Programmation et Algorithmique II

Devoir 1 : Structure de donnée contiguë et liste chaînée

### OBJECTIF 1:

On cible par ce devoir l’implémentation la notion mathématique de l’ensemble en utilisant un tableau de *n* éléments de type entier. Ecrire un programme pour effectuer des opérations sur les ensembles : - Union, Intersection, etc.

### Définition :

Un ensemble est une structure de données de collection qui stocke certaines valeurs de manière à ce que les valeurs ne soient pas répétées. Selon que ces valeurs sont stockées dans un ordre ou non, l'ensemble est appelé ensemble ordonner ou ensemble non ordonné. C'est une implémentation du concept mathématique d'ensemble fini.

### Tâches à réaliser

En utilisant la structure de donnée dynamique suivante :

**typedef** struct **{**

int NbEMax**;** // nombre d'élément maximum de l'ensemble

int NbE**;** // nombre d'élément présents dans l'ensemble

int **\***tab**;** // conteneur des éléments de l'ensemble

**}** Ensemble **;**

1. Ecrire une fonction ou procédure pour allouer dynamiquement un espace mémoire pour un tableau composé de n éléments
2. Ecrire une fonction ou procédure pour allouer de manière dynamique l’espace mémoire de la structure Ensemble
3. Ecrire une fonction pour libérer l’espace mémoire réserver pour un Ensemble donné est ces éléments
4. Ecrire une procédure pour afficher les éléments d’un ensemble donné
5. Ecrire une fonction pour créer un ensemble à partir d’un tableau, sachant que la répétition d’élément est autorisée dans les tableaux et non autorisé dans les ensembles.
6. Ecrire une fonction pour insérer un élément donné dans un ensemble donné, sachant que l’ensemble est
   1. Non-ordonné
   2. Ordonné
7. Ecrire une fonction qui recherche un élément donné dans un ensemble donné. La fonction renvoie 1 si l’élément recherché est existe dans l’ensemble non-ordonné sinon il renvoi 0.
   1. Pour un ensemble non-ordonné
   2. Pour un ensemble ordonné
8. Ecrire une fonction qui reçoit comme entrée deux ensemble N1 et N2, ensuite elle renvoie un ensemble avec les éléments de l’intersection des deux ensembles quand les deux ensembles sont
   1. Non-ordonné
   2. Ordonné
9. Ecrire une fonction qui reçoit comme entrée deux ensemble N1 et N2, ensuite elle renvoie un ensemble avec les éléments de l’union des deux ensembles quand les deux ensembles sont
   1. Non-ordonné
   2. Ordonné
10. Ecrire le programme principal pour test les fonctions demandés dans les questions précédents.

### Objectif 2 : Maitriser les trois types de liste chaînée.

### Taches à réaliser :

1. Ecrire la liste chaînée simple, circulaire, et double avec trois fichiers tête différents, avec les fonctions d’initialisation, insérer à la tête, à la queue, à la i-ème position, et supprimer à la tête, à la queue, à la i-ème position. Ecrire aussi une fonction d’affichage. (Vous pouvez travailler à base des codes des TPs précédents.)
2. Pour chaque type de liste chaînée, écrire une fonction qui supprime toutes les occurrences (si elle existe) d’un entier n dans une liste chaînée **triée**.

Exemple : 2, 3, 4 en jaune seront supprimé.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 2 | 3 | 3 | 4 | 4 |

1. Dans la fonction principale (main function), écrire les instructions pour tester tâche 1 et tâche2