

# Algo I - TD5

## Conception orientée fonctions

Frédéric RAYAR

Novembre 2012

### 1 Liste des noms propres

On souhaite rechercher les noms propres dans un fichier texte, c'est-à-dire tous les mots commençant par une majuscule (hypothèse simpliste), et les classer par ordre alphabétique. On désire par ailleurs, que les résultats présentés, n'affichent pas les doublons qui auraient éventuellement pu être trouvés.

1. Définir et représenter sous forme **arborescente** l'ensemble des modules fonctionnels, nécessaire pour résoudre le problème,
2. Pour les principaux modules, proposer des **spécifications formelles**, ainsi que les algorithmes correspondant.

### 2 Tournoi sportif

On souhaite gérer automatiquement un tournoi entre  $n$  équipes de manière à déterminer un classement de toutes les équipes en fonction du nombre de points obtenus (match perdu = 0, nul = 1, gagné = 3). Plus précisément, le système doit gérer les points obtenus pour chaque équipe, au fur et à mesure que les matches ont lieu. On part sur le principe que toutes les équipes vont se rencontrer exactement 1 fois, dans n'importe quel ordre. Le gestionnaire doit pouvoir entrer le résultat des matches et souhaite obtenir automatiquement le classement dès que tous les matches sont renseignés.

Effectuez le même travail que pour l'exercice précédent.

### 3 Pierre Feuille Ciseaux (facultatif)

Ecrire un algorithme permettant de jouer au jeu pierre-feuille-ciseaux contre l'ordinateur. L'utilisateur et l'ordinateur ont trois coups possibles : pierre, feuille ou ciseaux. Les règles sont les suivantes :

- les deux joueurs jouent la même chose : pas de point
- Pierre contre Feuille : Feuille gagne (et marque 1 point)
- Pierre contre Ciseaux : Pierre gagne (et marque 1 point)
- Ciseaux contre Feuille : Ciseaux gagne (et marque 1 point)

L'utilisateur entre 1, 2 ou 3 et l'ordinateur tire un nombre aléatoire compris entre 1 et 3. On affiche à chaque fois le coup correspondant au nombre choisi par chaque joueur (1=pierre, 2=feuille, 3=ciseaux) ainsi que le gagnant. La partie est remportée par le premier à atteindre 10 points.

On utilisera la fonction aléatoire pour obtenir une valeur aléatoire :  
Fonction *Aléatoire* (entier nb) : entier, Renvoie une valeur tirée aléatoirement entre 0 et nb.