Examen Codage Transmission Compression M2 Image 2009-2010

Le 14/12/2009, Tout document autorisé

Recopier tel quel des extraits de documents n'apporte jamais de points. Vous devez répondre aux questions qui vous sont posées. Vous devez argumenter (étayer) vos réponses. Une réponse correcte non étayée ne rapporte pas de points, et une réponse incorrecte bien étayée peut en rapporter.

Problème :

1. Une application gérant des réservations de ressources (planning) doit mémoriser des triplets : (numéro de ressource, début de la réservation, fin de la réservation).

   Proposez un codage compacte pour l'ensemble des ressources en imaginant ce que cela peut être dans un cas réel comme celui de l'université complète qui gère entre autre vos calendriers de cours.

2. Analysez votre codage en terme de temps d'accès.
3. Analysez votre codage en terme de temps nécessaire pour faire une modification dans le planning.
4. Quelle redondance reste-t-il dans votre codage ?
5. Comment faire mieux ?

Questions diverses :

6. Donnez la formule permettant de calculer la quantité d'information apporté par un tirage du loto (7 boules parmi 49).
7. Soit un texte écrit en français, comment calculeriez-vous la quantité d'information apportée par une lettre ?
8. Faut-il comprimer puis chiffrer (*crypter*) ou faire l'inverse ?
9. Quelle taille donner au dictionnaire de la méthode LZW ?
10. Pour un site web, dans quels cas utiliser GIF/PNG/JPG ?
11. Vous créez votre propre format de fichier compressé. Que faut-il prévoir pour que la décompression de la concaténation des fichiers comprimés fournisse la concaténation des fichiers non comprimés. $D(C(f_1) + C(f_2)) = f_1 + f_2$
12. Certaines langues humaines écrites (français, anglais...) ont un meilleur taux de compression que d'autres. Expliquez pourquoi (n'oubliez pas d'indiquer des exemples).
13. Dans quels cas la compression avec la méthode des ondelettes donne-t-elle de meilleurs résultats que la DCT de jpeg ?
14. Comment adapter le blocksorting à la compression d'image sans perte ?
15. Pourquoi comprime-t-on mieux en séparant les images des sous-titres dans les vidéos ?
16. Intel a économisé quelques watt en comprimant les transferts d'image entre la mémoire centrale et la mémoire de la carte graphique. À votre avis, quel algorithme ont-ils utilisés ?

1 Une correction

1. Une application gérant des réservations de ressources (planning) doit mémoriser des triplets : (numéro de ressource, début de la réservation, fin de la réservation).

   Proposez un codage compacte pour l'ensemble des ressources en imaginant ce que cela peut être dans un cas réel comme celui de l'université complète qui gère entre autre vos calendriers de cours.

   En vrac :

   - *Indiquer les dates en considérant une durée minimum. C'est 1/4 d'heure à l'université.*
   - *Stocker la durée au lieu de la date de fin.*
   - *Indiquer une date de début en relatif par rapport à la dernière date si les informations sont triées par date.*
   - *Fusionner les répétitions d'information. Souvent il y a le même créneau horaire d'une semaine à l'autre.*
   - *Regrouper les informations identiques. Par exemple une table par ressource, qui se subdivise en une table par durée, cette table contenant des dates de début.*

   Si le dernier point est utilisé il y a peu d'intérêt à faire un équilibrage de probabilités à part sur les dates de début.

2. Analysez votre codage en terme de temps d'accès.

   L'accès à une date nécessite de parcourir toutes les ressources, toutes les durées. Pour trouver la date le temps est en log *si l'on a pas codé les répétitions et que l'on a pas codé en relatif, sinon il risque d'être linéaire. Pour avoir un temps d'accès plus rapide il est nécessaire de mettre en place des structures de données adaptées mais qui prendront de la place.*

3. Analysez votre codage en terme de temps nécessaire pour faire une modification dans le planning.

   Dans le cas le plus courant, on ajoute une date de début dans la table. C'est donc linéaire en fonction du nombre de allocation de la ressource pour la durée indiquée.

4. Quelle redondance reste-t-il dans votre codage ?

   Généralement les ressources sont allouées ensembles : salle + étudiants + enseignants. Avec le codage proposé, on refait 3 fois la réservation.

5. Comment faire mieux ?

   Pour la place mémoire : faire des ressources virtuelles composées de plusieurs ressources. Cela aura pour effet de bord d'accélérer les recherches..

   Pour améliorer le temps d'accès par date : Un tableau indexé par le jour de l'année pointant vers une table contenant pour chaque heure de début possible une table de paires (liste de ressources, durée).

Questions diverses :

6. Donnez la formule permettant de calculer la quantité d'information apporté par un tirage du loto (7 boules parmi 49).

   On calcule la probabilité d'un tirage donné sachant que tous les tirages sont équiprobables. Si l'ordre compte : $-log_2 \frac{1}{49*48*47*46*45*44*43}$
   S'il ne compte pas : $-log_2 \frac{7!}{49*48*47*46*45*44*43}$

7. Soit un texte écrit en français, comment calculeriez-vous la quantité d'information apportée par une lettre ?

   Par exemple $-log_2\, l_n$ *ou* l_n *est la probabilité d'apparition de la lettre sachant qu'elle est précédée de la chaine de caractère n. Même en prenant n vide les probabilités changent suivant la langue. Si l'on prend n suffisamment grand pour contenir un mot alors la probabilité tient compte du dictionnaire de la langue. Si l'on prend plus grand, on tiendra compte de sa grammaire et de sa syntaxe.*

8. Faut-il comprimer puis chiffrer (*crypter*) ou faire l'inverse ?

   On ne peut pas comprimer un texte chiffré car il ne contient plus aucune redondance. Malheureusement si l'on chiffre un texte comprimé, quelqu'un qui veut casser le chiffrement sait d'avance quel est le début du fichier.... Le chiffrement sera plus rapide si le fichier est plus petit.

9. Quelle taille donner au dictionnaire de la méthode LZW ?

   Grand s'il y a une cohérence globale, petit si elle est seulement locale. Le mieux étant qu'elle soit adaptative.

10. Pour un site web, dans quels cas utiliser GIF/PNG/JPG ?

    JPG pour des photos réelles, PNG pour la transparence ou pour travailler sans perte et si l'image n'est pas naturelle, GIF pour la portabilité.

11. Vous créez votre propre format de fichier compressé. Que faut-il prévoir pour que la décompression de la concaténation des fichiers comprimés fournisse la concaténation des fichiers non comprimés. $D(C(f_1) + C(f_2)) = f_1 + f_2$

    Il faut stocker un événement de fin de fichier. Le décompresseur devra s'aligner en début d'octet suivant pour ne pas lire des bits sans significations.

12. Certaines langues humaines écrites (français, anglais...) ont un meilleur taux de compression que d'autres. Expliquez pourquoi (n'oubliez pas d'indiquer des exemples).

    Je n'en donne que deux : 'les pommmes' : 'the apples'. le pluriel est indiqué une seul fois en anglais et 2 fois en français. Et 'nous mangeons' : 'we eat' dans lequel le sujet est indiqué une seule fois en anglais et deux fois en français. Le français contient plus de redondance que l'anglais et devrait donc être plus comprimé.

    Évidemment les langues dont les mots comportent très peu de lettres (ou un seul symbole) ne sont pas autant comprimable.

13. Dans quels cas la compression avec la méthode des ondelettes donne-t-elle de meilleurs résultats que la DCT de jpeg ?

    Pour les images avec une très grande résolution car on peut coder de grandes zones de pixels en une seule fois.

14. Comment adapter le blocksorting à la compression d'image sans perte ?

    Par exemple en faisant un parcours de Péano des pixels. Il y a plus de dépendance entre 4 pixels d'un carré que 4 pixels d'une ligne. Pour la couleur, il faut considérer le caractère sur 3 octets.

15. Pourquoi comprime-t-on mieux en séparant les images des sous-titres dans les vidéos ?

    Parce que pour les images le plus efficace c'est la DCT ou les ondelettes et pour le sous titre c'est de faire un OCR.

Les hautes fréquences du texte perturbent la compression d'image en faisant baisser le taux de compression ou alors le sous titre est très dégradé.

L'image dans laquelle le sous-titre est "effacé" en le remplaçant par des pixels moyens sera comprimée beaucoup plus efficacement et sans effets de bord dus au texte.

16. Intel a économisé quelques watt en comprimant les transferts d'image entre la mémoire centrale et la mémoire de la carte graphique. À votre avis, quel algorithme ont-ils utilisés ?

    Le coût énergétique du transfert d'un mot est très faible. Il est clair qu'utiliser une centaine d'instruction du processeur pour faire le transfert ce n'est pas possible. Il ont donc utilisé l'algorithme qui utilise le moins de temps CPU : le RLE.

    D'autre part le REL va parcourir une seule fois l'image du début à la fin et générer directement le résultat comprimé.