

TD n0 1 : Programmation avancée et Méthodes Formelles

Programmation avancée

Exercice 1

Soit la pile P définie à partir du tableau suivant :

	0	1	2	3	4	5	6	7
P	14	5	1	9				

1. Quel est le nombre d'éléments de P
2. Déterminez le sommet de P et quelle est sa valeur ?
3. Dessinez l'état de la pile P après les appels Empiler(P,16) et Empiler(P,2)
4. On fait ensuite appel à Depiler(P). Quel est le résultat retourné.
5. Dessinez l'état de la pile après appel à Depiler(P). Quel est le sommet de P et quelle est sa valeur ?

Exercice 2

Ecrire des algorithmes qui implémentent les opérations suivantes sur les piles.

1. Pile_vide
2. Empiler(P,x)
3. Depiler(P)

Exercice 3

On considère un algorithme qui vérifie qu'un texte contenant des caractères standards est syntaxiquement correct du point de vue des parenthèses. Les parenthèses sont de 3 types (, [et { et leurs parenthèses fermantes correspondantes sont respectivement),], et }. La correction syntaxique implique qu'à chaque parenthèse ouvrante corresponde, plus loin dans le texte, une parenthèse fermante du même type. Le texte compris entre ces deux parenthèses doit également être correct au point de vue des parenthèses (une parenthèse ouverte doit y être refermée).

Donner une version non récursive de l'algorithme de correction syntaxique précédent.

Donner en une version récursive.

Exercices 4

1. On lit une suite d'entiers positifs. En constitué une liste dont les valeurs sont triées par valeurs décroissantes.
2. La liste étant construite, lire un entier $x \geq 0$ et trouver s'il existe un élément dont la valeur y est la plus grande valeur $\leq x$, dans l'affirmative écrire dans l'ordre croissant les valeurs de la liste $[y, 2y]$

Exercice 5

Soit la file F définie à partir du tableau suivant :

	0	1	2	3	4	5	6	7	8	9	10	11
F							14	5	8	7	4	

1. Quel est le nombre d'éléments de F et quels sont leurs emplacements ?
2. Quels sont la tête et le sommet de cette liste et quelles sont leurs valeurs ?
3. Dessinez l'état de la file F après les appels Enfiler(F,16) et Enfiler(F,2)
4. On fait ensuite appel à Défiler(P). Quel est le résultat retourné.

Exercice 6

Ecrire des algorithmes (Fonction ou procédure en LDA) qui implémentent les opérations suivantes sur les Files.

1. **init_list** : initialise la liste à la liste vide
2. **hors_liste** : VRAI si *ec* est hors de la liste FAUX sinon
3. **liste_vide** : VRAI si la liste est vide FAUX sinon
4. **entête** : place l'élément courant sur la tête de la liste
5. **en_queue** : place l'élément courant sur la queue de la liste
6. **suc** : déplace l'élément courant sur son successeur
7. **pred** : déplace l'élément courant sur son prédécesseur
8. **val_ec**(var *val* :t) : reçoit la valeur de l'élément courant
val \leftarrow valeur de *l'ec*
9. **modif_ec**(*val* :t) : Modifie l'élément courant
10. **ajoutg(val :t)** : ajoute un élément dans la liste à gauche de *ec*
11. **ajoutd**(*val* :t)
12. **ote_ec**: supprime l'*ec* de la liste
13. **vider_liste** vider la liste.