



**《程序设计原理》实验报告**

**——时间计算器**

**学 号 3022244290**

**姓 名 陈秋澄**

**学 院 智能与计算学部**

**年 级 22级**

**任课教师 李雪威**

**2022年12月10日**

# 实验目的

学习time（）函数相关知识，设计时间计算器，实现“自某个时刻开始经过t秒之后的时刻”。

# 二．实验内容

1. **设计思路**

先把时间拆成天数和剩余的秒数；

然后用秒数算出答案当天的时分秒；

推到初始日期是初始年份的第几天；然后算出从初始年份的第一天一直到答案的当天一共有多少天；然后算过了多少年，剩余多少天；剩余的天数就是新的年份的日期。

1. **代码实现**

#include <iostream>

#include <string>

using namespace std;

const int day\_tab[2][12] = {{31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31},

{31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31}};

const string month\_name[13] = {" ", "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sept", "Oct", "Nov", "Dec"};//因数组编号从0开始，故第一位用“ ”来占位

const string day\_name[13] = {"th", "st", "nd", "rd"};

int runnian(int y)

{

return (y % 4 == 0 && y % 100 != 0 || y % 400 == 0);

}//判断闰年

int month\_to\_day(int leap, int month, int day)

{

for (int i = month - 2; i >= 0; i--)

day += day\_tab[leap][i];

return day;

}//月换算成天

int year\_to\_day(int year)

{

return 365 + runnian(year);

}//年换算成天

void times(int data)

{

if (data <= 9)

cout << 0;

}//注意输出格式

int main()

{

int n,d;

long long y0, m0, d0, h0, mi0, s0;

long long year\_new, month\_new, day\_new, hour\_new, minute\_new, second\_new;

long long s1, second\_add, day\_add;

cin >> n;

while (n--)

{

cin >> y0 >> m0 >> d0 >> h0 >> mi0 >> s0 >> s1;

int month[2][13]={{0,31,28,31,30,31,30,31,31,30,31,30,31},

{0,31,29,31,30,31,30,31,31,30,31,30,31}};

int year0=y0,month0=m0,day0=d0,s=h0,f=mi0,a=s1;

int hh=0,mm=0,ss=0; //秒转成的时分秒

long long b=0; //输入的秒数

int dd=0,sum=0;

if((y0%4==0&&y0%100!=0)||(y0%400==0)){

for(int i=0;i<m0;i++){

sum+=month[1][i];

}

}else

{

for(int i=0;i<m0;i++){

sum+=month[0][i];

}

}

sum+=d; //输入的日期是初始年份的第几天

hh=b/3600;

mm=b/60%60;

ss=b%60;

if(ss+a>=60){

a=ss+a-60; //新秒

f=f+mm+1; //新分

if(f>=60){

f=f-60;

s=s+hh+1;

if(s>=24){

dd=s/24;

s=s%24;

}

}

else{

s=s+hh;

if(s>=24){

dd=s/24;

s=s%24;

}

}

}

else{

a+=ss;

f+=mm;

if(f>=60){

f=f-60;

s=s+hh+1;

if(s>=24){

dd=s/24;

s=s%24;

}

}

else{

s=s+hh;

if(s>=24){

dd=s/24;

s=s%24;

}

}

}

day\_add = s1 / 86400;

long long second\_add = h0 \* 3600 + mi0 \* 60 + s0 + s1 % 86400;

//秒数求和

if (second\_add >= 86400)

{

day\_add += second\_add / 86400;

second\_add %= 86400;

}

hour\_new = second\_add / 3600;

second\_add %= 3600;

minute\_new = second\_add / 60;

second\_add %= 60;

second\_new = second\_add;

day\_add += month\_to\_day(runnian(y0), m0, d0);

long long y = y0;

while (day\_add > year\_to\_day(y))

{

day\_add -= year\_to\_day(y);

y++;

}

while (day\_add <= 0)

{

y--;

day\_add += year\_to\_day(y);

}

long long m = 1;

long long d1;

while ((d1 = day\_add - day\_tab[runnian(y)][m - 1]) > 0)

{

day\_add = d1;

m++;

}

year\_new = y;

month\_new = m;

day\_new = day\_add;

cout << month\_name[month\_new] << " " << day\_new;

if (day\_new % 10 > 0 && day\_new % 10 <= 3 && day\_new / 10 != 1)

cout << day\_name[day\_new % 10] << " ";

else

cout << day\_name[0] << " ";

times(hour\_new);

cout << hour\_new << ":";

times(minute\_new);

cout << minute\_new << ":";

times(second\_new);

cout << second\_new << " " << year\_new <<endl;

}

}

1. **结果展示**

# 三．实验中遇到的问题和解决办法

问题：格式输出存在偏差。

解决办法：创建简易函数（如下），数值若小于10，十位自动补零。