Etablissement : ISET-Charguia	**Département :** Technologies de l'Informatique
Matière : Atelier programmation C - 2	**Année Universitaire :** 2017- 2018
Niveau : TI 1ère année	

TP n°3 : Les Piles et les Files

Exercice 1:

Une structure de pile doit posséder quelques fonctions particulières :

- creerPile() qui crée une pile vide et la renvoie ;
- empiler(p,v) qui positionne la valeur v au sommet de la pile p ;
- depiler(p) qui renvoie la valeur du sommet de la pile p et supprime l'élément correspondant ;
- estVide(p) qui renvoie True si la pile p est vide, False sinon ;
- sommet(p) qui renvoie le sommet de la pile sans le supprimer,

Créer les fonctions de base permettant la gestion des piles.

Exercice 2:

1) Égalité entre deux piles : écrire une fonction egalite(p1,p2) qui retourne True si les deux piles sont identiques et False sinon.

2) Couper le sommet d'une pile : écrire une fonction decouper(p,i) qui retourne une nouvelle pile contenant la liste des i-èmes derniers éléments de p .

3) Empiler deux piles : écrire une fonction empiler_les_piles(p1,p2) qui entasse dans le bon ordre la pile p2 sur la pile p1 .

Exercice 4:

On souhaite réaliser la fonction, présente dans votre éditeur de script, d'association des parenthèses.

Pour commencer, on considère une chaîne de caractère ne comportant que des parenthèses ouvrantes et fermantes. On souhaite déterminer s'il s'agit d'un mot est bien parenthésé : soit le mot vide, soit la concaténation de deux mots bien parenthésés, soit un mot bien parenthésé mis entre parenthèses. Ainsi ", '()()' et '(()) ()' sont des mots bien parenthésés. Mais les mots '(()', '())' ou encore ')(' ne le sont pas. On souhaite de plus afficher les correspondances entre parenthèses, sous la forme de couples d'indices. Par exemple, le mot '(())()' doit donner '(0, 3), (1, 2), (4, 5)'.

1) Réaliser la fonction par(expr) qui affiche les différents couples et qui renvoie True si l'expression est correctement parenthésée.

2) Si l'on ne voulait pas fournir les couples d'indices mais uniquement renvoyer un booléen, que devrait-on simplifier ? La structure de pile serait-elle toujours nécessaire ?

3) Adapter la fonction pour qu'elle puisse traiter des mots constitués de parenthèses et d'autres caractères, qui n'interfèrent pas avec les parenthèses (nombres et opérateurs).

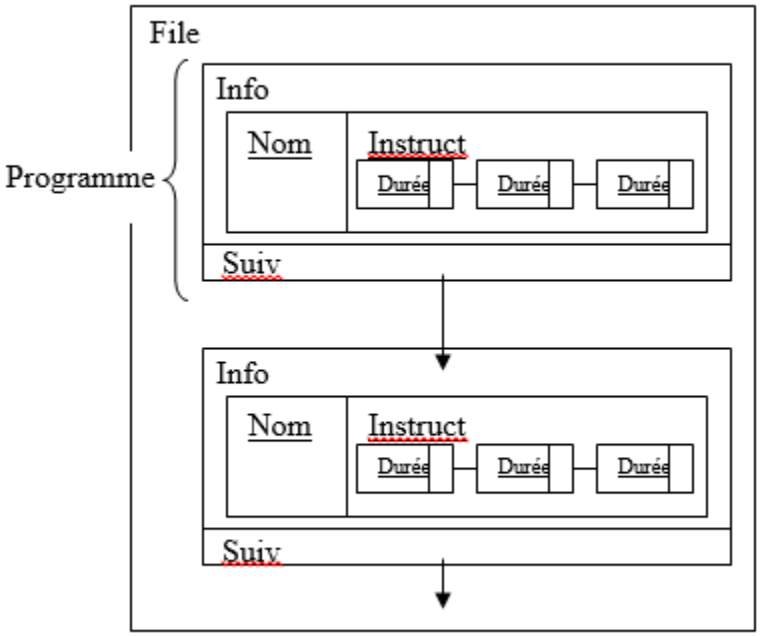
4) Adapter la fonction pour qu'elle puisse traiter des mots constitués aussi de crochets et d'accolades.

Exercice 5:

On désire gérer une file de programmes sachant que chaque programme est caractérisé par son nom et une liste d'instructions dont chacune est caractérisée par sa durée.

Tout au long de l'exercice, on utilisera une représentation chaînée des données c'est-à-dire de la file de programmes et de la liste de durées de chaque programme.

On aura ainsi les structures de données suivantes :



- Déclarer les enregistrements nécessaires.

- Ecrire la procédure Ajout_Instruction (var P : Programme, D : Réel) qui permet d'ajouter une instruction à un programme en début de sa liste d'instructions.

- Ecrire la fonction itérative Durée_Programme (P : Programme) : Réel qui retourne la durée d’un programme c'est-à-dire la sommes des durées de ses instructions.

- En utilisant la fonction Durée_Programme et les procédures Enfiler et Défiler, écrire la fonction itérative Durée_totale (F : File) : Réel qui retourne la somme des durées de tous les programmes en gardant la file d’origine.

- Ecrire la fonction récursive Durée_totale (F : File) : Réel qui retourne la somme des durées de tous les programmes sans penser à garder la file d’origine.