

TP 26 - Les graphes

DÉTECTION DE CYCLE

Exercice 1 Propose un algorithme de détection de présence d'un cycle dans un graphe non orienté en vous inspirant de celui vu en cours qui concerne les graphes orientés.

LES PILES

Exercice 2 Notation polonaise inversée

Une expression algébrique écrite en notation polonaise inversée est de la forme :

$$1 \ 2 \ + \ 3 \ 4 \ / \ -$$

Le résultat attendu est : $(1 + 2) - (3/4)$.

L'opérateur est placé après ses opérandes. L'évaluation d'une telle expression utilise une pile. On parcourt l'expression (ou la liste) :

- Si l'on rencontre un nombre on l'empile.
- Si l'on rencontre un opérateur, on dépile les deux derniers nombres, on effectue le calcul et on empile le résultat.

Deux exemples :

- La notation du nombre $\frac{2 \times 3^2 - 1}{3}$ est $[2, 3, 2, ^, \times, 1, -, 3, /]$.
- Le nombre de notation $[1, 2, 3, 2, *, *, -,]$ est $1 - 2 \times 3^2$.

1. Donner le résultat de l'expression suivante :

$$1 \ 2 \ 3 \ 4 \ 5 \ / \ + \ \times \ -$$

2. Donner une écriture en notation polonaise inversée de

$$\frac{3}{4 * 5 + 1} + 2$$

3. Écrire le script d'une fonction `eval_npi(L)` permettant d'évaluer une expression écrite en notation polonaise inversée n'utilisant que les opérateurs usuels : $+$, $-$, $/$, $*$, $**$.

Les opérateurs seront traités sous la forme d'une chaîne de caractère. L'expression sera donnée dans une liste. Par exemple : $[1, 2, '+', 3, 4, '/', '-',]$.

Pour évaluer une expression de type "1, '+', 2" on pourra :

- soit discuter suivant l'opérateur et faire le calcul
- tout convertir en chaîne de caractère et utiliser la fonction `eval` : `eval(str(1)+'+'+str(2))`

4. Écrire une variante permettant l'appel de fonctions d'une variable réelle et à valeurs réelles.

Par exemple : $1 + \sqrt{2}$ correspond à $[1, 2, \text{sqrt}, '+']$.

Exercice 3 Parenthèses

1. Écrire le script d'une fonction `p(c)` qui permet de déterminer si une expression est bien parenthésée. Cette expression sera donnée sous la forme d'une chaîne de caractères.

Par exemple : $(1+2(4-5))$ ou $((()))$ conviennent mais $)($ ou $()()$ ne conviennent pas.

2. Écrire une nouvelle version de cette fonction `p2(c)` pour qu'elle affiche les couples de positions des parenthèses qui se correspondent ou retourne la position d'une parenthèse posant problème.

3. Généraliser cette fonction, `p3(c)`, pour traiter le cas où il y a plusieurs types de parenthèses : $(, [, \{$.

Par exemple : $[() \{ \}]$ convient mais $\{ (\})$ ne convient pas.