

C++方向编程题答案

第六周

day32

1、题目ID: 858 淘宝网店

链接: <https://www.nowcoder.com/questionTerminal/754921e9c98b43d1b2d70c227b844101>

【题目解析】:

这是一个变相的日期计算器。只不过2、3、5、7、11月算1天，其他7个月算2天。

【解题思路】:

既然是一个变相的日期计算器，那就写一个日期计算器，然后加以修改即可。那么，日期计算器怎么写呢？

日期计算器的话，我们将会把日期计算分为三个部分：第一个不足一年的年份，最后一个不足一年的年份，和中间的足年年份。足年年份我们只需要判断闰年后加365或366就行了。不足年，我们就要求出这个日期是这一年的第几天。假设要求的是1994年5月27日到2003年4月29日，那么，我们就要先求出5月27日是这一年的第几天，然后判断1994年不是闰年，不是，所以用365减去这个天数，就得到结果了。本题中第一天也要算，所以还要加上这一天。然后再算出4月29日是2003年的第几天，就可以解决问题了。所以，我们需要一个函数，功能是给出一个年月日，求出这是这一年的第几天。

这些功能全部实现后，再去改造使得1、4、6、8、9、10、12月的天数翻倍，那么程序就全部完成了。

【示例代码】:

```
#include <stdio>
#include <cmath>
#include <iostream>

//闰年判断函数
inline int leap_year(int year)
{
    return year % 400 == 0 || (year % 4 == 0 && year % 100 != 0);
}

//足年天数
inline int profit_of_year(int year)
{
    return 2 * 31
        + 1 * 28
        + 1 * 31
        + 2 * 30
        + 1 * 31
        + 2 * 30
        + 1 * 31
        + 2 * 31
        + 2 * 30
```

```

        + 2 * 31
        + 1 * 30
        + 2 * 31
        + leap_year(year);
    }

    //判断这个月份是不是质数月
    inline bool prime(int n)
    {
        return n == 2 || n == 3 || n == 5 || n == 7 || n == 11;
    }

    //求出一个日子是这一年的第几天
    int profit_of_this_year(int year, int month, int day)
    {
        if(!prime(month))
        {
            day *= 2;
        }

        while(--month)
        {
            switch(month)
            {
                case 1:
                case 8:
                case 10:
                case 12:
                    day += 62;
                    break;
                case 3:
                case 5:
                case 7:
                    day += 31;
                    break;
                case 4:
                case 6:
                case 9:
                    day += 60;
                    break;
                case 11:
                    day += 30;
                    break;
                case 2:
                    day += 28 + leap_year(year);
                    break;
                default:;
            }
        }
        return day;
    }

    int main()

```

```

{
    int year1, month1, day1, year2, month2, day2;
    int count_profit = 0;

    while (std::cin >> year1 >> month1 >> day1 >> year2 >> month2 >> day2)
    {
        count_profit = 0;

        count_profit += profit_of_year(year1) -
            profit_of_this_year(year1, month1, day1 - 1);
        //这里的day1 - 1虽然有可能出现0日，但是实际2月0日就相当于1月31日，所以不影响结果。
        count_profit += profit_of_this_year(year2, month2, day2);
        if(year1 == year2) //避免起点和终点是同一年，如果是同一年，要减掉这一年的天数。
        {
            count_profit -= profit_of_year(year1);
        }

        for(int i = year1 + 1; i < year2; i++) //中间足年每一年的天数
        {
            count_profit += profit_of_year(i);
        }

        std::cout << count_profit << std::endl;
    }

    return 0;
}

```

2、题目ID: 750 斐波那契凤尾

链接: <https://www.nowcoder.com/questionTerminal/c0a4b917a15f40a49ca10532ab9019fb>

【题目解析】:

题目要求输出斐波那契数列的第n项，最容易写的方法就是用循环求出每一项了。而它要求的是后六位，那么我们只需要存储后六位就行了。

【解题思路】:

先求斐波那契数列在100000以内的每一项的后六位，然后需要的时候直接输出数组里的对应值即可。

以下代码用通常的循环法解决。

【示例代码】:

```

#include <iostream>
#include <cstdio>

int main()
{
    int fib[100001];
    fib[0] = 1;
    fib[1] = 1;

```

```
//提前计算斐波那契数列，只保留后6位
for (int i = 2; i <= 100000; ++i)
{
    fib[i] = fib[i - 1] + fib[i - 2];
    fib[i] = fib[i] % 1000000; //由于是相加，所以只要后六位不影响下一个数的结果
}

//这里利用了oj平台计时从开始读取数据的trick，这个技巧可以用在很多题目里
int n;
while (std::cin >> n)
{
    if (n < 29)
    {
        //斐波那契数列小于6位
        printf("%d\n", fib[n]);
    }
    else
    {
        printf("%06d\n", fib[n]);
    }
}

return 0;
}
```