```java
import fr.jussieu.script.Deug;
class Dates {
    // Renvoie le nom du jour de la semaine dont le numéro est passé
    // en argument. Le numéro doit être compris entre 0 (dimanche) et 6.
    static String nomDuJour(int n) {
        switch (n) {
        // Remarque : normalement, tout « case » doit être terminé par
        // un « break ».
        case 0: return "dimanche";
        case 1: return "lundi";
        case 2: return "mardi";
        case 3: return "mercredi";
        case 4: return "jeudi";
        case 5: return "vendredi";
        case 6: return "samedi";
        }
        return "(appel incorrect a nomDuJour)";
    }

    // Renvoie le numéro du jour dans l'année correspondant au jour
    // numéro j (1 <= j <= 31) du mois m (1 <= m <= 12). Le résultat
    // est compris entre 1 et 365 (366 si l'année est bissextile).
    // (L'exercice de départ ne demandait pas de s'occuper des années
    // bissextiles.)
    static int jourDansLAnnee(int j, int m, boolean bissextile) {
        int x = j;
        switch (m - 1) { // on teste le numéro du mois précédent
        // Noter l'omission des « break ». Ainsi, on commence à faire
        // les additions à la ligne correspondant à la valeur de m -
        // 1, et on fait aussi les additions sur les lignes suivantes,
        // jusqu'à la ligne pour la valeur 1.
        case 11: x += 30;
        case 10: x += 31;
        case  9: x += 30;
        case  8: x += 31;
        case  7: x += 31;
        case  6: x += 30;
        case  5: x += 31;
        case  4: x += 30;
        case  3: x += 31;
        case  2: x += 28; if (bissextile) x += 1;
        case  1: x += 31;
        }
        return x;
    }

    // Une variante de jourDansLAnnee qui n'utilise pas la
    // fonctionnalité douteuse du switch sans case.
    static int jourDansLAnnee_ifret(int j, int m, boolean bissextile) {
        int x = j;
        // Noter qu'on a mis deux instructions sur chaque ligne, pour
        // des raisons esthétiques (chaque ligne correspond à un
        // mois).
        if (x ==  1) return x;   x += 31;
        if (x ==  2) return x;   x += 28; if (bissextile) x += 1;
        if (x ==  3) return x;   x += 31;
        if (x ==  4) return x;   x += 30;
        if (x ==  5) return x;   x += 31;
        if (x ==  6) return x;   x += 30;
        if (x ==  7) return x;   x += 31;
        if (x ==  8) return x;   x += 31;
```



```java
        if (x ==  9) return x;   x += 30;
        if (x == 10) return x;   x += 31;
        if (x == 11) return x;   x += 30;
        return x;
    }

    // Renvoie le nom du jour correspondant à la date j/m/2003.
    static String date2003(int j, int m) {
        // On va utiliser la méthode nomDuJour, qui demande un
        // argument compris entre 0 et 6. De plus, le 31/12/2002 est
        // un mardi, et nomDuJour part de 0 pour dimanche.
        return nomDuJour((jourDansLAnnee(j, m, false) + 2) % 7);
    }

    // Renvoie le nombre d'années bissextiles comprises entre a1 exclue
    // et a2 incluse. Si a2 < a1, la valeur renvoyée est négative.
    static int compteBissextiles(int a1, int a2) {
        // Le nombre de multiples de 4 compris entre 0 exclu et n
        // inclus est n/4 (quotient de n par 4).
        // Le nombre de multiples de 4 qui ne sont pas multiples de
        // 100 compris entre 0 exclu et n inclus est (n/4 - n/100),
        // car tout multiple de 100 est un multiple de 4.
        // De même, le nombre multiples de 4 qui ne sont pas multiples
        // de 100 sauf à être multiple de 400 compris entre 0 exclu et
        // n inclus est (n/4 - n/100 + n/400).
        return (a2/4 - a2/100 + a2/400) - (a1/4 - a1/100 + a1/400);
    }

    // Teste si l'année passée en argument est bissextile.
    static boolean estBissextile(int a) {
        return a % 4 == 0 && (a % 100 != 0 || a % 400 == 0);
    }

    // Renvoie le nom du jour de la semaine qu'est le j/m/a.
    static String jourDeNumeros(int j, int m, int a) {
        // On calcule dans x le nombre de jours entre le 31/12/2002 et
        // le j/m/a.
        // On commence par calculer le nombre de jours entre le
        // 31/12/2002 et le 31/12/(a-1) : on compte le nombre
        // d'années, on multiplie par le nombre de jours dans une
        // année normale, et on ajoute les 29 février.
        int x = 365 * (a - 2003) + compteBissextiles(2003, a - 1);
        // On ajoute le nombre de jours dans l'année a, puis on se
        // ramène à l'intervalle accepté par nomDuJour, comme dans
        // date2003.
        x += jourDansLAnnee(j, m, estBissextile(a));
        int r = (x + 2) % 7;
        // La ligne suivante est tout ce qu'il faut ajouter pour
        // traiter les dates avant 2003. Elle est nécessaire parce
        // que, lorsque x < 0 et y > 0, l'expression « x % y » renvoie
        // r - y si r != 0 (et 0 si r = // 0), r étant le reste de la
        // division de x par y (0 <= r < y).
        // Au lieu d'écrire cette ligne, on pourrait modifier
        // nomDuJour pour qu'elle fonctionne aussi si son argument est
        // compris entre -6 et -1.
        if (r < 0) r += 7;
        // Il ne reste plus qu'à transformer le numéro en nom.
        return nomDuJour(r);
    }
```

```
    // Renvoie le jour de la semaine correspondant à une date passée
    // sous la forme "JJ/MM/AAAA".
    static String jourDeTexte(String date) {
        // On vérifie sommairement que l'argument a l'air valide (ce
        // n'était pas demandé.)
        if (Deug.length(date) != 10) {
            Deug.println("Date mal formee : " + date);
            return "erreur";
        }
        // Les 3 lignes suivantes effectuent le décodage demandé.
        int j = Deug.stringToInt(Deug.subString(date, 0, 2));
        int m = Deug.stringToInt(Deug.subString(date, 3, 5));
        int a = Deug.stringToInt(Deug.subString(date, 6, 10));
        if (j < 1 || j > 31) {
            Deug.println("Jour incorrect dans " + date);
            return "erreur";
        }
        if (m < 1 || m > 12) {
            Deug.println("mois incorrect dans " + date);
            return "erreur";
        }
        if (a <= 0) {
            Deug.println("annee incorrecte dans " + date);
            return "erreur";
        }
        // Pour compléter les vérifications, il faudrait regarder si j
        // est un numéro de jour valide dans le mois m (en regardant
        // aussi a dans le cas du 29 févrifer). Voir le TP 2.
        return jourDeNumeros(j, m, a);
    }

    public static void main(String[] args) {
        // Ce programme principal lit des lignes au format jj/mm/aaaa
        // et affiche le jour de la semaine correspondant à chaque
        // date. On quitte le programme avec une ligne contenant juste
        // un « . ».
        String ligne = Deug.readLine();
        while (! Deug.equals(ligne, ".")) {
            Deug.println("Le " + ligne + " est un " + jourDeTexte(ligne));
            ligne = Deug.readLine();
        }
    }
}
```