

# EXAMEN

Sections: LGLSI1, LIRIS1

Épreuve d': Algorithmique, structures de données et complexité

Nature de l'épreuve:	DC <input checked="" type="checkbox"/> DS <input type="checkbox"/> EF <input type="checkbox"/>	Documents:	autorisés <input type="checkbox"/> non autorisés <input checked="" type="checkbox"/>
Date de l'épreuve:	01/04/2024	Session:	principale <input checked="" type="checkbox"/> contrôle <input type="checkbox"/>
Durée de l'épreuve:	1H00	Enseignant:	Fethi Mguis

### Exercise 1 : (10 pts)

Ecrire un algorithme utilisant le formalisme pointeur pour remplir deux tableaux par des entiers, calculer et afficher leur schtroumpf. Pour calculer le schtroumpf de deux tableaux, il faut multiplier chaque élément du premier tableau par chaque élément du deuxième tableau et additionner le tout.

Example :

Tableau 1 : 

5	4	7	2	6
---	---	---	---	---

Tableau 2 :

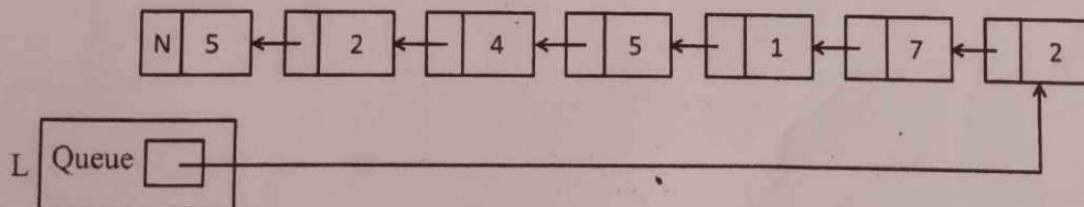
2	8	1
---	---	---

Le schtroumpf=  $5*2+5*8+5*1+4*2+4*8+4*1+7*2+7*8+7*1+2*2+2*8+2*1+6*2+6*8+6*1=264$ .

**Exercise 2 :** (10 pts)

On considère une liste simplement chaînée d'entiers. La liste est identifiée par l'adresse de son dernier élément (Queue). Chaque cellule contient un champ Val de type entier et un champ Prec qui est un pointeur vers la cellule précédente.

Example :



Une telle liste peut être représentée par les types suivants :

Type

Cellule : Enregistrement

Val : entier

Prec : \*Cellule

Fin Enreg

Liste :Enregistrement

Queue : \*Cellule

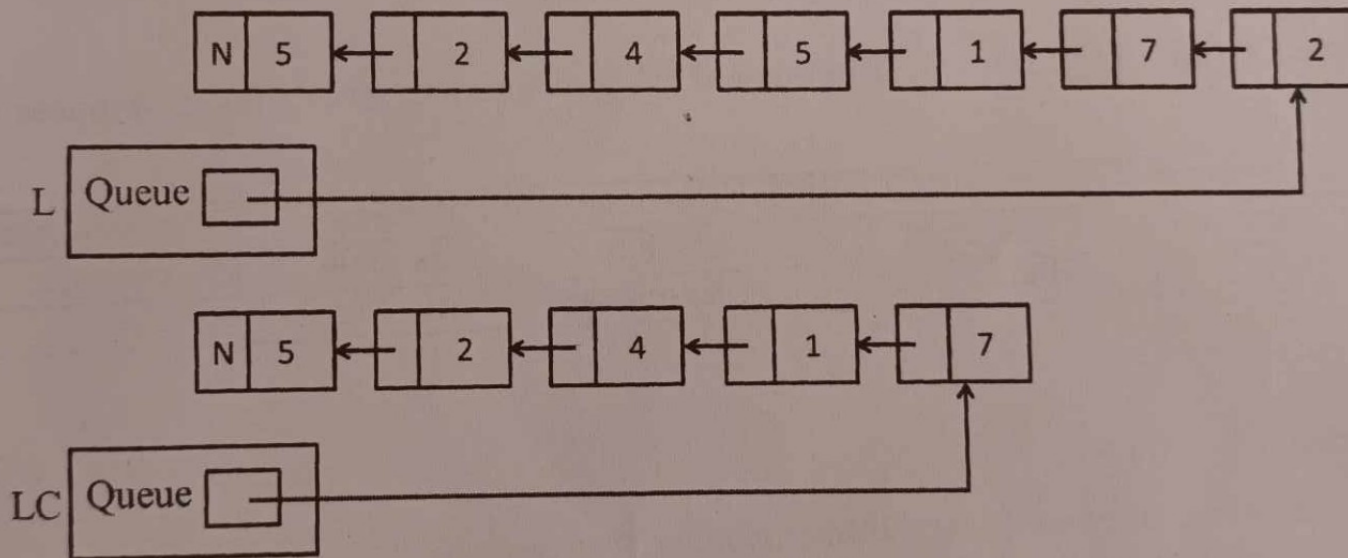
Fin Enreg

Nous supposons également que nous avons à notre disposition les fonctions et procédures suivantes :

- La procédure **Init(L :\*Liste)**, permettant de créer une liste vide.
- La fonction **EstVide(L :Liste) :Booléen**, permettant de vérifier si la liste L est vide ou non.
- La fonction **Recherche(L :Liste, X :Entier) :Booléen**, permettant de vérifier si l'élément X est présent dans la liste L ou non.

- ✓ 1. Ecrire une procédure **Ajout\_Queue** : permettant d'ajouter un élément en queue d'une liste.
- ✓ 2. Ecrire une procédure **Ajout\_Tête** : permettant d'ajouter un élément en tête d'une liste.
- ✓ 3. Écrire une procédure **Copie** : permettant de copier, sans redondance, les éléments d'une liste dans une autre liste.

**Exemple :**



- 4. Ecrire une procédure **Afficher** : permettant d'afficher les éléments d'une liste (Tête en premier).
- 5. Écrire un algorithme principal qui fait appel aux procédures ci-dessus pour remplir une liste L par N entiers, copier la liste L en une liste LC et afficher les deux listes.

*Bon Travail*