

Série d'exercices n°5

Exercice 1 :

Ecrire un algorithme qui permet d'inverser l'ordre de position des éléments d'une pile en utilisant une file.

Exercice 2 :

Soit P1 une pile contenant des entiers positifs.

1. Ecrire un algorithme qui permet de déplacer les éléments de P1 vers une autre pile P2 de en respectant la contrainte : « les nombres pairs sont en bas de la pile P2 ».

2. Refaire la question 1 en respectant la contrainte : « les nombres pairs sont en bas de la pile P2 et apparaissent dans le même ordre que dans P1, et les nombres impairs sont en haut de la pile et apparaissent dans le même ordre que dans P1 ».

Exercice 3 :

Il s'agit de déterminer si les parenthèses d'une chaîne de caractères sont écrites en respectant la syntaxe de la grammaire d'une expression. Par exemple, la chaîne $((()) ()) ()$ est bien écrite, tandis que les chaînes $) ()$ ou $())$ ne le sont pas.

Ecrire une fonction :

1. Qui retourne vrai si la chaîne de caractères est proprement écrite, et faux sinon.
2. Qui retourne la position de la première parenthèse qui ne respecte pas la règle d'écriture si la chaîne n'est pas bien écrite.

Exercice 4 :

Ecrire un algorithme qui transforme une expression infixée (avec des parenthèses) en notation post fixée (polonaise inversée). Par exemple, l'expression suivante :

$5 * (((12 - 3) / 3) - 1)$ devra être traduite en : $5 \ 12 \ 3 \ - \ 3 \ / \ 1 \ - \ *$

Les opérateurs valides sont +, -, *, /. L'algorithme lira une suite de caractères, et rangera le résultat de la conversion dans une file qui s'affichera.