package Tests;

import java.util.Scanner;

//input: 2013

//output: tuesday - january 1

//Su M T W Th F Sa

//inquiry:

//How many specific day are there in a specific month?

//How many specific weeks are there in a particular quarter? (three months)

//How many specific day are there in a semi year? (six months)

//How many complete weeks are there in a year?

public class Tests {

public static int spDayInMonth = 0;

public static int weeksInQuarter = 0;

public static int spDayInSemiYear = 0;

public static int cmpWeeks = 0;

public static int firstDayy = 0;

public static String days[] = {"S", "M", "T", "W", "Th", "F", "Sa"};

public static String months[] = {"JANUARY", "FEBRUARY", "MARCH", "APRIL", "MAY", "JUNE", "JULY", "AUGUST", "SEPTEMBER", "OCTOBER", "NOVEMBER", "DECEMBER"};

public static int max[] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a year: ");

int year = sc.nextInt();

sc.nextLine();

System.out.print("What is the first day of January? ");

String firstDay = sc.nextLine();

firstDayy = getDay(firstDay);

if (isLeap(year)) {

max[1] = 29;

}

//Question number one

System.out.print("\n1. HOW MANY SPECIFIC DAY ARE THERE IN A SPECIFIC MONTH?"

+ "\n - What is the specific day? ");

String q1Day = sc.nextLine();

System.out.print(" - What is the specific month? ");

String q1Month = sc.nextLine();

question1(year, q1Day, q1Month);

System.out.println("There are " + spDayInMonth + " " + q1Day + "s on " + q1Month+".");

//Question number two

System.out.print("\n2. HOW MANY WEEKS ARE THERE IN A SPECIFIC QUARTER?"

+ "\n - What is the particular quarter? [1,2,3,4]: ");

int q2Quarter = sc.nextInt();

String quarter = "";

switch(q2Quarter){

case 1:

quarter = "first";

break;

case 2:

quarter = "second";

break;

case 3:

quarter = "third";

break;

case 4:

quarter = "fourth";

break;

default:

System.out.print("ERROR");

break;

}

question2(year, q2Quarter);

System.out.println("There are "+weeksInQuarter+" weeks in the "+quarter+" quarter of "+year+".");

//Question number three

sc.nextLine();

System.out.print("\n3. HOW MANY SPECIFIC DAY ARE THERE IN A SEMI-YEAR?"

+ "\n - What is the specific day? ");

String q3Day = sc.nextLine();

System.out.print(" - What is the specific semi year? [1,2]: ");

int q3Semi = sc.nextInt();

question3(year, q3Day, q3Semi);

String semi = "first";

if(q3Semi == 2){

semi = "second";

}

System.out.println("There are " + spDayInSemiYear + " " + q3Day + "s on the " + semi +" half of "+year+".");

//Question number four

System.out.println("\n4. HOW MANY COMPLETE WEEKS ARE THERE IN "+year+"?");

question4(year);

System.out.println("There are "+cmpWeeks+" complete weeks in "+year+".");

System.out.println("========================================================");

printCalendar(year, firstDay);

}

static int dayOfJanOne(int year) {

int day = 0;

int yr = year - 1;

day = (1 + (5 \* (yr % 4)) + (4 \* (yr % 100)) + (6 \* (yr % 400))) % 7;

return day;

}

static boolean isLeap(int year) {

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

}

static void printCalendar(int year, String firstDay) {

int CurrentDayOfWeek = firstDayy;

int day = 1;

System.out.println();

for (int m = 0; m < months.length; m++) {

System.out.println(months[m] + " " + year);

for (int i = 0; i < days.length; i++) {

System.out.printf("%3s", days[i]);

}

System.out.println();

if(CurrentDayOfWeek == 7){

CurrentDayOfWeek = 0;

}

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

System.out.printf("%3s", "");

}

while (day <= max[m]) {

while (CurrentDayOfWeek < days.length && day <= max[m]) {

System.out.printf("%3s", day);

day++;

CurrentDayOfWeek++;

}

if (day-1 != max[m]) {

CurrentDayOfWeek = 0;

}

System.out.println();

}

System.out.println();

day = 1;

}

}

static void question1(int year, String q1Day, String q1Month) {

int q1D = getDay(q1Day);

int q1M = getMonth(q1Month);

int CurrentDayOfWeek = firstDayy;

int day = 1;

for (int m = 0; m < months.length; m++) {

if(CurrentDayOfWeek == 7){

CurrentDayOfWeek = 0;

}

while (day <= max[m]) {

while (CurrentDayOfWeek < days.length && day <= max[m]) {

if (m == q1M && CurrentDayOfWeek == q1D) {

spDayInMonth++;

}

day++;

CurrentDayOfWeek++;

}

if (day - 1 != max[m]) {

CurrentDayOfWeek = 0;

}

}

day = 1;

}

}

static void question2(int year, int q2Quarter){

int startMonth = 0, endMonth = 0;

switch(q2Quarter){

case 1:

startMonth = 0;

endMonth = 2;

break;

case 2:

startMonth = 3;

endMonth = 5;

break;

case 3:

startMonth = 6;

endMonth = 8;

break;

case 4:

startMonth = 9;

endMonth = 11;

break;

default:

System.out.print("ERROR");

break;

}

int CurrentDayOfWeek = firstDayy;

int day = 1;

for (int m = 0; m < months.length; m++) {

if(CurrentDayOfWeek == 7){

CurrentDayOfWeek = 0;

}

while (day <= max[m]) {

while (CurrentDayOfWeek < days.length && day <= max[m]) {

day++;

CurrentDayOfWeek++;

}

if (day - 1 != max[m]) {

CurrentDayOfWeek = 0;

}

if(m >= startMonth && m <= endMonth){

weeksInQuarter++;

}

}

day = 1;

}

}

static void question3(int year, String q3Day, int q3Semi) {

int startMonth = 0, endMonth = 0;

int q2D = getDay(q3Day);

if(q3Semi == 1){

startMonth = 0;

endMonth = 5;

}else if(q3Semi == 2){

startMonth = 6;

endMonth = 11;

}else{

System.out.print("ERROR.");

}

int CurrentDayOfWeek = firstDayy;

int day = 1;

for (int m = 0; m < months.length; m++) {

if(CurrentDayOfWeek == 7){

CurrentDayOfWeek = 0;

}

while (day <= max[m]) {

while (CurrentDayOfWeek < days.length && day <= max[m]) {

if ((m >= startMonth && m <= endMonth) && CurrentDayOfWeek == q2D) {

spDayInSemiYear++;

}

day++;

CurrentDayOfWeek++;

}

if (day - 1 != max[m]) {

CurrentDayOfWeek = 0;

}

}

day = 1;

}

}

static void question4(int year) {

int daysCounter = 0;

int CurrentDayOfWeek = firstDayy;

int day = 1;

for (int m = 0; m < months.length; m++) {

if(CurrentDayOfWeek == 7){

CurrentDayOfWeek = 0;

}

while (day <= max[m]) {

while (CurrentDayOfWeek < days.length && day <= max[m]) {

daysCounter++;

day++;

CurrentDayOfWeek++;

}

if (day - 1 != max[m]) {

CurrentDayOfWeek = 0;

}

if(daysCounter == 7){

cmpWeeks++;

}

daysCounter = 0;

}

day = 1;

}

}

static int getMonth(String month) {

int monthNum = 0;

switch (month.toLowerCase()) {

case "january":

monthNum = 0;

break;

case "february":

monthNum = 1;

break;

case "march":

monthNum = 2;

break;

case "april":

monthNum = 3;

break;

case "may":

monthNum = 4;

break;

case "june":

monthNum = 6;

break;

case "july":

break;

case "august":

monthNum = 4;

break;

case "september":

monthNum = 5;

break;

case "october":

monthNum = 6;

break;

case "november":

break;

case "december":

monthNum = 4;

break;

default:

System.out.print("Invalid month.");

break;

}

return monthNum;

}

static int getDay(String day) {

int dayNum = 0;

switch (day.toLowerCase()) {

case "monday":

dayNum = 1;

break;

case "tuesday":

dayNum = 2;

break;

case "wednesday":

dayNum = 3;

break;

case "thursday":

dayNum = 4;

break;

case "friday":

dayNum = 5;

break;

case "saturday":

dayNum = 6;

break;

case "sunday":

break;

default:

System.out.print("Invalid day.");

break;

}

return dayNum;

}

}