

**Série de TD N 04**  
**Piles et Files**

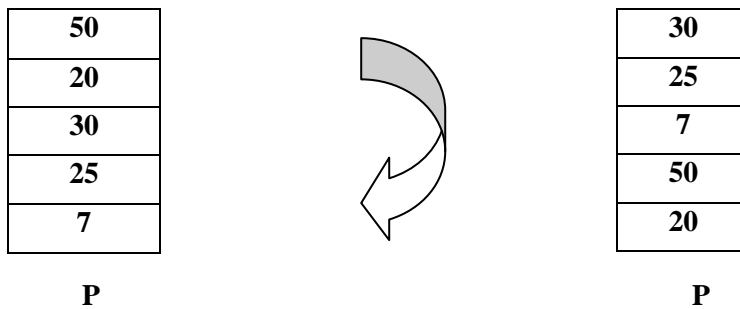
**Exercice 01 :** donner la déclaration d'une pile dynamique d'entiers et écrire en algorithmique les opérations permettant de créer une pile de N entier, d'afficher ses éléments pairs et de calculer la somme de ses éléments.

**Exercice 02 :** Refaire l'exercice 1 pour une file dynamique d'entiers.

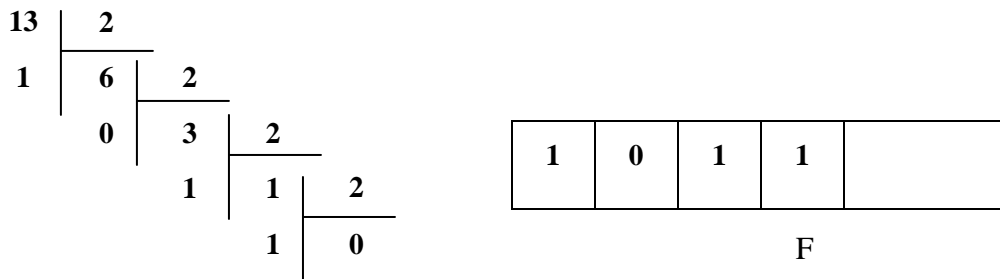
**Exercice 03 :** En utilisant une pile et ses primitives, écrire la fonction copier (L) permettant de créer une copie de la liste chaînée L.

**Exercice 04 (permutations circulaires):** écrire une fonction **permutation\_circ** qui reçoit en argument une pile P et un entier n ( $N \leq \text{taille}(P)$ ) et effectue sur la pile N permutations circulaires successives.

**Exemple avec N=2 :**



**Exercice 05 :** dans l'exemple sous-dessous la file F contient la représentation binaire du nombre 13 (1101).



1. Écrire la procédure *binnaire* qui reçoit un nombre décimal x et crée une File F contenant sa représentation binaire.
2. Écrire la fonction *décimale* qui reçoit une File F contenant la représentation binaire d'un entier x et retourne sa représentation décimale.

**Exercice 06 :** Nous voulons implémenter un programme pour calculer le temps d'impression approximatif d'un ensemble de documents enfilés dans une file d'attente d'une imprimante. Cette file d'attente est implémentée par une liste chaînée (file dynamique) où chaque maillon contient le nom d'un document avec le nombre de pages à imprimer.

- 1) Définir les types de données nécessaires pour l'implémentation de cette file.
- 2) Ecrire la fonction qui retourne le nombre de documents existe dans la file d'attente.
- 3) Si le temps d'impression d'une page est 5 Secondes, écrire la procédure qui affiche le temps nécessaire pour l'impression de tous les documents.

**Exercice 07 : supplémentaire**

En utilisant une pile, écrire une version itérative pour la fonction récursive puissance (x, n) en respectant l'algorithme de son exécution par les langages de programmation.

```
int puissance (int x, int n)
{ if (n==)
    return 1;
  else
    return x * puissance (x, n-1) ;
}
```

**Exercice 08 : supplémentaire**

Refaire l'exercice 6, en considérant maintenant que chaque document dans la file est représenté par le nom de document et une file de pages à imprimer.

**Exercice 9 : supplémentaire**

En utilisant une pile, écrire la fonction permettant d'évaluer une expression arithmétique post fixé, représentée par une liste chaînée, contenant seulement des chiffres (0,1,...9) comme opérandes et les opérateurs suivants : l'addition '+', la soustraction '-', la multiplication '\*', et la division '/'.

Exemple : 2 3 \* 8 +

