

# Exercices INF3140 : série #1

## 1. Quantification sur des ensembles de String

### Le type String en OCL

En OCL, les chaînes de caractères — type **String** — sont des suites de caractères indiquées par un simple guillemet (apostrophe), par exemple :

```
'bonjour'  
'100'  
''
```

Il n'y a pas de type **Char** en OCL.

Les principales opérations sur les chaînes de caractères sont les suivantes :

- `ch1.concat(ch2)` retourne une chaîne composée des caractères de `ch1` suivis de ceux de `ch2` :

```
'100'.concat('12') = '10012'  
''.concat('xyz') = 'xyz'
```

- `ch.size()` retourne la longueur de `ch` :

```
''.size() = 0  
'10123'.size() = 5
```

- `ch.substring(i,j)` retourne la sous-chaîne allant de la position `i` à la position `j` — le premier caractère est à la position 1 :

```
'10123'.substring(2,4) = '012'  
'10123'.substring(5,5) = '3'
```

- `ch.toLowerCase()` et `ch.toUpperCase()` produisent une nouvelle chaîne où les lettres sont transformées, respectivement, en minuscules ou en MAJUSCULES.

## Quelques ensembles

Soit les ensembles suivants :

`pairs` : `Set(String) = Set{'0', '2', '4', '6', '8'}`

`impairs` : `Set(String) = Set{'1', '3', '5', '7', '9'}`

`chiffres` : `Set(String) = Set{'0', '1', '2', '3', '4', '5', '6', '7', '8', '9'}`

`s1` : `Set(String) = Set{'100', '200', '300'}`

`s2` : `Set(String) = Set{'100100', '200', '300300'}`

## Énoncés à formaliser (avec des quantificateurs)

- a. Toutes les chaînes qui représentent des chiffres `pairs` sont aussi des chaînes qui représentent des chiffres.
- b. L'ensemble des chaînes qui représentent des chiffres peut être décomposé (partitionné) en deux parties distinctes et indépendantes : l'ensemble des chiffres `pairs` et l'ensemble des chiffres `impairs`.
- c. Parmi les chaînes de `s2`, il y en a plusieurs (2 ou plus) qui sont formées en «dédoublant» l'une des chaînes de `s1` — par exemple, `'100100'` de `s2` est obtenue en dédoublant `'100'` de `s1`.
- d. Toutes les chaînes qui font partie de `s1` sont composées uniquement de chiffres.