

CI 2 – ALGORITHMIQUE ET PROGRAMMATION

TP – ALGORITHMES DE TRIS

1 Principe

Lors de l'acquisition d'un grand nombre de données, il est souvent nécessaire de les trier suivant un critère donné. Par exemple, dans les cas où les données sont des nombres, il peut être nécessaire de les ordonner dans l'ordre croissant ou décroissant. Ainsi, il devient plus rapide de déterminer le plus grand ou le plus petit nombre de la liste.

Lorsque le nombre de valeurs devient important, il devient indispensable d'optimiser cette opération afin de réduire le temps d'exécution d'un programme.

2 Le tri par insertion

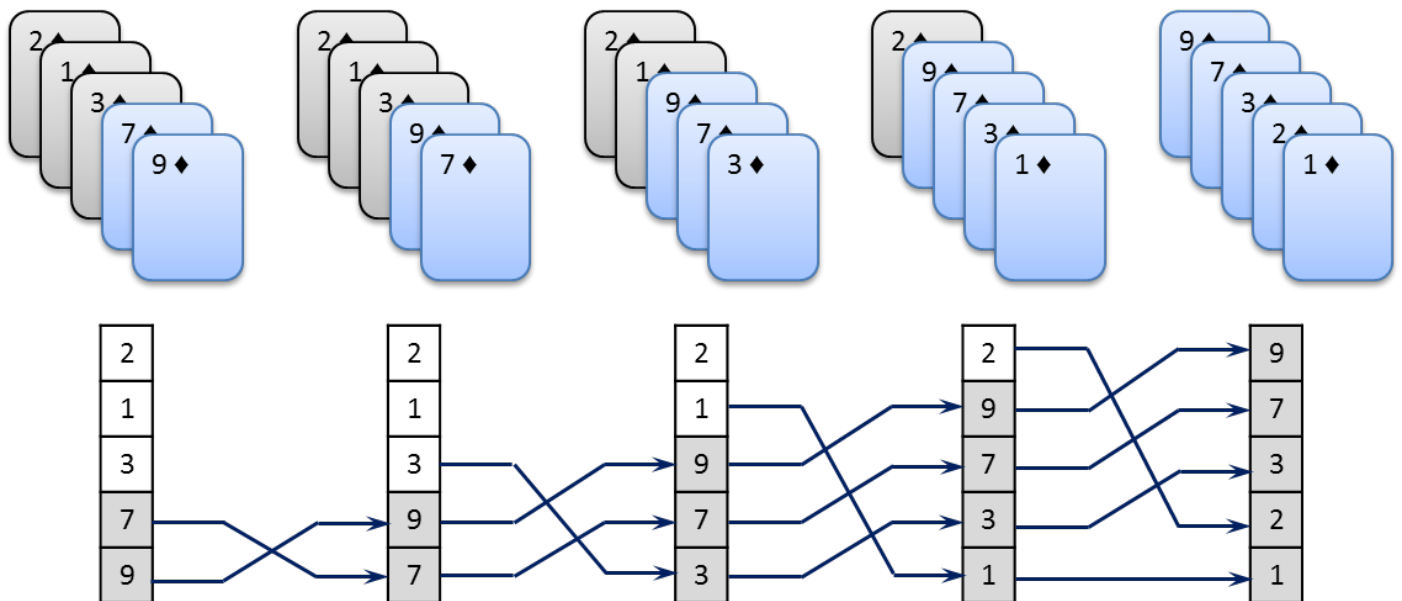
2.1 Principe

Le tri par insertion est celui utilisé naturellement pour trier sa main lorsqu'on joue au carte. Le joueur reçoit un certain nombre de cartes mélangées. Pour les trier, le joueur procède souvent ainsi :

1. le joueur prend la première carte ;
2. le joueur prend la seconde carte :
 - il la compare à la première carte :
 - si la carte 2 est plus petite que la première il la positionne en premier ;
 - sinon il la positionne en second ;
3. le joueur prend la troisième carte :
 - il la compare à la première carte :
 - si la carte 3 est plus petite que la première, il la positionne avant en premier ;
 - sinon, il compare la carte 3 à la seconde :
 - si la carte 3 est plus petite que la seconde, il la positionne avant en second ;
 - sinon, il la positionne en troisième.
4. *etc.*

2.2 Illustration

On peut illustrer ainsi le tri de 5 cartes :



2.3 Travail demandé

On donne en pseudo code un algorithme permettant de réaliser un tri par insertion :

Pseudo Code

```

Données : Tableau  $tab$  de  $N$  éléments

Début Fonction
  Tri par insertion ( $tab$ ) :
  pour  $i$  de 1 à  $N$  faire
     $a \leftarrow tab[i]$ 
     $j \leftarrow i - 1$ 
    tant que  $j \geq 0$  et  $tab[j] > a$  faire
       $tab[j + 1] \leftarrow tab[j]$ 
       $j \leftarrow j - 1$ 
    fin
     $tab[j + 1] \leftarrow a$ 
  fin
  retourner  $tab$ 
Fin
  
```