



ISIMM

Examen – S2 – 2024/2025

Filière : 1 ^{ère} LGL	Matière : Atelier de programmation 2		Enseignants : S.R Taoufik & B.S Kais
Date : 23/05 / 2025	Nbr de Crédits : 7	Coefficient : 1	Documents autorisés : Non
Durée de l'examen : 1h30	Régime d'évaluation : Mixte		Nombre de pages : 04
	EX (70%) + DS (10%) + TP (20%)		
Nom & Prénom :			Matricule :
Signature :	Code confidentiel :		

NOTE : Répondre directement sur les feuilles de l'examen

Note

/20

Exercice 1 :

Soit la définition de la Structure de Données Liste Linéaire Unidirectionnelle suivante:

```
struct cellule
{
    int cle;
    struct cellule * suivant;
};
struct liste
{
    struct cellule * premier;
    struct cellule * dernier;
};
```

Indication : Vous pouvez utiliser les fonctions ci-dessous sans les ré-implémenter.

- ✓ void creer_liste(struct liste *L)
- ✓ int liste_vide(struct liste L)
- ✓ void inserer_apres_dernier (struct liste * L , int x)
- ✓ void inserer_avant_premier (struct liste * L , int x)
- ✓ void inserer_apres_reference(struct cellule * p, int x)
- ✓ void inserer_avant_reference(struct cellule * p, int x)
- ✓ struct cellule * recherche_reference (struct liste L , int x)

On vous demande d'écrire les fonctions suivantes :

N. B : Toutes les questions sont indépendantes

Ne rien écrire ici

- 1- Une fonction de prototype « **struct liste Inverser_Liste (struct liste L)** » : Cette fonction crée et renvoie une nouvelle liste **Linv** qui contient les mêmes éléments que L, mais dans l'ordre inverse. La liste L d'origine ne doit pas être modifiée. (3 points)

- 2- Une fonction de prototype « **unsigned Est_Triee(struct liste L)** » : Cette fonction retourne 1 si la liste est triée en ordre strictement croissant, 0 sinon. (3 points)

- 3- Une fonction de prototype « **struct liste Intersection_Liste (struct liste L1, struct liste L2)** » : Cette fonction crée et renvoie une nouvelle liste contenant uniquement les éléments qui sont présents à la fois dans L1 et dans L2. La nouvelle liste ne doit pas contenir de doublons. (3 points)

4- Une fonction «**struct cellule * Adresse_Dernier_Element_Impair (struct liste L)**»: Cette fonction retourne l'adresse de la dernière cellule contenant une valeur impaire dans la liste L. Elle renvoie NULL si aucune valeur impaire n'est présente dans L. (3 points)

5- Une fonction ayant pour prototype «**void supprimer_tous_elements_pairs(struct liste *L)**» : Cette fonction supprime tous les éléments pairs dans L. (NB : sans utiliser les fonctions vues en cours et traiter tous les cas possibles.)
(4 points)

--

