Centre Universitaire de Mila
Institut des sciences et de la technologie
2eme année LMD informatique

Module A.S.D.D.2

Série de TD N 04
Arbres Binaires de recherche

Exercice 01 : écrire les procédures qui permettent d'afficher un arbre binaire d'entier dans le cas d'un parcours infixé et post-fixé.

Exercice 02 : donner la déclaration d'un arbre binaire d'entier et écrire les opérations suivantes sur cet arbre.

1) Taille (a) : qui retourne la taille de l'arbre.
2) Hauteurs (a) : retourne la hauteur de l'arbre.
3) Nb_feuilles (a) : qui retourne le nombre de feuilles de l'arbre a.
4) appartient (x, a) : qui vérifie si x existe dans a ou non.
5) max (a) : qui retourne le maximum de l'arbre a.
6) père (a, N) : qui retourne le père d'un nœud identifié par son adresse.

Exercice 03:

a. Compréhension
   1) Construire un arbre binaire de recherche à partir des clés ordonnées suivantes :
      25 60 35 10 5 20 65 45 70 40 50 55 30 15
   2) Ajouter à l'arbre obtenu et dans l'ordre, les éléments suivants :
      22 62 64 4 8
   3) Supprimer de l'arbre obtenu et dans l'ordre les éléments suivants :
      15 70 50 35 60 25
b. Écrire sur les arbres binaires de recherche les fonctions et les procédures récursives suivantes :
   1) Appartient (x, a) : qui vérifie si x existe dans a ou non.
   2) Max (a) : qui retourne le maximum de l'arbre a.
   3) Insérer (a, x) : permettant d'insérer un élément dans l'arbre a.
   4) Afficher_pairs (a) permettant d'afficher les éléments pairs de **a** trié dans un ordre croissant.
   5) Père (a, N) : qui retourne le père d'un nœud identifié par son adresse.

Exercice 04 : supplémentaire

1. Écrire les opérations suivantes sur les arbres binaires :
   - Somme_positif (arbre a) : qui retourne la somme des éléments positifs des éléments de l'arbre a.
   - Est_positif (arbre a) : qui retourne *true* si l'arbre a est non vide et tous ces éléments sont positifs, sinon retourne *false*.
2. Récrire les opérations précédentes sur les arbres binaires de recherche.

Exercice 05:

Une supérette désire disposer d'un outil automatique pour traiter les informations concernant son stock. On vous propose de représenter ces informations sous forme d'un arbre binaire de rechercher. Chaque nœud de l'arbre contient un produit représenté par sa référence (entier), sa désignation (chaine de caractères), son prix unitaire (réel) et la quantité disponible (réel), date de fabrication et date d'expiration. Les éléments sont insérés selon leurs références.

1. Définir les types de données permettant de représenter ce stock.
2. Définir les opérations permettant de saisir et d'afficher les informations d'un produit.
3. Ecrire la fonction permettant d'insérer un élément dans cet arbre.
4. Écrire la procédure crée (a : arbre, n :entier) permettant de créer un arbre de produits en lisant les informations à partir du clavier ou à partir d'un fichier.
5. Écrire la procédure Vendre (a, Ref, Quantite) permettant de retirer, si possible, la quantité 'Quantité' du produit qui a pour référence 'Réf'. La procédure doit afficher un message d'erreur lorsque la vente est impossible.
6. Écrire la procédure Alimenter (L, Ref, Quantite) permettant d'ajouter la quantité 'Quantité' au produit qui a pour référence 'Ref'. Si le produit n'existe pas, la procédure doit demander la désignation et le prix du produit et ensuite ajoutera ce produit dans la liste.
7. Ecrire la procédure expiration (a, d) permettant d'afficher les produit qui expirent avant une date donnée 'd'.
8. Refaire la question 5 et 6 en considérons maintenant que **a** est un arbre binaire simple.

Exercice 06 : supplémentaire

Refaire l'exercice 03 en écrivant des versions itératives pour les fonctions proposées.

Exercice 07 : supplémentaire

En utilisant une pile, écrire la procédure du parcours préfixe d'un arbre binaire.

Exercice 08 : supplémentaire

Écrire la fonction permettant de créer un arbre binaire équilibré à partir d'un tableau d'entiers trié dans un ordre croissant.