1.回文日期

**题目描述**

2020 年春节期间，有一个特殊的日期引起了大家的注意：2020 年 2 月 2 日。因为如果将这个日期按 “yyyymmdd” 的格式写成一个 8 位数是 20200202，恰好是一个回文数。我们称这样的日期是回文日期。

有人表示 20200202 是 “千年一遇” 的特殊日子。对此小明很不认同，因为不到 2 年之后就是下一个回文日期：20211202 即 2021 年 12 月 2 日。

也有人表示 20200202 并不仅仅是一个回文日期，还是一个 ABABBABA 型的回文日期。对此小明也不认同，因为大约 100 年后就能遇到下一个 ABABBABA 型的回文日期：21211212 即 2121 年 12 月 12 日。算不上 “千年一遇”，顶多算 “千年两遇”。

给定一个 8 位数的日期，请你计算该日期之后下一个回文日期和下一个 ABABBABA 型的回文日期各是哪一 天。

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        int num = scanner.nextInt();

        int ans1 = 0, ans2 = 0;

        boolean flag = false;

        while (true) {

            num = runTime(num);

            if (!flag && judge(num)) {

                ans1 = num;

                flag = true;

            }

            if (judge(num) && judge2(num)) {

                ans2 = num;

                break;

            }

        }

        System.out.println(ans1);

        System.out.println(ans2);

        scanner.close();

    }

    public static boolean judge(int num) {

        String string = Integer.toString(num);

        StringBuilder stringBuilder = new StringBuilder(string);

        String  s=stringBuilder.reverse().toString();

        if (string.equals(s)) {

            return true;

        }

        return false;

    }

    public static boolean judge2(int num) {

        String string = Integer.toString(num);

        if (string.charAt(0) == string.charAt(2) && string.charAt(1) == string.charAt(3)) {

            return true;

        }

        return false;

    }

    public static int runTime(int num) {

        int year = num/10000, month = (num/100)%100, day = num%100;

        day++;

        if(day > 31) {

            day = 1;

            month ++;

            if (month > 12) {

                month = 1;

                year++;

            }

        }

        return year\*10000+month\*100+day;

    }

}