

## TP5 - Un simplificateur de “Glob patterns”

Fichiers du projet : TP5\_ACF.zip.

Le TP est à rendre **sur le MOODLE du cours** au plus tard le *Lundi 21/10 à 14h00*.

Les TP's devant être mis en ligne pour être évalués en même temps, **aucun retard ne sera possible**. Déposez sur MOODLE une archive au format **ZIP** dont le nom sera de la forme **Nom1\_Nom2\_TP5\_ACF.zip** et contenant :

1. **un JAR** de votre projet TP5 au format `Nom1_Nom2_TP5.jar`. Pour générer le `.jar`, dans le shell SBT, tapez `package`. Le `.jar` doit être généré le répertoire `target/scala-2.13` du projet. **Renommez** le `.jar` avec le nom de votre binôme `Nom1_Nom2_TP5.jar`. Remarque : le `.jar` n'a pas besoin d'être exécutable.
2. **le répertoire de projet TP5 complet**.

Remarque : Tout **votre** code Scala devra être contenu dans le fichier `Simplify.scala`.

### 1 Préliminaires

Décompressez l'archive `/share/m1info/ACF/TP5/TP5_ACF.zip` dans votre répertoire ACF. Dans votre éditeur Scala, ouvrez le projet TP5\_ACF.

### 2 Objectif

L'objectif de ce TP est de programmer en Scala un simplificateur de “glob patterns” (voir wikipédia). Les “glob patterns” (globs dans la suite) représentent un ensemble de noms de fichiers à l'aide de “jokers”. Par exemple en Bash, la commande `mv *.txt` déplace tous les fichiers dont les noms se terminent par `.txt`. Le glob est ici `*.txt` où `*` est le joker représentant n'importe quelle chaîne de caractères (possiblement vide). Le second joker que nous considérerons est `?` qui représente exactement un caractère. A titre d'exemple, le glob `a?c*d` :

- reconnaît les chaînes `"abcd"`, `"acceed"`, `"accddd"`, `"acccddd"`, ...
- mais il ne reconnaît pas les chaînes `""`, `"ac"`, `abc`, `acd`, ...

Enfin, le dernier joker sera `+` qui représente n'importe quelle chaîne de caractères **non vide**. Pour simplifier, les chaînes de caractères sont définies sur l'ensemble de caractères `'a' à 'z'`, `'A' à 'Z'`, `'0' à '9'` ainsi que le caractère `'.'`. Dans le projet Scala que vous avez chargé, un glob est représenté par une liste d'objets de type `Symbol` où un `Symbol` est soit `Char(x)` un caractère, `Star` un joker `*`, `Qmark` un joker `?`, ou `Plus` un joker `+`. La définition Scala du type `Symbol` est la suivante :

```
sealed trait Symbol
case class Char(a: ScalaChar) extends Symbol
case object Star extends Symbol
case object Qmark extends Symbol
case object Plus extends Symbol
```

L’objectif de ce TP est de développer un objet Scala qui implémente le trait `Simplifier` et propose une fonction `process` qui simplifie un glob.

```
trait Simplifier{  
  def simplify(p:List[Symbol]):List[Symbol]  
}
```

Un glob simplifié doit être **le plus court possible** mais il doit reconnaître le même ensemble de chaînes que le glob initial.

### 3 Marche à suivre

1. Dans le projet TP5, vous devez remplacer la chaîne `LE_NOM_DE_VOTRE_BINOME` par le nom de votre binôme à plusieurs endroits : dans les noms de répertoires et dans quelques fichiers sources. Certains éditeurs permettent de remplacer un nom de package rapidement (“refactoring”).
2. Dans la classe `MySimplify` qui implémente `Simplifier`, programmez la fonction `simplify` qui simplifie un glob de façon optimale : Un glob simplifié doit être **le plus court possible** mais il doit reconnaître le même ensemble de chaînes que le glob initial.
3. Testez votre implantation en exécutant l’objet exécutable `Application` qui lit un glob et donne la version simplifiée avec votre implantation du `Simplifier`.
4. Vous pouvez aussi définir des tests unitaires dans le répertoire `test` du projet, classe `TestSimplify`.