

Exercices de conversions de types

Exercice 7.1.1 *conversion de String vers int*

Ecrire une méthode appelée `stringToInt` qui réalise la conversion d'une chaîne de caractère de la même façon que la méthode prédéfinie `Integer.parseInt`, c'est-à-dire qui renvoie la valeur numérique de la chaîne si celle-ci représente un nombre entier et qui provoque une erreur dans les autres cas (par exemple si la chaîne contient des lettres). Vous vous assurerez que la méthode fonctionne également pour des nombres négatifs.

Vous pourrez utiliser les deux méthodes `Character.isDigit` qui détermine si un caractère est un chiffre et `Character.getNumericValue` qui renvoie la valeur numérique d'un chiffre pour faire ce programme. Ces deux méthodes prennent un caractère en paramètre.

Exercice 7.1.2 *extraction de mots*

On va considérer qu'un mot est une suite de caractères qui ne comprend que des lettres. Tous les autres caractères seront considérés comme des séparateurs utilisés pour séparer les différents mots. Pour déterminer si un caractère est une lettre, on utilisera la méthode `Character.isLetter`.

Ecrire une méthode qui transforme une chaîne de caractères en un tableau contenant les mots de cette chaîne dans le même ordre.

Par exemple la chaîne : "bonjour mon ami" sera convertie dans le tableau de chaînes {"bonjour", "mon", "ami"}. Pour ce faire, il faut parcourir la chaîne à convertir et mémoriser à tout moment le début déjà lu du mot courant. Il faut aussi distinguer si l'on est en train de lire un mot ou un séparateur séparant deux mots. Il ne faut pas négliger le fait qu'il peut y avoir plusieurs caractères séparateurs successifs tels que plusieurs espaces entre deux mots.

Exercice 7.1.3 *conversion de chiffres romains*

Les nombres romains sont des nombres écrits au moyen de lettres qui peuvent être considérées comme les chiffres de ce mode de notation. Chaque lettre est associée à un nombre qui s'ajoute ou se retranche selon la position du chiffre dans le nombre.

Voici la correspondance entre lettres et valeurs numériques :

lettre	I	V	X	L	C	D	M
valeur	1	5	10	50	100	500	1000

Question 1 *conversion de chiffre*

Ecrire une méthode qui convertit un chiffre romain donné au moyen d'un caractère du type `char` vers la valeur numérique correspondante (type `int`). Dans le cas où le caractère n'est pas une des sept lettres utilisée par les nombres romain, la méthode doit provoquer une erreur.

Question 2 *conversion de nombre*

Voici l'explication du système d'écriture donné par l'encylopédie *wikipedia* : Pour connaître la valeur d'un nombre écrit en chiffres romains, il faut lire le nombre de droite à gauche, il suffit d'ajouter la valeur du chiffre, sauf s'il est inférieur au précédent, dans ce cas, on le soustrait. Ainsi :

- XVI = 1 + 5 + 10 = 16 ;
- XIV = 5 - 1 + 10 = 14, car I est inférieur à V ;
- DIX = 10 - 1 + 50 = 59, car I est inférieur à X ;
- MMMCMXCIX = 10 - 1 + 100 - 10 + 1 000 - 100 + 1 000*3 = 3 999 ;

Ecrire une méthode qui convertit un nombre romain écrit dans une String en une valeur numérique de type `int`. Il sera utile de faire appel à la méthode écrite à la question précédente.

Exercice 7.1.4 *cryptographie élémentaire*

L'encodage et le décodage sont deux opérations symétriques qui permettent de transformer un certain message soit pour conserver sa confidentialité (cryptographie) soit pour respecter des contraintes techniques. Par exemple, les ordinateurs utilisent des codages binaires de toutes les données qu'il manipulent : textes, nombres, images, sons, etc.

Question 1 *code de Jules César*

Pour crypter ses messages, Jules César a utilisé un code qui consistait à remplacer chaque lettre par la lettre située trois crans plus loin dans l'alphabet. Par exemple, le a est codé par un d, le b par un e, etc. Les trois dernières lettres de l'alphabet sont codées respectivement par a, b et c.

Ecrivez les fonctions d'encodage et de décodage pour ce code. Ces fonctions ne seront définies que pour les lettres, les autres caractères restant inchangés. Ces fonctions permettront d'encoder et de decoder des chaînes de caractère.

Notez qu'en Java, le type `char` est numérique : on peut lui appliquer les opération arithmétiques et notamment l'addition et la soustraction utiles dans cet exercice.

Question 2 *un code à clé*

On veut garder le principe de remplacement d'une lettre par une autre caractérisée par son décalage dans l'alphabet, mais on veut maintenant que ce décalage varie d'une lettre à l'autre selon une clé numérique. Chaque chiffre de la clé donne le décalage d'une lettre.

Prenons par exemple la clé 23781. Cette clé va s'appliquer successivement à plusieurs morceaux de chaîne de longueur 5. La première lettre sera remplacée par la deuxième lettre qui suit dans l'alphabet car 2 est le premier chiffre de la clé. La deuxième lettre sera remplacée par la 3ème suivante, la troisième par la septième suivante, etc.

Par exemple, le message : `bonjour, comment va?` sera codé comme suit :

b	o	n	j	o	u	r	c	o	m	m	e	n	t	v	a
2	3	7	8	1	2	3	7	8	1	2	3	7	8	1	2
d	r	u	r	p	w	u	j	w	n	o	h	u	b	w	c

Donnez les fonctions d'encodage et de décodage pour cette méthode de codage. La clé sera donnée en paramètre sous la forme qui vous semble la plus pratique.