

Algorithmique avancée

TD n° 1

1. Evaluer une expression postfixée :

On veut évaluer des expressions arithmétiques postfixées simples formées à l'aide de chiffres (de 0 à 9) et d'opérateurs arithmétiques usuels (+ - * /). L'expression est stockée dans une chaîne de caractères (sans séparateur).

Rappel : notation **postfixée** : l'opérateur est noté après les 2 opérandes.
Exemple : "283-*4+"

L'évaluation de l'expression se fait en parcourant l'expression **une seule fois**.
A chaque étape on peut avoir :

- soit un opérande (et on le met de côté)
- soit une opération (et on l'applique aux deux derniers opérandes rencontrés, puis on met de côté le résultat obtenu).

A la fin de l'évaluation de l'expression (et du parcours) on doit avoir un seul nombre mis de côté : le résultat final.

a) Evaluer **plusieurs** expressions simples bien formées (ne contenant pas trop d'opérandes, ni d'opérations).

b) Quelle SAD est la plus adaptée pour stocker les nombres mis de côté ?

c) Evaluer une expression bien formée en utilisant cette SAD, tout en indiquant, étape par étape, la position dans la chaîne et le contenu de la SAD.

d) Ecrire l'algorithme qui permet d'évaluer une expression bien formée.

2. Implémenter une pile avec deux files :

On doit écrire les opérations spécifiées par la définition de la pile en utilisant uniquement les opérations abstraites des files.

Pour y parvenir il faut observer les bonnes propriétés qui permettront d'exploiter les deux files.

On considère qu'en dehors de l'exécution des opérations empiler et dépiler tous les éléments se trouvent dans une seule file (donc au moins une des files est vide à tout moment) : leur nom respectif est **fileTravail** et **fileVide**.

a) Comment implémenter l'opération **estVide** pour la pile ?

b) Dans quel ordre les éléments empilés successivement se retrouveront dans une des files ? Faites un choix !

c) En tenant compte du b) décrivez en français en quoi consiste l'implémentation de l'opération **empiler** en n'utilisant que des opérations des files et éventuellement une place de plus nommée **transit** (un conteneur).

Donner ensuite l'algorithme correspondant.

d) Quel est le coût (en nombre d'opérations élémentaires sur les files) pour **empiler** ?

e) Mêmes questions que c) et d) mais pour l'opération **dépiler**.