

J1IN3001 ASD1 Devoir surveillé n°3 Durée : 30mn Aucun document autorisé 10/12/2013	Nom : Prénom :
---	-----------------------

Question de cours :

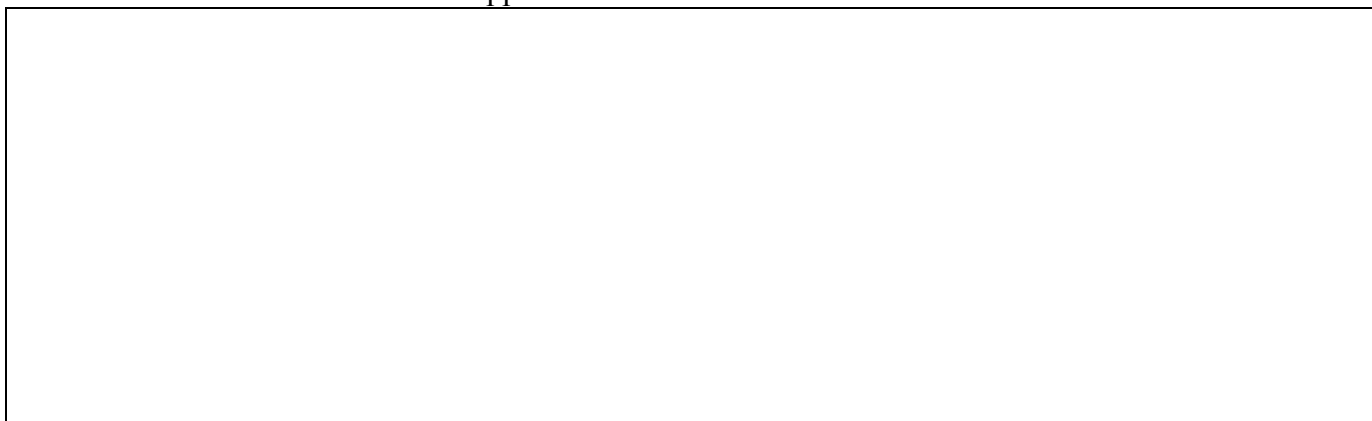
Lorsqu'on utilise une table de hachage chaînée comment gère-t-on les collisions ?

Exercice1 :

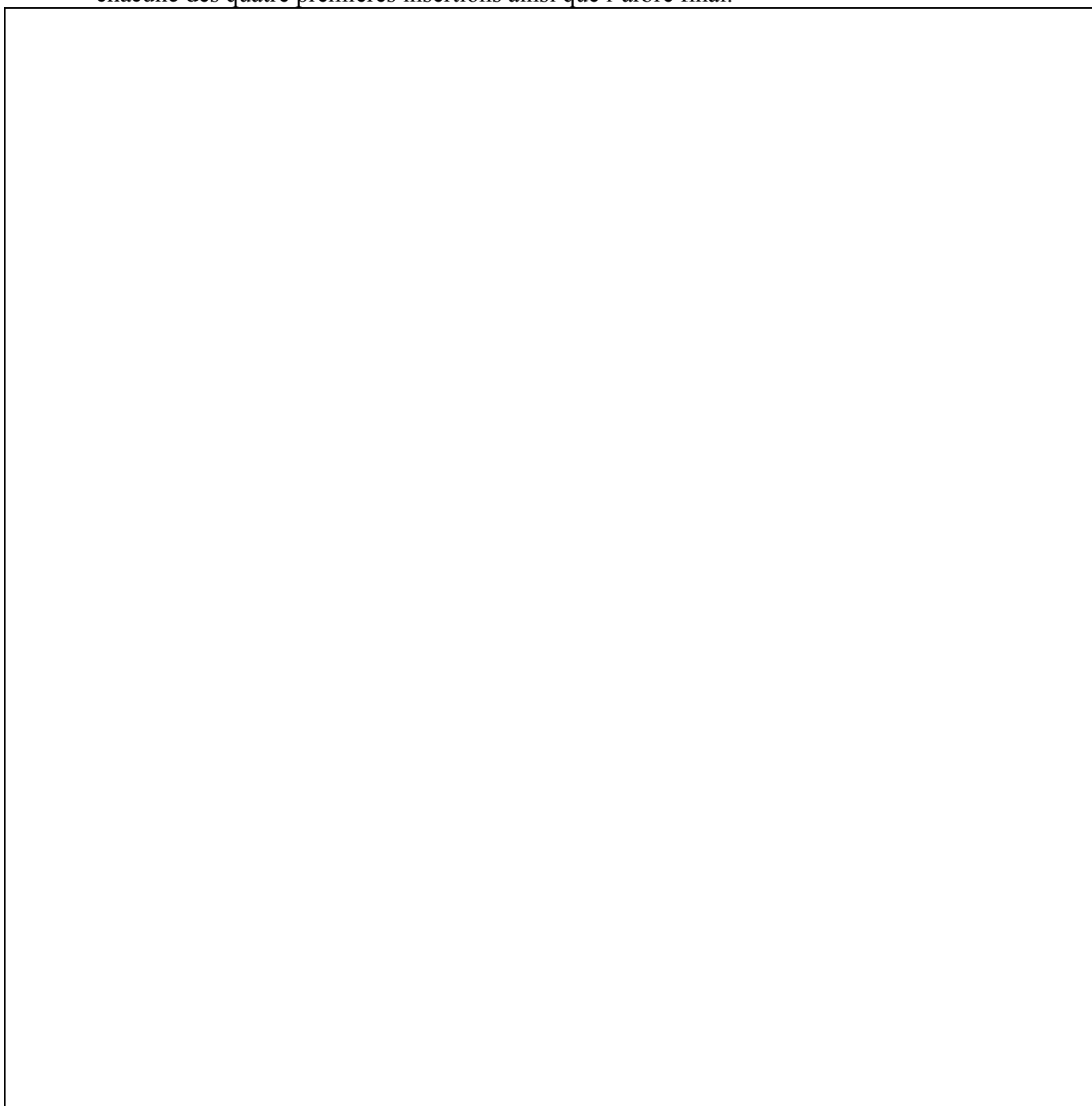
Soit la liste de clé $A=(12, 5, 3, 8, 10, 2, 6, 11)$.

1. Construire le tas Min T correspondant à l'insertion consécutive des clés de la liste A . On dessinera l'arbre après chacune des insertions.

2. Montrez l'exécution de la suppression sur l'arbre **T**.



3. Soit La liste de clés **B** = (10, 2, 14, 25, 7, 9, 16, 23, 5) Construire l'arbre binaire de recherche **B** correspondant à l'insertion consécutive des clés de la liste. On dessinera l'arbre après chacune des quatre premières insertions ainsi que l'arbre final.



Exercice2 :

1. Dans un arbre binaire de recherche (ABR) indiquez où se trouve le maximum des valeurs qui le composent ?

2. On appelle deuxième maximum la plus grande valeur présente dans l'ABR si on enlève le maximum. Où se trouve le 2^{ème} maximum dans un ABR ?

3. En modifiant l'algorithme de parcours en profondeur d'un arbre binaire et sans supprimer le maximum écrire une fonction `deuxiemeMaximum (ref A : AB)` qui retourne le sommet dont la valeur est le 2^{ème} maximum.

Arbres binaires

```
fonction getValeur(val S:sommet):objet;  
fonction filsGauche(val S:sommet):sommet;  
fonction filsDroit(val S:sommet):sommet;  
fonction pere(val S:sommet):sommet;  
fonction setValeur(ref S:sommet;val x:objet):vide;  
fonction ajouterFilsGauche(ref S:sommet,val x:objet):vide;  
fonction ajouterFilsDroit(ref S:sommet,x:objet):vide;  
fonction supprimerFilsGauche(ref S:sommet):vide;  
fonction supprimerFilsDroit(ref S:sommet):vide;  
fonction detruireSommet(ref S:sommet):vide;  
fonction créerArbreBinaire(val Racine:objet):sommet;  
fonction detruireArbreBinaire(val Racine:objet):sommet;
```

Tas.

```
fonction valeur(val T: tas d'objet): objet;  
fonction ajouter(ref T: tas d'objet, val v: objet): vide;  
fonction supprimer(ref T: tas d'objet): vide;  
fonction creerTas(ref T: tas d'objet, val v: objet): vide;  
fonction detruireTas(ref T: tas d'objet): vide;
```

Type abstrait file de priorité

```
Primitive supplémentaire :  
fonction changeValeur(ref T: tas d'objet, val s: sommet, val v:  
objet): vide;
```