

TP – Pile et Files 1 Evaluation d’une expression mathématique postfixée

1. Evaluation d’une expression mathématique

Objectif : évaluer l’expression : $x = (3-2)* 2$ à l’aide d’une pile

On rappelle : **Méthode d’évaluation d’une expression postfixée (ou encore polonaise inverse):**

L est une liste python

- on utilise les éléments de la liste L les uns à la suite des autres pour remplir une pile P. **Question 1.3**

Question 1.5 fonction evaluation

- si $L[i]$ est un nombre alors on l’empile dans P
- si $L[i]$ est un opérateur alors on évalue l’opération entre les deux premiers éléments de la pile (que l’on dépile), et on empile le résultat **Question 1.4 fonction calcul**

- à la fin du déroulement de la pile le résultat est au sommet de la pile

1.1. Compléter le cadre avec l’expression de $x = (3-2)* 2$ en notation postfixée

Rappel sur les priorités des opérateurs arithmétiques

- * et / sont plus prioritaires que + et –
- = est le moins prioritaire

Les différentes notations possibles

- Notation infixé : $a + 1$
- Notation préfixée : $+ a 1$
- Notation postfixée : $a 1 +$

1.2. Inscrivez l’expression postfixée dans une liste « expression »

Retirer le x des éléments de la liste car x est le résultat quand la Pile est dépilée

1.3. Ecrivez le code permettant de remplir une Pile «Mon op » avec la liste « expression ». Faites afficher le résultat. Ce principe sera réutilisé dans la question suivante.

TP – Pile et Files 1 Evaluation d’une expression mathématique postfixée

1.4. Compléter la fonction calcul qui traduit la présence d'un opérateur au moment où on dépile un opérateur

```
def calcul(pile,op) :  
    b =  
    #print(pile)  
    a =  
    #print(pile)  
    if op == '*' :  
        result =  
    elif op == '+' :  
        result =  
    elif op == '-' :  
        result =  
    elif op == '/' :  
        result =  
    elif op == '**' :  
        result =  
    elif op == "=" :  
        result =  
    pile.  
    #print(pile)
```

1.5. Compléter les parties soulignées de la fonction évaluation qui renvoie la valeur numérique du résultat de l’expression mathématique postfixée.

```
def evaluation(expression):  
    m_p = Pile()  
  
    operateurs = liste de caractères pour les différents opérateurs  
  
    for elt in expression : # parcours des éléments de expression  
        if appartenance à la liste des opérateurs:  
            traitement adéquat  
        else:  
            traitement adéquat  
  
    fin = m_p.?  
  
    return fin
```

1.6. Appliquer la fonction evaluation à la liste qui traduit l’expression mathématique en post fixe . Vérifier le résultat

1.7. Même chose pour : $x = (1 + 2) * (3 - 4/5^2)$