

# 自定义函数讲解

---

## 人员

韩承睿、刘嘉航、辛帅辰、崔吉诺、夏硕承、秦显森、谢亚锴、王静嘉、牛同泽 到课

高健桓、李翰如、方俊喆、刘祺 线上

## 作业检查

韩承睿 已完成

牟茗 已完成

刘嘉航 已完成

辛帅辰 已完成

高健桓 已完成

李翰如 已完成

方俊喆 已完成

崔吉诺 已完成

刘祺 已完成

夏硕承 已完成

秦显森 上周请假

齐振玮 未打卡

谢亚锴 已完成

王静嘉 未打卡

牛同泽 上周请假

徐浩然 已完成

## 作业

必做：noi 1.13 18:Tomorrow never knows? 和 noi 1.13 25:计算两个日期之间的天数

选做：noi 1.13 05:素数回文数的个数

## 课堂表现

这节课同学们纪律都保持非常好，听讲也很认真，课上内容有些复杂，同学们要课下认真进行复习。

## 课堂内容

## 自定义函数

自定义函数的构成：

```
返回值类型 函数名(参数)
{
    函数内容;
    return ...; // 注意如果函数是void类型，那么不需要返回值。
}
```

调用：直接函数名（实参）；有返回值直接找响应变量接收

其实自定义函数和咱们用的`pow`，`max`这些函数一样，他们本身就是被写好的自定义函数

没有返回值使用`void`即可

想要传实参的话：可以在参数前加 `&` 符号

### noi 1.13 18:Tomorrow never knows?

函数解释：

- `bool isRunNian(int y)`: 判断 `y` 年是不是闰年
- `bool isYueDi(int y, int m, int d)`: 判断 `y`年`m`月`d`日 这一天是不是月底
- `bool isNianDi(int m, int d)`: 判断是不是年底
- `date nextDay(int y, int m, int d)`: 求 `y`年`m`月`d`日 的下一天

```
#include <iostream>
#include <cstdio>

using namespace std;

struct date {
    int y, m, d;
};

int a[] = {0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

bool isRunNian(int y) {
    if (y%400==0 || (y%4==0&& y%100!=0)) {
        return true;
    }
    return false;
}

bool isYueDi(int y, int m, int d) {
    if (isRunNian(y) && m==2) {
        if (d==29) {
            return true;
        }
    }
}
```

```
    }
    return false;
}

if (d == a[m]) {
    return true;
}
return false;
}

bool isNianDi(int m, int d) {
    if (m==12 && d==31) {
        return true;
    }
    return false;
}

date nextDay(int y, int m, int d) {
    if (isNianDi(m, d)) {
        return {y+1, 1, 1};
    }
    if (isYueDi(y, m, d)) {
        return {y, m+1, 1};
    }
    return {y, m, d+1};
}

int main()
{
    int y, m, d;
    scanf("%d-%d-%d",&y,&m,&d);
    date t = nextDay(y, m, d);
    printf("%d-%02d-%02d\n", t.y, t.m, t.d);
    return 0;
}
```

### noi 1.13 25:计算两个日期之间的天数

函数解释:

- bool isRun(int y): 判断 y 年是不是闰年
- int yue(int y, int i): 求 y 年 i 月有多少天
- int jisuan(int y, int m, int d): 计算从 1.1.1 到 y.m.d 一共经过了多少天

```
#include <iostream>

using namespace std;

bool isRun(int y) {
    if (y%400==0 || (y%4==0&& y%100!=0)) {
```

```
        return true;
    }
    return false;
}

int yue(int y, int i) {
    if (i==1 || i==3 || i==5 || i==7 || i==8 || i==10 || i==12) {
        return 31;
    }
    if (i==4 || i==6 || i==9 || i==11) {
        return 30;
    }
    if (isRun(y)) {
        return 29;
    }
    return 28;
}

int jisuan(int y, int m, int d) {
    int sum = 0;
    for (int i = 1; i <= y-1; i++) {
        if (isRun(i)) {
            sum += 366;
        }
        else {
            sum += 365;
        }
    }

    for (int i = 1; i <= m-1; i++) {
        sum += yue(y, i);
    }

    sum += d;
    return sum;
}

int main() {
    int y1, m1, d1, y2, m2, d2;
    cin >> y1 >> m1 >> d1 >> y2 >> m2 >> d2;
    cout << jisuan(y2,m2,d2) - jisuan(y1,m1,d1) << endl;
    return 0;
}
```