

Faculté des Sciences Exactes et d'Informatique  
Département d'Informatique

**Sujet 1 : du Concours d'Accès au Doctorat LMD Informatique 2018/2019**

**Épreuve Commune : Système d'Information et Bases de Données (durée : 1H30)**

**Exercice 1 (6pts)**

Soit une structure d'un fichier à organisation séquentielle indexée (permet l'accès séquentiel et direct en utilisant la clé), contenant des enregistrements occupant 160 octets chacun dont 10 octets sont réservés pour la clé d'accès.

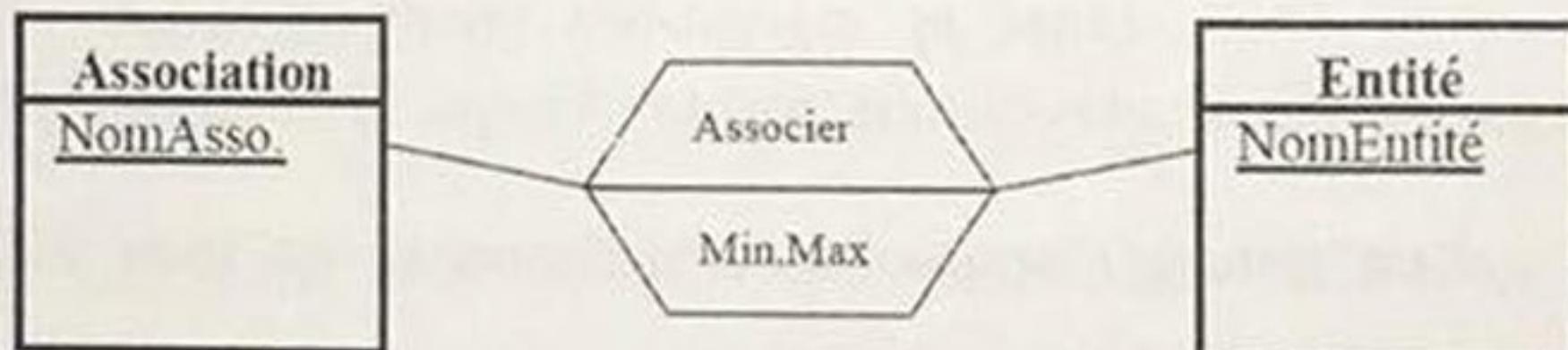
- Quelle taille minimale de bloc de données (enregistrements) permet d'éviter d'avoir de l'espace disque inutilisé dans le fichier si on suppose que les pages (blocs) sont des multiples de 1024. En déduire le nombre d'enregistrements par bloc.
- Si l'on prend la taille retrouvée ci-dessus aussi bien pour les blocs de données que pour les blocs d'index et supposons que la référence d'un bloc d'index vers un bloc de données est de 6 octets, comme le montre la table d'index suivante :

Clé <sub>1</sub>	Clé <sub>2</sub>	....	Clé <sub>n</sub>
Ref <sub>1</sub>	Ref <sub>2</sub>	....	Ref <sub>n</sub>

- Combien de données par bloc faudrait-il mettre pour atteindre un taux de remplissage d'environ 85%.
- Quelle place disque occupera le fichier (enregistrement et index) lorsqu'il contiendra cent mille enregistrements si on remplit les blocs de données à 85 %. Pour trouver cette taille il faut d'abord chercher :  
Le nombre de blocs de données, le nombre de blocs d'index nécessaire pour référencer les blocs de données et l'arborescence d'index (le nombre de blocs d'index dans chaque niveau de l'arbre).

**Exercice 2 (6pts)**

Soit le modèle de données,



- Compléter le modèle en cardinalités, puis expliquer l'association « Associer ».
- Compléter le modèle en tenant compte des règles de construction d'un modèle de données
  - Une propriété permet de décrire une entité et au moins une propriété appartient à la description de l'entité.
  - Une des propriétés d'une entité joue le rôle de clé ou identifiant.
  - Une association peut ne pas avoir de propriétés, comme elle peut avoir autant de propriétés.
  - Une propriété appartient à un type de données qui peut avoir les valeurs (élémentaire, paramètres, calculé, etc.).
- Donner la base de données correspondante au modèle développé en (2) (l'ensemble des tables).

### Exercice 3 (8pts)

Soit le schéma relationnel suivant représentant une bases de données de gestion de commandes d'une entreprise, décrit comme suit: une commande d'un client peut contenir plusieurs produits. Chaque produit est fourni par un seul fournisseur. Un produit peut être stocké dans plusieurs entrepôts avec des quantités différentes. Pour chaque entrepôt, on garde la trace des produits qui y sont stockés effectivement.

Produit (# id\_produit, description\_produit, nom\_produit, prix\_produit, id\_fournisseur)

Commande (# id\_commande, description\_commande, date\_commande, date\_livraison\_commande, id\_client)

Client (# id\_client, nom\_cli, prenom\_cli, num\_mobile\_cli, num\_tel\_cli, adresse\_cli, id\_ville)

Fournisseur (# id\_fournisseur, raison\_sociale\_fournisseur, adresse\_fournisseur, num\_tel\_fournisseur, delai\_moyen\_livraison\_fournisseur, id\_ville)

Ville (# id\_ville, nom\_ville, id\_pays)

Pays (# id\_pays, nom\_pays, region\_pays)

Entrepot (# id\_entrepot, nom\_entrepot, superficie\_entrepot, adresse\_entrepot)

Stockage (# id\_entrepot, # id\_produit, quantité)

Ligne\_commande (# id\_commande, # id\_produit, prix, quantité)

Ecrire en SQL les requêtes permettant de :

1. Connaitre les noms des villes des fournisseurs qui ne comportent aucun client.
2. Connaitre, à la demande, les noms, prénoms et numéros de téléphones des clients ayant passé au moins une commande durant les 30 derniers jours.
3. Classer les clients (leur noms et prénoms) par ordre croissant du montant de leurs commandes, toutes commandes confondues tout en affichant le montant.
4. Connaitre, à la demande, les noms des entrepôts qui contiennent tous les produits de l'entreprise.
5. Réduire de 5% les prix affichés des produits ayant été commandés au moins une fois

Sujet 2 : du Concours d'Accès au Doctorat LMD Informatique 2018/2019

Spécialité : Intelligence Artificielle

Épreuve : Technique pour Intelligence Artificielle (durée : 02H)

**Exercice (6pts)**

Soit  $G = (V, E)$  un graphe connecté non orienté,  $V$  est l'ensemble de sommets et  $E$  un ensemble d'arêtes, tel que  $|V| = n$ ,  $|E| = m$ .

Soit l'algorithme :

**Debut**

Trier les arêtes du graphe selon le poids ascendant ;

$R \leftarrow \{\}$  ;  $S \leftarrow \{\}$ ;

Pour tout sommet  $s_i \in V$  Faire  $S \leftarrow S \cup \{s_i\}$ ;

Tant que ( $i \leq m$ ) Faire

$a_i \leftarrow (u, v)$  /\* selon l'ordre du tri effectué \*/

Si  $F(u, S) \neq F(v, S)$  Alors //  $F(s, S)$  : retourne l'indice de l'ensemble contenant le sommet  $s$  ;

$R \leftarrow R \cup \{a_i\}$ ;

$Fusion(S(F(u, S)), S(F(v, S)))$  // Fusionner les deux ensembles dans  $S$  contenant  $u$  et  $v$

FinSi

$i \leftarrow i + 1$ ;

FinTantque

Retourne( $R$ ) ;

**Fin**

1. Soit  $I$  une instance du graphe  $G$  définie par l'ensemble  $I = \{(u, v, w)\}$  où  $(u, v)$  est une arête et  $w$  le poids associé,  $I = \{((1,2), 4), ((1,3), 8), ((1,6), 8), ((2,3), 11), ((2,6), 8), ((3,4), 7), ((3,5), 1), ((4,5), 6), ((4,6), 2), ((4,8), 7), ((5,7), 2), ((6,7), 4), ((6,8), 7), ((7,9), 10), ((8,9), 9)\}$ . Dérouler l'algorithme sur cette instance, que produit l'algorithme.
2. Ecrire les deux algorithmes traduisant la fonction  $F()$  et la procédure  $Fusion()$ , puis évaluer leurs complexités.
3. En déduire que la complexité de l'algorithme ci-dessus est  $O(m \log m)$ .

**Problème (14pts)**

Le problème de remplissage des boîtes (Bin Packing) est un problème qui consiste à ranger un ensemble d'objets  $E = \{e_1, e_2, \dots, e_n\}$  dans un minimum de boîtes de taille égale à 1. Le nombre de boîtes est illimité. Chaque objet  $i$  possède une taille  $s_i$  avec  $0 < s_i < 1$ . Chaque boîte peut contenir tout sous-ensemble d'objets dont la somme de tailles n'excède pas 1.

Il existe plusieurs stratégies (heuristiques) d'arrangement des objets dans les boîtes :

- Stratégie FirstFit : on prend chaque objet l'un après l'autre et on le place dans la première boîte qui peut l'accueillir.

- Stratégie NextFit : on prend chaque objet l'un après l'autre et on le place dans la dernière boîte si on peut, sinon l'objet est placé dans une nouvelle boîte.
  - Stratégie FirstFitDecreasing : trier les objets dans l'ordre décroissant de taille et sur cet ordre exécuter l'algorithme FirstFit().
1. Dérouler les trois stratégies sur l'ensemble  $E = \{0.4, 0.8, 0.5, 0.7, 0.6, 0.1, 0.4, 0.2, 0.2, 0.1\}$
  2. Ecrire les algorithmes qui implémentent les trois heuristiques.
  3. La complexité algorithmique de problème de Bin Packing est NP-complet, c'est-à-dire qu'il n'existe pas d'algorithme polynomial capable de résoudre d'une manière exacte une instance du problème. A cet effet nous cherchons à implémenter un algorithme génétique capable de trouver des solutions proches de l'optimum envisagé.
    - a. Proposer une structure de codage des chromosomes puis de population. Expliquer à l'aide d'un exemple.
    - b. Proposer une fonction objectif pour le problème de Bin Packing puis écrire la fonction d'évaluation d'un individu.
    - c. Développer une procédure d'initialisation aléatoire des solutions qui vérifient les contraintes imposées.
    - d. Proposer un algorithme de sélection basé sur le principe de tournoi.

Concours d'accès au Doctorat 3<sup>ième</sup> Cycle Informatique 2018 – 2019

Le 24/10/2018

Matière 1 : Algorithmique avancée et complexité,  
Coefficient 1, durée 1 h 30  
(Spécialités : IA, MFA, SIGL)

## Exercice 1 : (12 points)

On considère un tableau A à deux dimensions de valeurs entières positives telles que les valeurs d'une même ligne et celles d'une même colonne sont triés par ordre croissant, non forcément strictement.

Exemple:

$$A = \begin{array}{|c|c|c|c|c|} \hline & 2 & 14 & 25 & 30 & 69 \\ \hline .3 & 15 & 28 & 30 & 81 & \\ \hline 7 & 15 & 32 & 43 & 100 & \\ \hline 20 & 28 & 36 & 58 & 101 & \\ \hline \end{array} \leftarrow$$

Il est à noter aussi que tout sous-tableau les valeurs d'une même ligne et celles d'une même colonne sont triées par ordre croissant. Le but de l'exercice est de rechercher efficacement une valeur  $val$  dans A. Pour simplifier, on supposera que le tableau A comporte  $n = 2^k$  lignes et colonnes.

1. Une première solution consiste en un balayage séquentiel de A par ligne (ou par colonne) jusqu'à trouver  $val$ .
  - Donner l'algorithme.
  - Quelle est sa complexité au pire (en termes de nombre de comparaisons), si m est le nombre de lignes et n le nombre de colonnes de A ? Peut-on l'améliorer en utilisant le fait que les lignes et les colonnes sont triées ?
2. On considère la recherche de val dans une matrice A quelconque (m lignes, n colonnes) en distinguant la valeur  $z = A[1, n]$ . Que convient-il de faire selon que z est égal, inférieur, ou supérieur à val ? Ecrire la procédure récursive de la recherche de la valeur val correspondante. Quelle est sa complexité au pire cas ?
3. On considère ici que la matrice A est carrée avec  $m = n = 2^k$ .

On distingue les deux valeurs :  $x = A\left[\frac{n}{2}, \frac{n}{2}\right]$  et  $y = A\left[\frac{n}{2} + 1, \frac{n}{2} + 1\right]$

- a. Montrer que si la valeur recherchée  $val$  est telle que  $val > x$ , on peut éliminer une partie (à préciser) de A pour poursuivre la recherche. Préciser ce qu'il est possible de faire quand  $val < y$ .
- b. Appliquer la recherche à une matrice où  $k=0$  puis  $k=1$ .

En déduire un algorithme « diviser pour régner » pour résoudre ce problème, et préciser le coût dans le pire des cas, en nombre de comparaisons. Avec cette méthode a-t-on amélioré la complexité ?

## Exercice 2 : (8 points)

(Les questions suivantes sont indépendantes)

1. Qu'est-ce qu'un problème NP ? Qu'est-ce qu'un problème NP-Complet ?  
(Donnez des définitions précises).

Appliquer la définition précédente pour affirmer ou infirmer la NP-complétude des problèmes suivants: le problème 2-SAT, le problème 3-SAT.

2. Résoudre par récurrence l'équation suivante :



## Résolution de Problèmes (1)

Soit le problème d'états suivant, consistant à trouver la séquence d'actions nécessaire à partir d'un état initial pour atteindre un état but. On dispose d'une caisse, composée de  $n \times n$  cases, dans laquelle on veut placer  $n \times n$  boites de différentes couleurs (pas nécessairement de couleurs différentes). Un robot automatique peut permute simultanément deux boites adjacentes dans leurs cases respectives (en ligne ou en colonne). Les permutations doivent cependant respecter la condition suivante: une boite de couleur X ne peut pas être placée dans une autre case dont les cases adjacentes (en horizontal et en vertical) sont de même couleur qu'elle. L'état initial peut contenir des cases adjacentes en ligne et/ou en colonne de même couleur.

1. Modélisez ce problème par un système d'état en donnant:

- Une description d'état
- L'état initial
- La description de l'état but (sous forme algorithmique)
- Les règles de changement d'état.

### Cas particulier:

2. On suppose maintenant que nous avons 3 boites noires (N1, N2, N3), 3 boites blanches (B1, B2, B3) et 3 boites vertes (V1, V2, V3), disposées initialement selon l'état ci-dessous:

B3	B2	V3
N3	B1	V2
N1	N2	V1

Soit l'état final représenté par:

B3	N3	V3
V2	B2	N2
N1	V1	B1

Décrivez l'état initial et l'état final en utilisant les prédictats (logique du 1er ordre).

Indications: Vous pouvez utiliser les prédictats suivants:

- Adjacent (X, Y): la boite X est adjacente à la boite Y.
- Couleur(X, C): C est la couleur de la boite X.

3. Donner l'espace de recherche en chaînage avant avec la stratégie en largeur d'abord, en précisant les règles utilisées ainsi que l'ordre de développement des nœuds.

Indication: Vous pouvez utiliser la règle (en respectant la condition de départ ci-dessus sur les permutations):

A constater qu'il est inutile de permute des boites de même couleur, ainsi que les permutations ramenant à un état précédent. De ce fait, les actions doivent éviter une étape inutile de ce type.



Concours d'accès au Doctorat 3ème Cycle Informatique 2018 – 2019

Le 24/10/2018

Matière 2 : Résolution de Problèmes, Représentation des Connaissances et Raisonnement,  
Coefficient 1, durée 2 Heures.  
(Spécialité : IA)

Partie 2 : Représentation des Connaissances et Raisonnement

Exercice 1 : (10 points)

Etant donné le langage de la logique de description avec la syntaxe :

$$C \rightarrow A \mid \top \mid \perp \mid \neg C \mid C \sqcap D \mid \forall R.C \mid \exists R.C \mid C \sqcup D$$

- A- Supposons donnés les concepts atomiques et les rôles nécessaires, décrivez les concepts définis ci-dessous :
- 1- Les animaux marins sont des animaux.
  - 2- Les animaux marins sont généralement ovipares, de sang-froid et respirent sous l'eau.
  - 3- Les poissons sont des animaux marins.
  - 4- Les mammifères sont généralement non ovipares.
  - 5- Les mammifères sont de sang chaud, de respiration aérienne et allaitent leurs petits.
  - 6- Les mammifères marins sont des animaux marins.
  - 7- L'ornithorynque est un mammifère terrestre ovipare.
  - 8- Les mammifères marins sont généralement de grande taille. Les marsouins du pacifique sont des mammifères marins de petite taille. Les baleines sont des mammifères marins.
- B- Exprimez dans la A-box que p est un poisson, m un marsouin, o un ornithorynque et b une baleine. Que pouvez-vous déduire ?
- C- A partir des T-Box et A-box décrites ci-dessus, que pouvez-vous déduire du système dans le cas p, m, o et b. Les déductions sont-elles cohérentes ?
- D- Proposez une écriture, dans une autre logique de description, du concept marsouins.
- E- Proposez une autre représentation des concepts définis en A, en Logique de Défauts Normaux. Quelles sont les extensions dans le cas p, m, o, et b



جامعة هواري بومدين للعلوم والتكنولوجيا

Université des Sciences et de la Technologie Houari Boumediene

Faculté d'Electronique et d'Informatique

Département d'Informatique

**Concours d'accès au Doctorat 3ème Cycle Informatique 2018 – 2019**

**Exercice 2 : (10 points)**

A- Quatre économistes discutent à propos des conséquences de la crise économique entre la Chine et les USA.

Le premier économiste atteste cette crise va provoquer un ralentissement économique en Chine à 25%.

Le deuxième expert affirme que la crise va induire :

- une augmentation des prix des produits de consommation à 20%,
- un risque pour la croissance américaine à 40%,
- des dégâts aux secteurs manufacturiers à 40%.

Le troisième économiste atteste que cette crise va provoquer :

- un ralentissement économique en Chine et des dégâts aux secteurs manufacturiers à 45%,
- une augmentation des prix des produits de consommation à 25%.

Le quatrième économiste dit qu'il ne sait pas.

- 1- Représentez ces connaissances en utilisant la théorie de Dempster et Shafer.
- 2- Dans le contexte de la deuxième expertise, calculez les degrés de croyance et de plausibilité.
- 3- Quelles sont les particularités des quatre expertises ? Explicitez.
- 4- Comment prendre en compte les quatre expertises ? Explicitez chaque étape.
- 5- Quelles sont les conséquences les plus soutenues ?

B- De plus, sachant que cette crise économique est due essentiellement à la décision de surtaxer des produits chinois importés et ceci afin de réduire l'excédent commercial chinois envers les USA,

- 6- Modélisez l'ensemble de ces connaissances (partie A et partie B de l'exercice 2) en utilisant un diagramme d'influence.