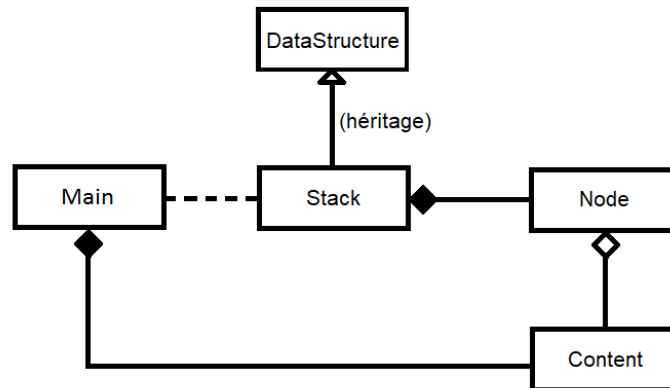


Exercice – Stack (pile)

Dans cet exercice, nous allons développer une structure dynamique, soit la **Stack (pile)**.

Le contenu qui sera empilé/dépilé sera un Livre, dont la classe est fournie avec l'énoncé. Le diagramme de composition de l'exercice n'est pas étranger à celui vu au cours :



Dans la solution de départ vous retrouverez le code complet de :

- 1) La classe Node
- 2) La classe Content (ici un livre)

ainsi que le code partiel de la classe DataStructure et Stack (codez le TODO).

Le code de la fonction main() est donné, mais il reste un petit bloc de code à effectuer en lien avec la composition indiquée dans le diagramme (voir TODO).

Codage de la classe Stack

Le but de l'exercice est de coder les méthodes de la classe Stack pour faire fonctionner la structure c'est-à-dire :

- 1) D'avoir un stockage des données conforme (les contenus sont réellement empilés sur un push et dépilés sur un pop)
- 2) De respecter la composition du diagramme ci-haut
- 3) D'avoir un programme sans fuite (leak) de mémoire

Conseil : Implémentez la méthode d'affichage fournie. En parcourant la pile depuis le premier nœud, vous aurez un visuel à la console qui vous aidera grandement.

Afin de coder la classe Stack, revoici les spécifications vues au cours (sous forme de « CheckList»). Le défi de ce travail est sans contredit le chaînage des pointeurs, il ne faut pas hésiter à dessiner sur du papier notre approche avant de commencer à lier des pointeurs de façon arbitraire, ce qui peut rapidement conduire à des anomalies de fonctionnement.

Spécifications de la Pile

Spécifications	<input checked="" type="checkbox"/>
Lorsque le pointeur vers le premier nœud est NULL, c'est l'indicateur que la pile est VIDE.	
La méthode push va créer un nœud, y insérer le contenu reçu et effectuer les chaînages pour empiler le nœud adéquatement.	
La méthode pop va retourner le contenu stocké dans le premier nœud en prenant soin de détruire ce nœud (composition !) et de refaire les chaînages adéquatement pour dépiler le nœud adéquatement	
Effectuer un pop sur une pile déjà vide retournera NULL	
Si la pile est détruite et qu'il reste des nœuds, la pile devra respecter sa composition forte. <u>Indice</u> : faire des appels de pop sans se soucier du contenu.	

Utilisation et tests de la structure

La solution de cet exercice ne sera pas donnée car vous êtes en train de construire du code essentiel au TP2. Votre Pile sera évaluée lors de celui-ci. Il est fortement conseillé de terminer cet exercice avant le TP2 car l'allocation du temps du TP2 prend pour acquis que votre structure de Pile fonctionne. En d'autres mots, vous aurez à coder la Pile, donc aussi bien le faire maintenant.

Il va de soi que la robustesse de votre structure est primordiale pour réussir le TP2. **Une série de test complète est donc à produire sur votre Pile et fera partie de son évaluation.** Afin de vous aider à débiter, un fichier de départ (dans le projet de tests) est donné avec ce travail pour vous guider sur les tests à écrire. Il est fortement recommandé d'écrire vos tests au fur et à mesure que vous progressez dans la pile, le but étant d'arriver au TP2 avec une structure fonctionnelle et sans faille, autrement, le problème ne pourra pas être résolu.

Bon Travail !