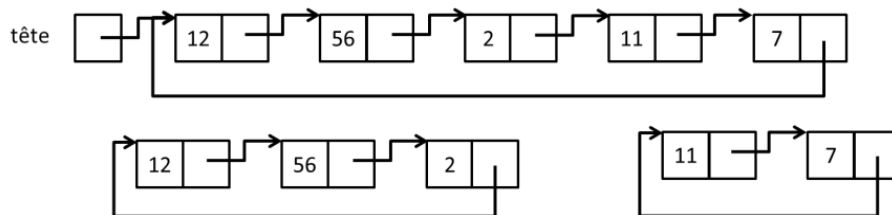


Exercice I [15 points]

Écrire une fonction "split" qui divise une liste chaînée circulaire donnée en deux listes éventuellement de longueurs égales (à un nœud de prêt) et renvoie leurs têtes à travers ses paramètres. Si la longueur de la liste donnée est impaire, le nœud du centre de cette liste termine la première liste obtenue; Sinon, les deux listes obtenues auront le même nombre de nœuds. Il n'est pas autorisé de supprimer des nœuds, ni d'en créer. Aussi l'ordre des nœuds dans la liste donnée doit être maintenu dans les deux listes obtenues. Le type "nœud" et le prototype de la fonction "split" sont définis comme suit :

```
typedef struct Noeud { int d; struct Noeud *suiv;} noeud;
void split(noeud *tete, noeud **t1Ref, noeud **t2Ref);
```

Exemple :



Exercice II [30 points]

Le type noeud est défini par :

```
typedef struct Noeud{ char d; int freq; struct Noeud *suiv;} noeud;
```

- Considérons une liste chaînée de type «noeud», où la valeur du champ "freq" est 1 pour tous les nœuds. Écrire la fonction "encode" qui condense chaque groupe de noeuds répétés successifs en un seul noeud, affecte le nombre de répétitions à la valeur de son champ "freq" et supprime les autres. Aucune allocation n'est autorisée dans cette tâche.

Exemple :

avant : tête → a,1 → a,1 → a,1 → c,1 → c,1 → d,1 → d,1 → d,1 → d,1 → a,1 → a,1
après : tête → a,3 → c,2 → d,4 → a,2

- Écrire la fonction "decode" qui lit un fichier compressé, décompresse son contenu et l'affiche sur l'écran, et retourne le taux de compression. Le taux de compression est le rapport du nombre de caractères avant et après la compression. On suppose que la valeur de fréquence d'un caractère ne dépasse pas 9.

Exemple :

contenu du fichier : a3c2d4a2
sortie sur l'écran : aaaccdddaa
taux de compression = 11/8 = 1.375

Exercice III [25 points]

Le type noeud est défini par :

```
typedef struct Noeud { int d; struct Noeud *suiv, *arbit;} noeud;
```

On considère une liste chaînée dont chaque noeud contient un pointeur arbitraire supplémentaire, appelé "arbit". Le pointeur arbit peut être NULL, ou peut pointer vers un nœud de la liste. Écrire une fonction "clone" pour créer une copie exacte de cette liste.

Exemple : dans la figure ci-dessous, nous représentons le pointeur «suiv» à l'aide d'une ligne continue et le pointeur «arbit» à l'aide d'une ligne pointillée.

