

TP6 : L'univers impitoyable des pointeurs

C Seconde Langue

Chances are the baby's yours. I've been just as faithful to our marriage vows as you have been, darling.

— Sue Ellen to J.R.

★ **Objectifs pédagogiques.** À la fin de ce TP vous saurez :

- Manipuler les pointeurs en C
- Modéliser un problème sous forme de pseudo-objets C (des pointeurs sur structure)
- Débugger des problèmes d'allocation mémoire (aidez-vous de valgrind et gdb)
- Comprendre les séries téléés d'un autre âge

★ Introduction

On a retrouvé à la mairie de la petite ville américaine de Southfork Ranch un registre de déclaration des naissances au contenu pour le moins sommaire :

JR Ewing Jr.	;	Jock Ewing	;	Miss Ellie
Gary Ewing	;	Jock Ewing	;	Miss Ellie
Bobby Ewing	;	Jock Ewing	;	Miss Ellie
Tina Ewing	;	JR Ewing Jr.	;	Sue Ellen Shepard
JR Ewing III	;	JR Ewing Jr.	;	Sue Ellen Shepard
Lucy Ewing	;	Garry Ewing	;	Valene Clements
Christopher Ewing	;	Bobby Ewing	;	Pamela Barnes
Ray Krebs	;	Jock Ewing	;	Margaret Hunter
Margaret Krebs	;	Ray Krebs	;	Donna Culver
Lucas Ewing	;	Ray Krebs	;	Jenna Wade

Après un gros effort de recherche, on a fini par comprendre l'histoire de ces personnes, bientôt relatée dans un feuilleton qui déchaîne les passions du monde tout entier.

Si vous avez raté les 5683 premiers épisodes, consultez votre magazine télé préféré qui vous résume l'histoire (en «s'inspirant» honteusement de Wikipédia) :

Jock et Miss Ellie ont eu trois fils, J.R., Gary et Bobby. J.R., le fils Ewing aîné sans scrupule était marié sans amour à une ancienne Miss Texas, Sue Ellen Shepard Ewing. Il était fréquemment en conflit avec son plus jeune frère, Bobby, garant de la morale et l'intégrité qui manquait tant à son frère aîné. Gary, le second fils, était le mouton noir de la famille. Longtemps mal aimé de Jock et mal traité par J.R., il eut une relation profonde quoique distante avec Bobby et Ellie. Il s'est marié avec Valene "Val" Clements Ewing, qui mis au monde une fille, Lucy, la nièce impertinente mais compliquée de J.R. et Bobby. Elle passa la plus grande partie de sa vie à Southfork avec ses grands-parents. Elle eut une affaire avec le contremaître du ranch Ray Krebs. Ray s'avéra être son demi-frère, issu d'une aventure extra-conjugale de Jock avec Margaret Hunter, une infirmière rencontrée par Jock pendant la seconde guerre mondiale. Ray se maria finalement avec Donna Culver, et une enfant du nom de Margaret Krebs naquit de leur union, bien que le divorce fut prononcé bientôt. Plus tard, Ray se maria à Jenna Wade, qui donna bientôt naissance à un petit Lucas.

En attendant l'épisode suivant, vous avez déjà traduit l'histoire par le graphe présenté plus loin.

★ Exercice 1: Comprendre le problème.

Entre deux épisodes, vous décidez de développer un outil informatique qui vous permettra de suivre en temps réel les évolutions démographiques des 63521 épisodes restants. Pour cela vous résoudrez les problèmes suivants.

▷ **Question 1:** Que signifient 6 références de chaque nœud dans le graphe suivant ?

Réponse

Dans le sens des aiguilles d'une montre, en partant d'en haut à gauche.

- Père
- Mère
- Frère ou demi-frère par le père (ou sœur)
- Frère ou demi-frère par la mère (ou sœur)
- L'un des enfants
- Chaînage contenant tous les nœuds existants

Fin réponse

▷ **Question 2:** À quoi sert le chaînage général de tous les nœuds (celui en pointillés) ?

Réponse

Il contient donc tous les éléments de la structure. Ça peut être utile pour chercher une personne d'après son nom, par exemple.

Fin réponse

▷ **Question 3:** Quel est l'intérêt des chaînages cycliques horizontaux de ce graphe ?

Réponse

Il s'agit donc des liens de fraterie. Il faut couper en deux car il y a une fraterie par le père, et une fraterie par la mère.

Fin réponse

Tant que vous n'avez pas compris les liens entre l'histoire, le registre et le graphe, n'allez pas plus loin et recommencez cet exercice.

★ Exercice 2: Réalisation de la structure correspondante.

▷ **Question 1:** Écrivez la structure décrivant une personne. Nous allons l'utiliser comme pseudo-classe en C.

Réponse

```
1 typedef struct node {
2     struct node *pere;
3     struct node *mere;
4     struct node *frere_pere;
5     struct node *frere_mere;
6     struct node *enfant;
7     struct node *next;
8     char *nom;
9     int ismale;
10 } node;
```

Fin réponse

▷ **Question 2:** Écrivez le constructeur de cette classe qui prendra comme argument le nom de la personne et son sexe. Ce nouveau nœud du graphe sera pour l'instant considéré comme n'ayant pas de parents, ni frère ni sœur (comme Jock dans l'exemple). On pourra considérer qu'il n'existe pas 2 personnes portant le même nom.

Réponse

```
1 node *node_create(char *name, int ismale) {
2     node *res=malloc(sizeof(struct node));
3     strcpy(res->name,name);
4     res->ismale = ismale;
5     res->pere = NULL;
6     res->mere = NULL;
7     res->frere_pere = res; // on boucle sur soi-meme
8     res->frere_mere = res; // on boucle sur soi-meme
9     res->enfant = NULL;
10    res->next=res; // on boucle sur soi-meme
11    return res;
12 }
```

Fin réponse

▷ **Question 3:** Écrivez la fonction permettant d'insérer un nouveau nœud dans la liste de tous les nœuds présents dans le graphe.

Réponse

```
1 node *allnodes=NULL;
2 void node_link(node *n) {
3     if (allnodes==NULL) {
4         allnodes=n;// premier noeud créé. C'est plus simple
5     } else {
6         // cherchons la fin du cycle (allnodes est la sentinelle)
7         node *iterator = allnodes;
8         iterator=allnodes;
9         while (iterator->next != allnodes)
```

```

10     iterator = iterator->next;
11     // Inserons n à la fin du cycle
12     n->next=allnodes;
13     iterator->next=n;
14 }
15 }

```

Fin réponse

▷ **Question 4:** Écrivez une fonction modifiant différentes références entre nœuds du graphe, lors de l'établissement d'une filiation père/enfant. Cette fonction aura deux paramètres : l'objet représentant le père et celui représentant l'enfant.

Réponse

```

1 void node_change_pere(node *pere,node*fils) {
2     if (pere->fils) { // il y a deja des enfants
3         // Cherchons le dernier fils (en utilisant pere->fils comme sentinelle)
4         node *last = pere->fils;
5         while (last->frere_pere != pere->fils)
6             last = last->frere_pere;
7         // On insere le nouveau avant le premier et après last
8         fils->frere_pere = pere->fils;
9         last->frere_pere = fils;
10    } else {
11        pere->fils=fils;
12        fils->pere=pere;
13    }
14 }

```

Fin réponse

▷ **Question 5:** Même question mais lors de l'établissement d'une filiation mère/enfant.

Réponse

Même genre de réponse (je n'ai rien tenté de compiler de toute façon, c'est une correction à l'arrache).

Fin réponse

▷ **Question 6:** Écrivez une fonction qui prend un nom en paramètre et qui renvoie l'objet correspondant (ou NULL si le nom n'existe pas dans le graphe).

Réponse

```

1 node*node_by_name(char *name) {
2     node *iterator = allnodes;
3     do {
4         if (!strcmp(iterator->name,name))
5             return iterator;
6         iterator=iterator->next;
7     } while (iterator != allnodes);
8     return NULL;
9 }

```

Fin réponse

▷ **Question 7:** Écrivez une fonction qui affiche l'état du graphe à un instant donné. Vous pouvez par exemple écrire une fonction qui, pour chaque nœud, écrit sur la sortie standard le nom de la personne et ceux de ses six voisins.

▷ **Question 8:** Écrivez une fonction qui lit un registre de naissances et crée le graphe correspondant. A cet instant du projet, vous devriez être capables de créer le graphe correspondant à l'épisode du jour. Vous êtes donc prêts pour les rebondissements à venir :

★ Exercice 3: Modifications à la structure (*attention spoiler!*)

▷ **Question 1:** Terrible rebondissement durant l'épisode de cette semaine! Lucas n'est pas le fils de Ray, mais de son demi-frère Bobby!! Écrivez une fonction qui prend en paramètre les noms de l'enfant et du *vrai* père et qui modifie le graphe en conséquence.

▷ **Question 2:** Encore un rebondissement sordide cette semaine : Tina Ewing n'est pas la fille de J.R. et Sue Ellen. En effet, Sue Ellen ayant eu peur que Pamela donne un héritier à la famille Ewing avant elle, elle avait acheté un bébé à une femme. Quand J.R. découvrit l'histoire (dès l'épisode 10), il renvoie le bébé auprès de sa mère, affirmant à sa femme qu'il veut un enfant autant qu'elle, mais qu'il doit être «notre enfant, pas celui de quelqu'un d'autre».

Écrivez une fonction qui prend en paramètre les noms de l'enfant et de sa *vraie* mère et qui modifie le graphe en conséquence.

▷ **Question 3:** Suite à ces différentes révélations, écrivez une fonction qui détecte les éventuelles anomalies généalogiques d'un graphe (avoir son père comme fils ou comme frère par exemple). Appliquez cette fonction au graphe de l'exemple.

▷ **Question 4:** Écrivez une fonction qui affiche les ascendants successifs d'un nœud (son arbre généalogique donc). Appliquez cette fonction à tous les nœuds du graphe.

▷ **Question 5:** Écrivez une fonction qui affiche les descendants d'un nœud pour une génération dont le numéro sera passé en second paramètre.

▷ **Question 6:** Écrivez une fonction qui affiche les descendants d'un nœud, génération par génération, jusqu'à une génération passée en second paramètre.

★ **Exercice 4: Aller plus loin.**

▷ **Question 1:** Écrivez une fonction qui ordonne le chaînage général du graphe selon l'ordre alphabétique des noms. Affichez le graphe ainsi obtenu.

▷ **Question 2:** Ray est devenu fou. Excédé par les vicissitudes des Ewing, il décide de tuer tous les membres de la famille (tuant également tout espoir des producteurs d'une suite possible sur TNT l'an prochain). Exercez sa vengeance en exécutant un par un la famille de Jock. Affichez le graphe résultat publié par la presse après chaque acte de vengeance.

▷ **Question 3:** Écrivez la méthode qui sauvegarde la structure de données sur fichier (on parle de sérialisation).

▷ **Question 4:** Écrivez la méthode qui crée le graphe à partir d'un fichier de sauvegarde produit par la méthode de la question précédente.

