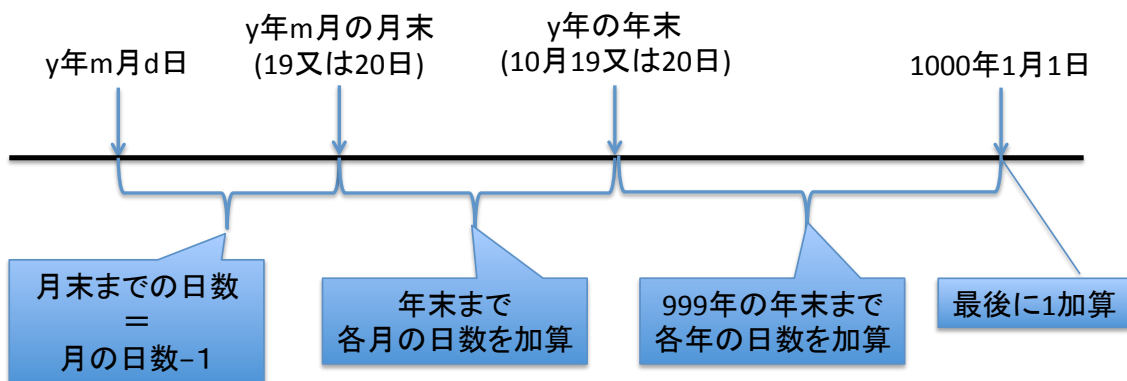


ACM ICPC 2012 [国内予選問題 A](#) ミレニアム

- 与えられた y, m, d から、直接解答を与える計算式を考えても良いが、煩雑になりそうなので、
 1. 月末までの日数を求める
 2. これに、年末まで各月の日数を加算する
 3. さらに、翌年から 999 年まで各年の日数を加算する
 4. これに 1 加えると、1000 年 1 月 1 日までの日数となるように、計算するものとする。



【アルゴリズム 1】

関数 $ndm(int\ y, int\ m)$ を y 年 m 月の日数を返す関数とすると全体の流れは

1. y 年 m 月 d 日から y 年 m 月の月末までの日数 $days$ を求める
 - $days = ndm(y, m) - d;$
2. y 年 $(m+1)$ 月から y 年の年末 (10 月) までの日数を月毎に $days$ に加算
 - $for(i=m+1; i<=10; i++)\ days += ndm(y, i);$
3. $(y+1)$ 年から 999 年までの日数を年ごとに $days$ に加算
3 で割り切れる年は 200 日/年、3 で割り切れない年は 195 日/年なので
 - $for(i=y+1; i<=999; i++)\ days += (i \% 3) ? 195 : 200;$
4. 求めるのは、1000 年 1 月 1 日までの日数なので、 $days+1$ が答えとなる

3 で割り切れる年は全て大の月であり、また、奇数月も大の月であるので、

```
int ndm(int y, int m){ if(y % 3 || m % 2) return 20; else return 19;}
```

で求めることができる。

プログラム例 1

```
// ICPC 2012 国内予選問題 A
//
//  Filename = pal.c
//  Compile : cc pal.c
//  Execution : ./a.out < A0 > A0.result
//  Check : diff A0.ans A0.result

#include <stdio.h>

// 大の月なら 20、小の月なら 19 を返す関数 (y = 年、 m = 月)
int ndm(int y, int m)
{
    // 3 で割り切れる年は全て大の月。また、奇数月も大の月。
    if(y % 3 == 0 || m % 2) return 20;
    else return 19;
}

// 1000 年 1 月 1 日までの日数を求める
int calc(int y, int m, int d)
{
    int days; // 日数計算に用いる変数
    int i;
    days = ndm(y, m) - d; // まず、月末までの日数を数える
    for(i=m+1; i<=10; i++) // 年末までの日数を数える
        days += ndm(y, i);
    for(i=y+1; i<=999; i++) // 999 年末までの日数を数える
        days += (i % 3)?195:200; // 3 で割り切れない年は 195 日、割り切れる年は 200 日
    days++; // 求めるのは 1000 年 1 月 1 日までの日数なので 1 増やす
    return days;
}

int main()
{
    int n, y, m, d, i;
    scanf("%d", &n); // n = データセット数
    for(i=0; i<n; i++){
        scanf("%d %d %d", &y, &m, &d); // 年月日の読み込み
        printf("%d\n", calc(y, m, d)); // 日数を求めてプリント
    }
}
```

【アルゴリズム 2】

アルゴリズム 1 のステップ 2 とステップ 3 を、次のステップ 2 とステップ 3 のように各々直接計算する。

2. y 年末までの日数は、 y が 3 で割り切れるなら $(10-m)*20$ を加算し、そうでなければ $(10-m)*19.5$ の小数点以下を切り捨てた値を加算すれば良い¹。
3. 999 年末までの日数は、まず $(999-y)/3*590$ を加算し²、 $(999-y)$ を 3 で割った余りが 1 なら 200 を更に加算し、余りが 2 なら $395=200+195$ を更に加算すれば良い³。

プログラム例 2

```
// ICPC 2012 国内予選問題 A
//
//  Filename = pa2.c
//  Compile  :  cc pa2.c
//  Execution :  ./a.out < A0 > A0.result
//  Check    :  diff A0.ans A0.result

#include <stdio.h>

// 大の月なら 20、小の月なら 19 を返す関数 (y = 年、 m = 月)
int ndm(int y, int m)
{
    // 3 で割り切れる年は全て大の月。また、奇数月は大の月
    if(y % 3 == 0 || m % 2) return 20;
    else return 19;
}

// 1000 年 1 月 1 日までの日数を求める
int calc(int y, int m, int d)
```

¹ 整数型の変数に代入すれば、小数点以下は切り捨てられる。例えば、 $10-m$ が 1 の場合、10 月の日数を加算すれば良いので、 y が 3 で割り切れるなら 20 を加算し、割り切れないなら $19=(19.5$ の小数点以下切り捨て)を加算すれば良い。 $10-m$ が 2 の場合、10 月と 9 月の日数を加算すれば良いので、 y が 3 で割り切れるなら $20*2$ を加算し、割り切れないなら $39=19+20=19.5*2$ を加算すれば良い。

² 3 年間の日数は、 $200+195+195=590$ である。

³ 例えば、 $999-y$ が 4 の場合、999 年、998 年、997 年、996 年の 4 年間の日数を加算すれば良い。このためには、999 年～997 年の 3 年間の日数 590 日に加えて、996 年（3 で割り切れる年）の日数 200 日を加えれば良い。また、 $999-y$ が 5 の場合、更に 995 年（3 で割り切れない年）の日数 195 日を加えれば良い。

```

{
    int days;                // 日数計算に用いる変数
    int i;
    days = ndm(y, m) -d;    // まず、月末までの日数を数える
    // 年末までの日数を数える
    if(y % 3)
        days += 19.5*(10-m); // days は整数型変数なので小数点以下は切り捨てられる
    else
        days += (10-m)*20;
    // 999 年末までの日数を数える
    days += (999-y)/3*(590); // 3 年分の日数は 200+195+195
    if((999-y)%3 == 1) days += 200; // 残り 1 年ならその日数は 200
    else if((999-y)%3 == 2) days += 395; // 残り 2 年ならその日数は 200+195
    days++; // 求めるのは 1000 年 1 月 1 日までの日数なので 1 増やす
    return days;
}

int main()
{
    int n, y, m, d, i;
    scanf("%d", &n); // n = データセット数
    for(i=0; i<n; i++){
        scanf("%d %d %d", &y, &m, &d); // 年月日の読み込み
        printf("%d¥n", calc(y, m, d)); // 日数を求めてプリント
    }
}

```