달력 출력하기(과제11번)

문제 해설

20243108 장민주

문제 30

달력 출력하기

그레고리언 방법으로 달력을 제작할 경우 주어진 달(몇년 몇월)에 해당하는 달력을 출력하는 프로그램을 작성하시오. 그레고리언 방법이 제정된 1582년 1월 1일은 금요일이라고 가정한다.

입력

입력 파일의 이름은 "input.txt" 이다. 입력은 t개의 테스트 케이스로 주어진다. 입력 파일의 첫 번째 중에 테스트 케이스의 개수를 나타내는 경주 t가 주어지다. 두 번째 공부터 t개의 중에는 한 줄에 한 개의 테스트 케이스에 해당하는 두 개의 경수 I/이 10 의원단다. 입력되는 첫 번째 경수 I/이 1582 ≤ I/≤ 99999)는 번도를 나타내고, 두 번째 경수 I/이 1≤ I/≤ I/2)은 월을 나타낸다. 두 전수 시어에는 한 개의 공백이 있으며 공부되 테이터가 입력되는 경수는 없다

출력

총액은 표준층액(standard output)을 사용한다. 업력되는 테스트 케이스의 순시대로 다음 중에 이어 사 대 테스트의 결과를 충득한다. 각 테스트 케이스에 해당하는 충액의 첫 출매는 업명되는 달의 반도와 단을 나타내는 청수를 입력 참으로 유입하게 충액한다. 그 다음 충부터는 한 중에 한 주에 해당하는 날부를 일요일부터 도요일까지 자재대로 출력한다 (단. 첫 중에서 주어진 단의 첫 배 날의 요일이 일요일이 아닌 경우에는 첫째 날의 요일 전까지 해당되는 요일에 모두 0 음 출택한다. 또한 마지막 날의요일이 토요일이 아닌 경우에는 바지막 날의요일 다음차려 도요일까지 자꾸 오루 0 음 충액한다. 따라서, 한 주에 관한 데이터를 충ዛ하는 한 개의 중에는 모두 7 개의 장수가 충색하다.

- 1. 그 달이 며칠까지 있는지 -> 윤년 계산 필요
- 2. 그 달의 1일이 무슨 요일 인지
- -> zeller 공식 이용

```
#include <iostream>
                                                                                     ifstream inputFile( s: "input.txt");
                                                                                     inputFile >> t:
int calculateEndDay(int year, int month) {
 if (month == 2) {
   if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) return 29;
                                                                                       inputFile >> Y >> M:
 } else if (month == 4 || month == 6 || month == 9 || month == 11) return 30;
                                                                                       int startDavOfWeek = calculateDavOfWeek( year: Y, month: M, dav: 1):
int calculateDayOfWeek int year, int month, int day) {
                                                                                       int endDay = calculateEndDay(year: Y, month: M);
 if (month == 1 || month == 2) {
                                                                                       int Day = 1;
                                                                                         for (int j = 0; j < 7; j++) {
 int k = year % 100;
                                                                                           if (--startDavOfWeek >= 0) cout << 0:
                                                                                           else if (Day > endDay) cout << 0;
                                                                                           else cout << Day++:
                                                                                         if (Day > endDay) break:
```

1. 그 달이 며칠까지 있는지 계산

```
int calculateEndDay(int year, int month) {
  if (month == 2) {
    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) return 29;
    else return 28;
} else if (month == 4 || month == 6 || month == 9 || month == 11) return 30;
  return 31;
}
```

```
int calculateDayOfWeek(int year, int month, int day) {
 if (month == 1 || month == 2) {
   vear--;
   month += 12;
 int k = year % 100;
 int j = year / 100;
 int result = (day + 13*(month+1)/5 + k + k/4 + j/4 + 5*j) % 7;
 return (result+6)%7;
```

<Zeller의 공식>

```
• h = (d + 13(m + 1)/5 + K + K/4 + J/4 + 5J) \mod 7
```

- <u>h: 요일 결과</u> (0=토요일, 1=일요일, ..., 6=금요일)
- <u>d: 일 (day)</u>
- m: 월 (month) → 3=3월, ..., 12=12월, 1=1월, 2=2월은 작년의 13월, 14월로 간주
- <u>K: year % 100</u> (세기 안의 연도, 즉 연도의 뒷 두 자리)
- <u>J: year / 100</u> (세기, 즉 연도의 앞 두 자리)

```
int calculateDayOfWeek(int year, int month, int day) {
 if (month == 1 || month == 2) {
   vear--;
   month += 12;
 int k = year % 100;
                                          Zeller 공식
 int j = year / 100;
 // 0 = 토요일, 1 = 일요일, ..., 6 = 금요일
 <u>int result = (day + 13*(month+1)/5 + k + k/4 + j/4 + 5*j) % 7;</u>
 return (result+6)%7;
```

```
int calculateDayOfWeek(int year, int month, int day) {
 if (month == 1 || month == 2) {
   vear--;
   month += 12;
 int k = year % 100;
 int j = year / 100;
  int result = (day + 13*(month+1)/5 + k + k/4 + j/4 + 5*j) % 7;
  return (result+6)%7;
```

```
ifstream inputFile( s: "input.txt");
                                                                            inputFile >> t;
int calculateEndDay(int year, int month) {
                                                                            for (int i = 0; i < t; i++) {
   if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) return 29;
                                                                              int Y.M:
                                                                              int startDayOfWeek = calculateDayOfWeek( year: Y, month: M, day: 1);
int calculateDayOfWeek(int year, int month, int day) {
                                                                              int endDay = calculateEndDay(year: Y, month: M);
 if (month == 1 || month == 2) {
                                                                              int Day = 1;
 int k = year % 100;
                                                                                  if (--startDayOfWeek >= 0) cout << 0;
                                                                                  else if (Day > endDay) cout << 0:
                                                                                  else cout << Dav++:
                                                                                if (Day > endDay) break;
```

문제 30

달력 출력하기

그레고리언 방법으로 달력을 제작할 경우 주어진 달(몇년 몇월)에 해당하는 달력을 출력하는 프로그 탭을 작성하시오. 그레고리언 방법이 제정된 1582년 1월 1일은 금요일이라고 가정한다.

입력

입력 파일의 이용은 "input.txt" 이다. 입력은 / 개의 테스트 케이스로 주어진다. 입력 파일의 첫 번째 중에 테스트 케이스의 계수를 나타내는 경수 /가 주어진다. 두 번째 중부터 / 개의 중에는 현 중에 한 개의 테스트 케이스에 해당하는 두 개의 정수 》 # 사이 인터린다. 입력된다. 입력자는 첫 번째 경수 》 [1582 ≤ 1'≤ 99999])는 번도를 나타내고, 두 번째 경수 》 (1 ≤ 1/5 2)은 월을 나타낸다. 두 경수 사이에는 한 개의 곳뿐이 있으며 경우된 데이터가 입력되는 경우는 입다.

출력

출락은 표준출락(standard output)을 사용한다. 임력되는 테스트 케이스의 순서대도 다운 중에 이어 사 각 테스트의 경과를 슬리한다. 각 테스트 케이스에 해당하는 슬리의 첫 중에는 입력되는 단의 선도와 당은 나타내는 경수를 입력 행식과 동일한지 출택한다. 그 다운 중부터는 한 중에 한 주에 해당하는 날짜를 일요일부터 도요일까지 자레대로 출택한다. 전, 첫 중에서 주어진 당의 첫 제 날의 요일이 일요일이 아닌 경우에는 첫째 날의 요일 전까지 해당되는 요일에 모두 0 을 출택한다. 또한 마지막 날짜의 요일이 도요일이 아닌 경우에는 마지막 날의 요일 다음부터 도요일까지 모두 0 을 출택한다. 또한 마지막 날짜의 요일이 도요일이 아닌 경우에는 마지막 날의 요일 다음부터 도요일까지 모두 0 을 출택한다. 따라서, 한 주에 관한 테이터를 출력하는 한 개의 중에는 모두 7 개의 경수가 출력된다.

```
int main() {
 ifstream inputFile( s: "input.txt");
 inputFile >> t:
 for (int i = 0; i < t; i++) {
   int Y.M:
   int startDayOfWeek = calculateDayOfWeek( year: Y, month: M, day: 1);
   int endDay = calculateEndDay(year: Y, month: M);
   int Day = 1;
                                   일주일 출력
     for (int i = 0; i < 7; i++) {
       if (--startDayOfWeek >= 0) cout << 0;
       else if (Dav > endDav) cout << 0:
       else cout << Dav++:
     cout << "\n";
     if (Day > endDay) break:
```

```
입력
                                         출력
                                         1582 1
1582 1
                                         0000012
2003 9
                                         3 4 5 6 7 8 9
99999 12
                                         10 11 12 13 14 15 16
                                         17 18 19 20 21 22 23
                                         24 25 26 27 28 29 30
                                         31 0 0 0 0 0 0 0
                                         2003 9
                                         0 1 2 3 4 5 6
                                         7 8 9 10 11 12 13
                                         14 15 16 17 18 19 20
                                         21 22 23 24 25 26 27
                                         28 29 30 0 0 0 0
                                         99999 12
                                         0001234
                                         5 6 7 8 9 10 11
                                         12 13 14 15 16 17 18
                                         19 20 21 22 23 24 25
                                         26 27 28 29 30 31 0
```

```
- startDayOfWeek = 3- endDay = 31
```

```
for (int i = 0; i < t; i++) {
 int Y.M:
 int startDayOfWeek = calculateDayOfWeek( year: Y, month: M, day: 1);
 int endDay = calculateEndDay(year: Y, month: M);
 int Day = 1;
                                  일주일 출력
   for (int j = 0; j < 7; j++) {
     if (--startDayOfWeek >= 0) cout << 0;
      else if (Dav > endDav) cout << 0:
      else cout << Dav++:
   cout << "\n";
   if (Day > endDay) break;
```

ifstream inputFile(s: "input.txt");

int main() {