

TP n°2 / Structures de données / 1 CP / 2021

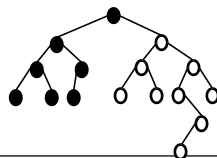
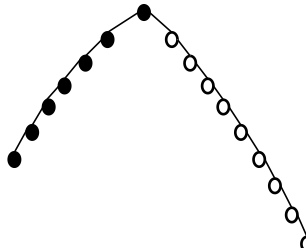
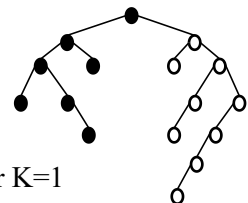
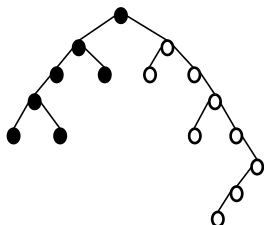
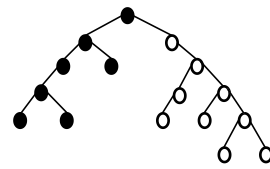
Dessiner des arbres avec des rotations

Considérer un arbre de recherche binaire construit comme suit:

- Les nœuds renferment des doublets (Valeur, Couleur)

Valeur est de type entier et Couleur de type booléen (deux couleurs possibles: noir et blanc)

- La racine est de couleur noire avec la valeur infini
- Les nœuds du sous arbre gauche sont noirs et ceux du sous arbre droit sont blancs.
- Les sous arbres gauche et droit sont ordonnés selon le champ Valeur.
- L'arbre global peut donc contenir des doubles avec des couleurs différentes.

<u>Création</u> Construire un tel arbre avec n doublets aléatoires.		
<u>Première transformation</u> Transformer - le sous arbre gauche en une liste ordonnée décroissante (arbre dégénéré) de telle sorte que tous les nœuds soient dépourvus de fils droits. - le sous arbre droit en une liste ordonnée croissante de telle sorte que tous les nœuds soient dépourvus de fils gauches.		
<u>Autres transformations</u> Transformer le sous arbre gauche (liste) obtenu par la première transformation comme suit La racine a n1 fils droit Le fils gauche de la racine à n2 fils formant une liste croissante Le fils gauche du fils gauche de la racine à n3 fils formant une liste croissante Etc. Transformer le sous arbre droit (liste) obtenu par la première transformation de manière analogue en remplaçant fils gauche par fils droit et liste croissante par liste décroissante		
Considérer les cas suivants (i) n1, n2,... sont générés par la suite $U_n=U_{n-1} + k$ (n>0) , $U_0=0$  Pour K=1 $U_n=U_{n-1} + 1$	(ii) n1,n2,... sont générés aléatoirement(valeurs identiques) 	(iii) n1,n2,... sont générés aléatoirement(valeurs différentes) 

Toutes les transformations se feront sur l'arbre en utilisant uniquement des rotations.

Partie Khawarizm

Réalisez les algorithmes sous l'environnement Khawarizm

Rajouter les modules nécessaires pour confirmer que la création et les transformations sont bien faites.

Partie C

Demander la traduction automatique vers C et tester le programme obtenu

Rajouter une fonction C qui transforme un arbre en un fichier textuel comme suit

graph aaa { a -- b; a -- c; b -- e; b -- d; } dans lequel a, b, c, d, e sont les nœuds de l'arbre et '--' désigne la relation de parenté.

La visualisation de l'arbre se fera alors dans le programme C avec la commande

`system("dot -Tpng fich.dot -o fich.png & mspaint fich.png").`

Afin de rendre possible cette opération, il faut utiliser la commande *dot* du logiciel GraphViz et un programme quelconque disponible sous windows pour visualiser le fichier image obtenu (Ex : Paint).

Le fichier textuel peut être amélioré (voir documentation) afin de bien visualiser l'arbre binaire (Position des fils gauche et droit, couleur, ...)

Téléchargement du logiciel GraphViz : <https://graphviz.org/download/>

Documentation: <https://graphviz.org/documentation/>

N.B : Il est inutile d'essayer de bien présenter les résultats avec Khawarizm. Il n'est pas fait pour ce but.

Notation

Khawarizm : 15 pts

C : 5 pts

Passage au langage C

Il est peu probable que lors de la traduction, le programme obtenu ne tourne pas. Dans ce cas, l'étudiant devra faire l'effort pour chercher et corriger l'erreur. Les principales erreurs sont dues principalement aux initialisations implicites de variables dans l'environnement Khawarizm. Toute erreur reconnue pertinente et corrigée par l'étudiant sera récompensée par un Bonus. Dans ce cas, nous envoyez le code Z pour lequel la traduction ne fonctionne pas et le code C corrigé en nous indiquant les changements. Si une erreur de traduction ne peut être corrigée, veuillez nous envoyer le code Z. Nous tenterons de la corriger.

Attention au plagiat !!!

Date de remise : 16 juillet 2021 avant Minuit

* Les TP seront remis selon un formulaire qui vous sera transmis par la suite

* Tout TP non remis selon l'échéancier fixé entraîne un retrait de 1 point par jour sur un total de 3 jours tolérés.