Université Libanaise
Faculté des Sciences 1

INFO216 F

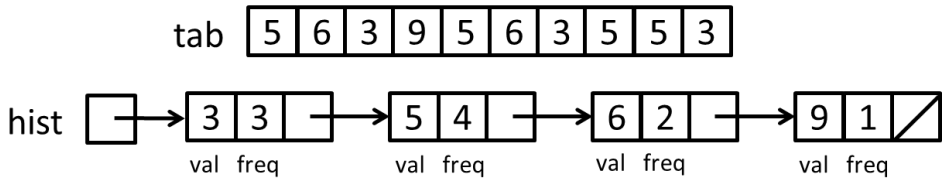
Programmation Impérative II

Final 11.12
Mer. 20 Juin am
Durée: 2 heures

Exercice I

Un histogramme est une présentation de données qui montre la fréquence avec laquelle chaque valeur est présente dans un tableau. On considère des valeurs entières.

Exemple



1. Définir le type de données "cel" pour représenter un histogramme comme une liste chaînée de valeurs (val) et fréquences (freq).
2. Écrire la fonction "histo" qui, étant donné un tableau de valeurs, retourne son histogramme. Noter que le tableau donné n'est pas trié, cependant, l'histogramme résultant doit être trié selon l'ordre ascendant des valeurs.
3. Écrire la fonction "histoTest" qui démontre l'utilisation de la function "histo".

Exercice II

On considère les types Point3d, Point2d et Noeud, définis ci-dessous. Un fichier binaire nommé "points3d" contient des données de type point3d.

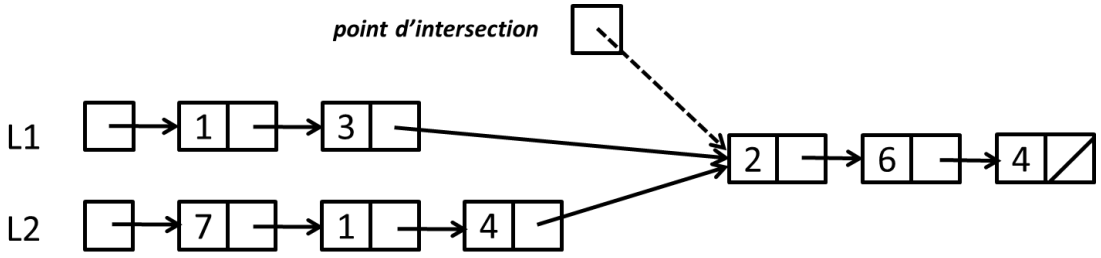
```
typedef struct Point3d { double x , y , z ; }  point3d ;
typedef struct Point2d { double x , y ;    }  point2d ;
typedef struct Noeud   { point2d p ; struct Noeud * suiv ; } noeud ;
```

4. Écrire la fonction "fich2liste" qui lit le fichier "points3d" et copie son contenu dans une liste chaînée de type noeud *. Manifestement, la coordonnée z sera négligée. L'ordre des points dans le fichier devrait être maintenu dans la liste. Le fichier et la liste doivent être parcourus, chacun, une seule fois.

Exercice III

5. Étant données deux listes chaînées d'entiers, écrire la fonction "intPoint" qui trouve et retourne leur point d'intersection. Noter qu'il ne s'agit pas de l'intersection ensembliste mais de nœuds communs.

Exemple



Fin