

### **Série 3 TD Structures de données en C : les piles, les files et les arbres**

#### **Exercice 1**

- 1) Si partant d'une pile P vide, on ajoute (en empilant) les entiers 1 puis 2 puis 3 puis 4 puis 5 et que en suite on supprime (par dépilement) deux éléments aux sommets quels entiers contient-elle la pile ?
- 2) Les mêmes questions avec une file F (utiliser les fonctions d'enfilement et de défilement de la file à la place des fonctions d'empilement et de dépilement de la pile).

#### **Exercice 2**

Soit à évaluer l'expression  $((A+B)*(C-D))$ , avec  $A=4$  ;  $B=2$  ;  $C=5$  ;  $D=3$ , à l'aide d'une pile :

On appel sous expression : partie d'expression délimitée par un couple de parenthèses (): on aura donc 3 sous expressions dans l'exemple. Les règles d'évaluation sont

- Expression lue de gauche à droite, pile gérée en parallèle :
- Si opérateur, on l'empile
- Si variable, on empile sa valeur
- Si parenthèse gauche, on l'empile (début de sous expression)
- Si parenthèse droite, on dépile jusqu'à une parenthèse gauche, on évalue la sous expression et on empile le résultat.

Représenter l'état de la pile d'évaluation de l'expression précédente ?

#### **Exercice 3**

- 1) Représenter l'arbre binaire de recherche, initialement vide, obtenu en insérant successivement les entiers suivants : 7, 2, 9, 0, 5, 6, 8 et 1.
- 2) Afficher le niveau de chaque nœud.
- 3) Afficher le contenu de cet arbre par le parcours infixe.
- 4) Afficher le contenu de cet arbre par le parcours préfixe.
- 5) Afficher le contenu de cet arbre par le parcours postfixe.
- 6) Donner la représentation graphique de l'arbre obtenu après l'insertion de l'entier 3.
- 7) Déterminer la hauteur de l'arbre obtenu.