



Semaine 3, Bloc 2

Groupe S3-B2-G1H

Antoine LEMAIRE

Vincent LAVENU

Stéphane BEAUDET

Eric DANTAS

Structures de données

CODAGE HUFFMAN

Structure de données

CODAGE HUFFMAN

Le but : faire prendre conscience aux élèves que l'on peut réduire en mémoire le nombre de bits utiles pour coder et décoder une information SANS PERTE par rapport au message initial

Le moyen : faire coder en binaire des messages à l'aide de l'algorithme d'Huffman

Structure de données

CODAGE HUFFMAN

Objectifs

- * Découvrir qu'un message peut être compressé
- * Comprendre que coder avec un gain de place en mémoire ne suffit pas : il faut penser au décodage et sa lisibilité ; algorithme de Huffman
- * Évaluer le gain obtenu en fonction des arbres choisis : faire ressentir que la notion de codage statique est plus universelle mais moins efficace que celle de codage semi-adaptatif (au texte à coder ; codage semi-adaptatif / adaptatif ; coûteux en temps d'exécution car l'arbre se modifie)

Structure de données

CODAGE HUFFMAN

Capacités :

- * Être capable de trouver les occurrences d'itérations de lettres et symboles dans un texte
- * Créer un arbre binaire des fréquences/occurrences
- * Exploiter cet arbre binaire pour encoder et décoder un message

Structure de données

CODAGE HUFFMAN

Contenus	Capacités attendues	Commentaires
Arbres : structures hiérarchiques. Arbres binaires : nœuds, racines, feuilles, sous-arbres gauches, sous-arbres droits.	Identifier des situations nécessitant une structure de données arborescente. Évaluer quelques mesures des arbres binaires (taille, encadrement de la hauteur, etc.).	On fait le lien avec la rubrique « algorithmique ».
Algorithmes sur les arbres binaires et sur les arbres binaires de recherche.	Calculer la taille et la hauteur d'un arbre. Parcourir un arbre de différentes façons (ordres infixe, préfixe ou suffixe ; ordre en largeur d'abord). Rechercher une clé dans un arbre de recherche, insérer une clé.	Une structure de données récursive adaptée est utilisée. L'exemple des arbres permet d'illustrer la programmation par classe. La recherche dans un arbre de recherche équilibré est de coût logarithmique.

Structure de données

CODAGE HUFFMAN

Pré requis :

- * Codage en binaire de l'information : code ASCII (1ère)
- * Fréquence d'un élément dans une liste : écrire un algorithme de recherche d'une occurrence sur des valeurs de type quelconque (1ère)
- * Programmation Python
- * Structure de dictionnaire avec le couple clé/valeur (1ère)

Structure de données CODAGE HUFFMAN

Plan de la présentation

- Activité d'introduction &
Institutionnalisation
- Exercices (corrigés)
- Aller plus loin : vers un projet NSI?