(x>0)
        printf("y=1");
    if(x==0)
        printf("y=0");
    if(x<0)
        printf("y=-1");
    return 0;
}
```

## 大小写转化

【问题描述】3.1.2 实现从键盘读入一个字符，如果该字符为小写字母，则转换成大写字母并输出；如果该字符为大写字母，则转换成小写字母输出；如果为其他字符则原样输出。

【输入形式】从键盘输入一个字符。

【输出形式】输出一个字符。

【样例输入1】 Enter one character:d

【样例输出1】 D

【样例输入2】 Enter one character:B

【样例输出2】 b

【样例输入3】 Enter one character:+

【样例输出3】 +

【样例说明】输入/输出为一个字符，本题输出结束后没有换行符。

- 输入时候一定是使用 `char` 型输入，单个字符建议使用 `getchar()`
- 大小写转换还是根据 `ASCII` 码进行变换，即小写变大写 `-32` , 大写变小写 `+32`
- 写好判断条件就ok了

### 示例代码

```
#include<stdio.h>
int main()
{
    char a;
    printf("Enter one character:");
    scanf("%c",&a);
    //如果为大写字母则输出其对应小写字母
    if(a>='A'&&a<='Z')
        printf("%c",a+32);
    //如果为小写字母则输出其对应大写字母
    else if(a>='a'&&a<='z')
        printf("%c",a-32);
    else
        printf("%c",a);
    return 0;
}
```

## BMI指数计算

### 【问题描述】

BMI指数（即身体质量指数），是世界公认的一种评定肥胖程度的分级方法，它的定义如下：

$$\text{体质指数 (BMI)} = \text{体重 (kg)} \div \text{身高}^2 \text{ (m)}$$

参考判断标准如下：

较轻：体重指数<18

正常：18<=体重指数 <25

超重：25<=体重指数 <28

肥胖：体重指数 >=28

输入体重和身高。要求：

- ① 计算BMI指数；
- ② 根据计算值参照判断标准评定体重情况。

【样例输入】 `please enter your w(kg)and height(m):50 1.6`

【样例输出】 `your BMI is:19.53,normal!`

【样例说明】输出BMI保留两位小数，四级体重等级为 `light`、`normal`、`a little fat`、`fat`。输出结果后不换行。

### 示例代码

```
#include<stdio.h>
int main()
{
    //输入身高和体重 (double) ,输入提示为：“please enter your w(kg)and height(m):”
    double h,w;
    printf("please enter your w(kg)and height(m):");
    scanf("%lf %lf",&w,&h);
    //计算BMI
    double bmi=w/(h*h);
    //输出BMI, “your BMI is:”, 两位小数
    printf("your BMI is:%.2f",bmi);
    //判断四级体质, 四级体重等级为light, normal, a little fat, fat, 判断精度为整数, 并且输出
    if(bmi<18)
        printf(",light!");
    if(bmi>=18.5&&bmi<25)
        printf(",normal!");
    if(bmi>=25&&bmi<28)
        printf(",a little fat!");
    if(bmi>=28)
        printf(",fat!");
    return 0;
}
```

- 这里没有使用嵌套的 `if`，个人还是不喜欢去嵌套 `if` 的

## 逆序输出

### 【问题描述】

给定一个10~1000的正整数（不包含10和1000）。要求：

- ① 求出它是几位数；
- ② 输出每一位数字；
- ③ 判断其逆序后是否仍与原数相同，并输出结果。

### 【样例输入输出】

please enter the number:686

count=3

the digits are 6 8 6

It's a palindrome

【样例说明】686代表输入的数字。如果逆序后与原数不相同，那么输出It's not a palindrome，否则输出It's a palindrome。最后一行输出后不换行。

- 这和之前的内容一样，可以使用字符串进行计算也可以用 `int` 进行计算
- 示例代码1中使用了自定义函数，目前还没有学到

## 示例代码1

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

int Digitscount(int n)
{
    int i=0;
    while(n != 0){
        i++;
        n /= 10;
    }
    return i;
}

int Digitsprint(int n)
{
    int x=Digitscount(n),sum=0;
    char b[5];//这里其实是编译器问题，选择DEV-C++编译器就是char b[x+1]就好，C/C++那个版本太老了，只能b[5]
    b[x]='\0';
    for (int i = x-1;i >=0 ;i--){
        b[i]=n%10+'0';
        n /= 10;
    }
    for (int i = 0; i < x; i++)
    {
        if (i==x-1)
            printf("%c\n",b[i]);
    }
}
```

```

        else
            printf("%c ",b[i]);
        sum += pow((double)10,(double)i)*(b[i]-'0');
    }
    return sum;
}
int main()
{
    int a,c;
    printf("please enter the number:");
    scanf("%d",&a);
    c=Digitscount(a);
    printf("count=%d\n",c);
    printf("the digits are ");
    if(Digitsprint(a)==a)
        printf("It's a palindrome");
    else
        printf("It's not a palindrome");
    return 0;
}

```

我觉得这个代码量其实是有点大的，我还是写一个用整数进行的简单的代码

## 示例代码2

```

#include<stdio.h>
#include<math.h>

int main()
{
    int a,b[6],c = 0,sum =0,n;
    printf("please enter the number:");
    scanf("%d",&a);
    n=a;//这里是用n来计算位数，主要是要对于变量有操作需要保留数据，这个时候用
    函数很不错。
    while(n != 0){
        c++;
        n /= 10;
    }
    printf("count=%d\n",c);
    printf("the digits are ");
    for (int i = c-1;i >= 0 ;i--){
        b[i] = (a%(int)10);
        a /= 10;
    }
    for (int i = 0; i < c; i++)
    {
        if (i==c-1)
            printf("%d\n",b[i]);
        else
            printf("%d ",b[i]);
    }
}

```

```

        sum += pow((double)10,(double)i)*b[i];
    }
    if(sum == a)
        printf("It's a palindrome");
    else
        printf("It's not a palindrome");
    return 0;
}

```

## 简易计算器

【问题描述】编程实现简易的计算器：读入两个整数运算数(data1和data2)及一个运算符(op)，计算表达式data1 op data2的值，其中op可以是+,-,\*,/,%。

【输入形式】

控制台输入运算数和运算符：

1. 首先输入data1;
2. 紧接着输入一个字符作为运算符op，op可以是+,-,\*,/,%。
3. 再输入data2;

【输出形式】控制台输出运算结果。作除法运算时，输出结果以%f的格式输出。

【样例输入】 **23\*5**

【样例输出】 **23\*5=115**

【样例说明】输入中先后输入第一个运算数23，运算符\*，第二个运算数5；要求对23和5进行乘法运算。计算结果为115。输出后不换行。

当运算符为除和取余时，若除数为 0 输出出错信息 **divide error!**

当运算符不合法时，输出出错信息 **op error!**

- 需要注意的输出问题
  - % 本身在 `printf()` 中就作为一个特殊符号会被编译掉，所以我们想要输出一个 % 只能通过语句 `printf("%%")` 来实现。
  - 在运算中除法可能产生小数，所以需要我们去手动转换数据类型
- 使用 `switch` 语句在这个示例中显然更加的高效便捷。
  - 注意在 `case` 后的符号使用 `' '` 单引号括起来表示单个字符
  - 每一个 `case` 结束以后都需要配上一个 `break`（就本题目而言是这样的，不然程序会继续判断下面情况浪费时间）

## 示例代码

```
//编程实现简易的计算器：读入两个整数运算数(data1和data2)及一个运算符(op)，计算表达式data1 op data2的值，
//其中op可以是+,-,*,/,%。
#include<stdio.h>
int main()
{
    int a,b;
    char op;
    scanf("%d%c%d",&a,&op,&b);
    switch (op)
    {
        case '+':
            printf("%d+%d=%d",a,b,a+b);
            break;
        case '-':
            printf("%d-%d=%d",a,b,a-b);
            break;
        case '*':
            printf("%d*%d=%d",a,b,a*b);
            break;
        case '/':
            if(b==0)
                printf("divide error!");
            else
                printf("%d/%d=%f",a,b,(float)a/(float)b);
            break;
        case '%':
            if(b==0)
                printf("divide error!");
            else
                printf("%d%%%d=%d",a,b,a%b);
            break;
        default:
            printf("op error!");
            break;
    }
    return 0;
}
```



## 出租车计费器

【问题描述】上海市出租车日间收费标准如下表所示，输入行驶里程（精确到0.1km）。要求：计算并输出乘客应支付的车费（元），结果四舍五入，保留到元。

公里数	日间 (5: 00~23: 00)
0~3km	14元
3~15km	2.5元/km
15km以上	3.6元/km

【样例输入】 `enter the km:21.5`

【样例输出】 `the distance is 21.5 km,the fee is 67 yuan`

【样例说明】输出后不换行

- 分段计算就行，俩 `if` 解决战斗

### 示例代码

```
#include<stdio.h>
int main()
{
    float l,sum=14.0;//只要上车就14元就不用计算了
    printf("enter the km:");
    scanf("%f",&l);
    if (l>=3)
        sum += (l-3)*2.5;//超出3km的计算
    if (l>=15)
        sum = 44.0 +(l-15)*3.6;//超过15km，直接算出前15公里44元，只用加上多出部分
    printf("the distance is %.1f km,the fee is %d yuan",l,
(int)sum);
    return 0;
}
```