Institut National Universitaire Champollion

TP 5 - Récursivité sur les chaînes de caractères

Programmation fonctionnelle en Caml
L3 INFO - Semestre 6

Le fichier *chaines.ml* vous fournit quelques primitives de traitement des chaînes de caractères. Vous pourrez notamment utiliser les fonctions suivantes :

1. *longChaine s* : longueur d'une chaîne *s*
2. *niemeCar(n,s)* : n-ième caractère d'une chaîne *s*
3. *sousChaine(s,n,m)* : extrait d'une chaîne *s* entre les caractères *n* et *m*
4. *tetec s* : donne l'initiale d'un mot sous forme de caractère
5. *tetes s* : donne l'initiale d'un mot sous forme de chaîne (pour pouvoir concaténer)
6. *reste s* : donne le mot privé de son initiale

1 - Fonctions usuelles opérant sur les chaînes de caractères

Écrire les fonctions suivantes :

1. `majuscule : char -> bool`, `minuscule : char -> bool` et `lettre : char -> bool`, des fonctions booléennes qui indiquent la propriété éponyme à un caractère donné en paramètre. Par exemple, `majuscule c` doit rendre vrai si et seulement si `c` est une lettre majuscule.
   Rappel : on a une relation d'ordre sur les caractères telle que 'A'< 'Z'<'a'<'z'.
2. `appartient : char * string -> bool` : fonction récursive booléenne déterminant si un caractère appartient à une chaîne
3. Écrire une fonction booléenne `debut : string * string -> bool` qui indique si une chaîne est le début d'une autre.
4. Écrire une fonction booléenne `incluse :string * string -> bool` déterminant si une chaîne est incluse dans une autre.
   Utilisez la fonction `debut`.
5. Écrire une fonction `frequence : char * string -> int` donnant le nombre d'occurrences d'un caractère donné dans une chaîne.
6. Écrire une fonction `elimine : char * string -> string` qui élimine toutes les occurrences d'un caractère donné dans une chaîne donnée.
7. Écrire une fonction `renverse : string -> string` qui renverse une chaîne.

2 - Chiffres romains

On souhaite écrire des fonctions permettant l'évaluation de nombres écrits en chiffres romains.

- On se restreindra aux nombres s'écrivant à l'aide des seuls chiffres romains I, V et X.
- On suppose dans tout l'exercice que les nombres fournis par l'utilisateur ont des syntaxes correctes, les fonctions à écrire n'auront donc pas à le vérifier.
- On suppose enfin que l'on dispose des fonctions `tetec`, `tetes` et `reste` définies sur les chaînes des caractères.

1. On suppose dans un premier temps que l'on n'utilise pas les écritures IV pour 4 ni IX pour 9. (Le système est donc purement additif).

   a) Écrire une fonction `chiffre : char -> int`, définie par filtrage, qui à un caractère parmi I, V et X, représentant un chiffre romain associe sa valeur entière.

   ```
   chiffre('V') ; ;
   - : int = 5
   ```

   b) Écrire une fonction récursive `rom1 : string -> int` , qui à une chaîne de caractères représentant un nombre romain associe sa valeur.

   ```
   rom1 "XXVIII" ; ;
   - : int = 28
   ```

2. On suppose maintenant que l'on autorise les écritures IV pour 4 et IX pour 9.

   a) Écrire une fonction `valeurI : char -> int`, qui à un caractère *c* parmi I, V, X, associe la valeur entière du nombre romain "Ic".

   ```
   valeurI 'X' ; ;
   - : int = 9
   ```

   b) Écrire une fonction récursive `romain : string -> int`, qui à une chaîne de caractères représentant un nombre romain associe sa valeur.

   ```
   romain "XXIV" ; ;
   - : int = 24
   romain "XXXIX" ; ;
   - : int = 39
   ```