

Structure de données Avancées
Licence 2 UPB
Travaux Dirigés 2 : Pointeurs-Piles-Files

Exercice 1 : On suppose que les déclarations suivantes ont été faites :

int a;

char tab[10];

Une expression avec pointeurs (resp. sans pointeurs) vous est donnée, donnez une réécriture de cette expression sans (resp. avec) l'usage explicite des pointeurs.

1. `*(&a)`
2. `*tab`
3. `*(tab + 0)`
4. `(*tab) + 1`
5. `&(tab [0])`
6. `&(tab[i])`

Exercice 2 :

Une pile est une structure de donnée qui enregistre des informations selon le mode dernier entré premier sorti (LIFO : Last In First Out). On manipule une pile en utilisant les quatre opérations suivantes :

- Création d'une nouvelle pile
- Test si la pile est vide
- Ajout d'un élément
- Suppression d'un élément

NB : Dépiler(p) est valide seulement si non EstVide(p).

- 1- En utilisant un tableau proposer une structure permettant de stocker une pile ; Dans un premier temps, on supposera que la taille est limitée.
- 2- Écrire précisément les invariants de la structure.
- 3- Écrire les quatre opérations précédentes.
- 4- Comment faire pour que la taille ne soit plus limitée sans perdre en complexité.

Exercice 3 :

- 1- Si partant d'une pile P vide, on ajoute (en empilant) les entiers 1 puis 2 puis 3 puis 4 puis 5 et qu'ensuite 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 4 :

Soit P une pile d'entiers. Écrire les fonctions pour déterminer :

- 1- Le nombre d'éléments.
- 2- La valeur maximale.
- 3- La valeur minimale.

Exercice 5 : Reprendre l'exercice précédent avec la structure de données files

Exercice 6 : Soit F une File d'entiers. Écrire une procédure qui permet de trier ses éléments selon un ordre croissant.