

TP 4 - Automatisation

Rapport

BACH William 20127144

LUNICA Albert 20101201

Fonctionnement de proto.py

Lorsque proto.py se retrouve dans le même dossier que tp1.jar, celui-ci peut être exécuté par la commande: `python .\proto.py <"Git URL"> <"PATH for metrics.csv">`.

Tâche 1

Notre première tâche était de créer un algorithme "proto.py" qui trouve les versions d'un projet sur github ainsi que le nombre de classes. Cette tâche a été accomplie, à l'aide des fonctions suivantes:

- **listFiles(*path*)**: cette fonction prend comme argument un chemin vers un dossier et celui-ci, de manière recursive, trouve tous les fichiers qui se retrouvent dedans et les retourne sous forme de liste de Strings.
 - **listJavaFiles(*filesArray*)**: cette fonction prend en argument la liste des fichiers retournée par listFiles(*path*). Elle utilise celle-ci pour trier tous les fichiers qui se trouvent à ne pas être des fichiers qui finissent par .java et retourne une liste de celle-ci, semblable à celle de listFiles(*path*).
 - **iterateVersions(*url,path*)**: cette fonction prend en argument le url vers un projet gitHub ainsi que le chemin où le fichier metrics.csv sera enregistré. Celle-ci utilise la commande "repo.git.rev_list('MASTER')" pour trouver toutes les versions en hexadecimale du projet ainsi que la fonction "repo.git.reset('--hard',hexVersion)" pour reseter le projet à la version désirée.
-

-
- **classMetrics(version,tempPath):** cette fonction s'occupe de chaque ligne du fichier csv. Il retourne une ligne de texte correspondant à "id_version,n_classes,m_c_BC".
 - **createCSV(path):** cette fonction prend en argument le chemin vers un dossier et crée un fichier "metrics.csv" qui détient toutes les métriques désirées.
 - **cleanTemp(path):** pour des raisons de propreté, tous les fichiers clonés ainsi que les fichiers utiles pour calculer les metrics se trouvent dans un fichier appelé "temp" dans le même dossier que le fichier metrics.csv . Ce dossier est effacé par cette fonction qui prend en argument le chemin vers ce fichier temp.

Tâche 2

Notre deuxième tâche a été de calculer n_classes et m_c_BC et de les enregistrer sous le même fichier csv que dans la tâche 1. Cette tâche a été complétée avec success grâce aux fonctions suivantes:

- **analyseJavaCSV(version,path):** cette fonction utilise le fichier tp1.jar pour calculer les metrics des classes pour ensuite calculer les medianes et de les transferer à `iterateVersions(url,path):` qui lui s'occupe de la structure du fichier metrics.csv .