



Exercice 1

Rappel : Si a et b sont deux valeurs entières, alors $a \% b$ représente le reste de la division entière de a par b . Par exemple, $7 \% 3$ vaut 1.

Nous nous intéressons dans cet exercice à inverser le processus de transformation d'une date en format date entier. Pour cela nous considérons trois fonctions permettant d'extraire les numéros de jour, de mois et d'année. La première fonction est :

```
public static int
    getMois(int date)
/* L'entier en paramètre représente une date au format date
 * entier.
 * La fonction retourne le numéro de mois de cette date
 */
{
    return (date / 100) % 100 ;
}
```

1. Compléter le corps de la fonction suivante :

```
public static int
    getJour(int date)
/* L'entier en paramètre représente une date valide au format
 * date entier. La fonction retourne le numéro de jour de cette
 * date.
 */
{
    à compléter
}
```

2. Ecrire une fonction `getAnnee` permettant d'extraire l'année d'un entier représentant une date au format date entier.
3. Compléter la fonction suivante :

```
public static String
    dateEnChaineCourte (int date)
/* L'entier en paramètre représente une date au format entier
 * date. La fonction retourne une chaîne de caractère
 * contenant la date au format jour/mois/annee
 */
{
    à compléter
}
```

Exercice 2

Un nombre représente une année bissextile s'il est divisible par 4 (comme 1904), à l'exclusion des années séculaires (comme 1900) qui ne sont pas bissextiles, sauf les années multiples de 400 (comme 2000) qui, bien que séculaires, sont bissextiles. Dans cet exercice, on désire compléter la fonction suivante de deux manières différentes :

```
public static boolean
    estAnneeBissextile(int an)
/* Le paramètre désigne un numéro d'année.
 * La fonction retourne vrai si l'année est une
 * année bissextile, elle retourne faux sinon.
 */
```

1. Donnez le corps de la fonction en utilisant une unique expression booléenne qui vaut vrai si la réponse à la question est oui, et qui vaut faux sinon.
2. Donnez le corps de la fonction en utilisant des structures conditionnelles et des tests simples (on n'utilisera pas les opérateurs et/ou) qui renvoie true si la réponse à la question est oui, et qui renvoie false sinon.

Exercice 3

Nous nous intéressons à écrire une fonction pour vérifier la validité d'une date. Pour cela, nous utiliserons la fonction suivante (on suppose que comme la fonction `estAnneeBissextile`, elle est rangée dans la bibliothèque `ManipsDates`) :

```
public static int
    nbJoursMois(int mois, int annee)
/* Le paramètre mois désigne un numéro de mois
 * (Précondition : c'est un entier entre 1 et 12)
 * Le paramètre année désigne un numéro d'année.
 * La fonction retourne le nombre de jours dans le mois
 * en tenant compte du fait que l'année concernée est
 * éventuellement bissextile.
 */
{
    if (mois == 2)
        if (estAnneeBissextile(annee))
            return 29 ;
        else
            return 28 ;
    else
        if ((mois == 4) || (mois == 6) || (mois == 9) ||
            (mois == 11))
            return 30 ;
        else
            return 31 ;
}
```

Compléter la fonction :

```
public static boolean
    estDateValide(int jour, int mois, int annee)
/* Les paramètres désignent une date dont on veut vérifier la
 * validité. La fonction retourne vrai si la date est valide,
 * elle retourne faux sinon.
 */
```