

# STRUCTURES DE DONNÉES ET ALGORITHMES (SINF1121)

## Mission 1 - Produit final

– 1 octobre 2013 –

### TRAVAIL DU GROUPE 9 :

GÉGO Anthony	28581100
GENA Xavier	xxxxxx00
JOVENEAU Quentin	xxxxxx00
LIBIOULLE Thibault	60271100

MOYAU Arnold	xxxxxx00
NAVEAU Adrien	xxxxxx00
PAYEN Marlon	xxxxxx00

# 1 Cahier des charges

Notre première mission consistait à réaliser une petite calculatrice en notation polonaise inverse. Des nombres sont stockés sur une pile, et les opérations s'exécutent sur cette dernière. Toutes les commandes sont lues à partir d'un fichier postscript. Les différents morceaux du programme sont :

- Une pile, sur laquelle seront stockées les données, à savoir, les nombres et les booléens.
- Des commandes, telle que l'addition ou la soustraction, qui agiront sur la pile, notamment pour enlever les données nécessaires à l'opération et pour y ajouter leur résultat.
- Un interpréteur, qui prend une décision pour chaque mot (token) lu dans le fichier. S'il s'agit d'une donnée, il l'ajoute sur la pile, ou lance une opération s'il s'agit d'une commande.

Les différentes tâches peuvent être réparties aisément. En effet, en définissant des interfaces, un premier membre du groupe peut s'occuper de la pile, un deuxième de l'interpréteur, et les autres peuvent se répartir les différentes commandes, sans avoir besoin de l'implémentation d'autrui.

## 2 Abstraction et réutilisation du code

L'abstraction de données nous permet de réutiliser et de compléter du code aisément. Ainsi, notre type pile, complètement isolée du reste du code de notre calculatrice, peut facilement être réutilisée dans un autre programme, pour autant qu'il respecte l'interface fournie. De même, d'autres opérations peuvent être ajoutées à la calculatrice sans devoir modifier le code en profondeur.

## 3 Changement d'implémentation

Puisqu'il s'agit d'un type abstrait de données, pour autant que la définition (l'interface) de la pile ne change pas, changer d'implémentation ne nécessite que de modifier le code à l'intérieur de la classe concernée. Ainsi, que la pile soit implémentée à l'aide d'une liste chaînée, doublement chaînée, ou d'un tableau, cela ne change rien aux yeux des autres classes qui l'utiliseront telle que définie dans l'interface.

## 4 Erreurs Postscript

Il est possible que différentes erreurs se posent lors de la lecture des mots (tokens) du fichier postscript. En voici une liste :

- Commande inconnue : si un mot ne correspond ni à une valeur, ni à une opération, le programme retourne une erreur, et s'arrête.
- Erreur de pile : si une valeur manque pour réaliser une opération, le programme retourne une erreur et s'arrête.
- Erreur de fichier : si le fichier d'entrée est introuvable, n'est pas lisible, ou s'il est impossible d'écrire dans le fichier de sortie, le programme retourne une erreur et s'arrête.
- Double définition : si une définition rentre en conflit avec les définitions existantes, le programme retourne une erreur et s'arrête.
- Opération sur types incompatibles : si une opération est réalisée sur un booléen et un nombre réel, le programme retourne une erreur et s'arrête.
- Division par zéro : si une division par zéro a lieu, un avertissement est retourné.