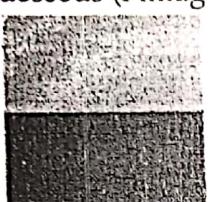


Connaissance du cours (3 x 3pts = 9pts)

En considérant le noyau de convolution suivant:

Présenter la réponse obtenue en appliquant le noyau à l'image ci-dessous (l'image est partagée en deux zones homogènes) :

1	0	-1
1	0	-1
1	0	-1

Exercice 1 (2x3 = 6 pts)

- Quelles sont les différences fondamentales entre l'égalisation et l'étirement d'histogramme ? Pour répondre à cette question, vous commencerez par définir chacune de ces deux notions, leur intérêt, puis leurs différences fondamentales en vous appuyant sur des exemples concrets.
- Expliquez comment, à partir d'une image en niveaux de gris, on obtient une image binaire dans laquelle les contours (fins) sont blancs et le reste de l'image est noir (avec la méthode de votre choix, que vous préciserez). Vous veillerez à bien justifier toutes les étapes importantes.

Exercice 2 (0,5+0,5+1+0,5+0,5+1+1+1+1+1 = 9 pts)

1) Définir la dynamique D d'une image.

2) Expliquer ce que représente l'histogramme d'une image en niveau de gris

3) Expliquer la notion de seuillage d'une image en niveau de gris. Quelles critères sur les niveaux de gris des objets et ceux du reste de l'image (le 'fond') permettent un seuillage automatique ?

4) Soit le noyau de convolution suivant:

1	2	1
2	4	2
1	2	1

- Calculer le coefficient de normalisation de ce noyau.
- Quel est l'effet de ce filtre ?
- Dans quelle cas a-t-on intérêt à utiliser un tel filtre ?
- Quels sont ses inconvénients ?

5) Soient

- l'image 1D suivante (L'image 1D : 56, 64, 79, 98, 115, 126, 132, 133) et
- le filtre de Roberts 1D (Le filtre de Roberts : 1, -1) pour détecter les contours (à droite)

(a) Calculez le résultat de la convolution en négligeant les effets de bords (7 valeurs seulement)

(b) Combien de contours trouvez-vous dans cette image et où sont-ils ? Expliquez

(c) En pratique, lorsqu'on utilise un détecteur de contour, on détecte beaucoup d'autres lignes qui ne

sont pas des "contours d'objets". Pourquoi ? Quels autres types de lignes détecte-t-on (donnez 3 exemples)

**Exercice 1: (9 points)**

1. Rappeler l'algorithme d'apprentissage des réseaux RBF vu en cours.
2. Montrer sur une figure que la fonction XOR n'est pas linéairement séparable.
3. On veut apprendre la fonction XOR avec un réseau RBF constitué de 2 entrées, deux neurones RBF dans la couche cachée, un neurone dans la couche cachée représentant le biais et un neurone à seuil en sortie. Représenter schématiquement le réseau.
4. Les coordonnées des neurones RBF sont  $\mu_1(0, 0)$  et  $\mu_2(1, 1)$  et leur fonction d'activation de la forme  $\phi_i(x) = \exp(-\|x - \mu_i\|^2)$  ( $i = 1, 2$ ). Calculer et présenter dans un tableau les activations des deux neurones RBF pour les 4 entrées de la fonction XOR.
5. Représenter dans un nouveau repère les positions des 4 entrées de la fonction XOR. Les nouvelles coordonnées de chaque point sont donnée par  $(\phi_1(x), \phi_2(x))$ . Montrer que dans ce nouveau repère la fonction XOR est linéairement séparable.
6. A partir de l'exemple de la fonction XOR justifier pourquoi la seconde partie de l'apprentissage dans un réseau RBF peut se faire en utilisant l'algorithme du perceptron simple.
7. Au lieu d'utiliser l'algorithme du perceptron simple il est possible d'utiliser la matrice pseudo-inverse pour obtenir les poids de connexion entre la couche cachée et la couche de sortie. Quel est le principal inconvénient de cette dernière méthode ?

**Exercice 2: (6 points)**

1. Rappeler le principe général de la descente du gradient dans le perceptron multi-couches.
2. Comment la valeur du taux d'apprentissage impacte la convergence de l'algorithme d'apprentissage du perceptron multi-couches.
3. Décrire comment fonctionne l'algorithme Full gradient descent ? Stochastic gradient descent ? Mini-batch gradient descent ?
4. Décrire comment l'algorithme de descente du gradient est optimisée avec l'utilisation du momentum.
5. Décrire comment l'algorithme de descente du gradient est optimisée avec l'utilisation de Nesterov accelerated gradient.
6. Décrire comment l'algorithme de descente du gradient est optimisée avec l'utilisation de RMSprop.

**Exercice 3: (5 points)**

Un biais cognitif et une interprétation erronée de la réalité par le cerveau. Quand le cerveau humain observe quelque chose qui ne correspond pas à une certaine

cohérence, alors le cerveau règle le problème automatiquement : il interprète. Sauf que cette interprétation ne correspond pas forcément la réalité. Par exemple, le biais de "bandwagon" ou "du monton de Parnasse" peut conduire le programmeur à suivre des modélisations qui sont populaires sans s'assurer de leur exactitude. Les biais d'anticipation et de confirmation peuvent conduire le programmeur à favoriser sa vision du monde même si des données disponibles peuvent remettre en question cette vision. Le biais de "corrélations illusoires" peut également conduire une personne à déceler des corrélations entre deux événements indépendants. L'une problématique actuelle de l'intelligence artificielle c'est l'impact de ces biais cognitifs dans la production de modèles d'apprentissage car l'on observe que les modèles construits ont une tendance à reproduire les biais cognitifs des humains (ceux qui travaillent dans la chaîne de création des modèles -de la collecte des données à la mise en production dans des outils-) et donc à guider parfois vers des décisions biaisées par rapport à un comportement rationnel ou à la réalité. Comment pensez-vous que ce problème peut être résolu en apprentissage artificiel ?

INF4248 : Apprentissage Artificiel II  
Contrôle continu  
Durée : 2 heures

**Exercice 1: (5 points)**

1. Lorsque les données sont linéairement séparables, la formulation utilisée pour construire un SVM est :

$$\text{Min} \left( \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i(w \cdot x_i + b) - 1) \right), \quad \alpha_i \geq 0 \quad \forall i \quad (1)$$

En posant  $L(w, b) = \frac{1}{2} \|w\|^2 - \sum_i \alpha_i (y_i(w \cdot x_i + b) - 1)$  et en utilisant les conditions de Karush-Kuhn-Tucker qui sont :  $\frac{\partial L(w,b)}{\partial w} = 0$ ,  $\frac{\partial L(w,b)}{\partial b} = 0$ ,  $\alpha_i \geq 0$  et  $\alpha_i(y_i(w \cdot x_i + b) - 1) = 0$ , retrouvez l'expression de  $L(\alpha)$  dans la formulation duale suivante :

$$\begin{cases} \text{Max } L(\alpha) \\ \text{sujet à } \sum_i \alpha_i y_i = 0 \text{ et } \forall i \alpha_i \geq 0 \end{cases} \quad (2)$$

2. En vous référant à l'expression de  $L(\alpha)$  pourquoi dit-on que l'apprentissage par les SVM n'est pas influencée par la dimension des patterns ?
3. En utilisant les conditions de Karush-Kuhn-Tucker, montrez que  $\alpha_i$  est nul pour tout vecteur  $x_i$  non support.
4. Comment déduire  $b$  après avoir obtenu le vecteur  $\alpha$  ?
5. Expliquez l'astuce des fonctions à noyau (Kernel Trick).

**Exercice 2: (5 points)**

Pedro Domingos a publié l'article : *Domingos Pedro, A few useful things to know about machine learning, Communications of the ACM, Vol. 55, No. 10, 78-87, 2012* dans lequel il présente l'apprentissage artificiel.

1. Dans cette article, l'auteur postule que "Learning = Representation + Evaluation + optimization". Expliquer.
2. L'auteur estime que la clé d'un bon apprentissage réside dans le choix des caractéristiques des patterns à apprendre. Votre avis ?
3. L'auteur pense qu'un algorithme de taux d'apprentissage théorique moyen et qui utilise beaucoup de données d'apprentissage est meilleur qu'un algorithme ayant de très bonne propriété théorique et qui utilise peu de données. Votre avis ?
4. L'auteur indique que "Après le sur-apprentissage, le plus grand problème de l'apprentissage artificiel est la malédiction de la dimensionnalité". Votre avis ?

**Exercice 3: (5 points)**

1. Rappeler le principe de l'algorithme AdaBoost.
2. Les algorithmes d'apprentissage par ensemble sont d'autant plus performants que les classificateurs de base sont instables. Expliquer et donner en justifiant deux classes de classificateurs de base instables.
3. Expliquer pourquoi les données n'ont pas besoin d'être normalisées lorsque l'on utilise les algorithmes de type Boosting ?
4. Indiquer en justifiant parmi les 3 classes d'algorithmes d'apprentissage par ensemble celle(s) qui est(sont) parallélisable sans trop de difficulté ?

**Exercice 4: (5 points)**

Une entreprise spécialisée dans les télécommunications et qui a entendu parlé de fouille de données et d'apprentissage artificiel vous recrute en tant qu'ingénieur chargé de mettre en œuvre des solutions relevant de ses disciplines pour améliorer son chiffre d'affaire, son nombre de clients et la productivité du personnel. Sachant que vous avez 1 an pour justifier de la nécessité de disposer d'un département permanent dédié à la fouille de données et à l'apprentissage artificiel dans l'entreprise, expliquez très précisément la démarche que vous allez entreprendre.



### Connaissance du cours (3 x 3 pts = 9 pts)

1. Définir : Base de données parallèle, Base de données distribuées
2. Donner avec explication et exemples 3 avantages des bases de données distribuées
3. Qu'est-ce que la duplication synchrone, Qu'est-ce que la duplication asynchrone, Faire une comparaison entre la duplication synchrone et la duplication asynchrone

### Exercice (11 pts)

schéma global de la base La base de données hospitalière a le schéma suivant :

service (Snum, nom, hôpital, bât, directeur)

le directeur d'un service est un docteur désigné par son numéro

salle (Snum, SANum, surveillant, nblits)

le numéro de salle est local à un service, i.e., il peut y avoir des salles avec le même numéro dans des services différents d'un même hôpital.

nblits est le nombre total de lits d'une salle, in surveillant de salle est un infirmier désigné par son numéro employé (Enum, nom, adr, té)

docteur (Dnum, spéc) -- sp̄c est la sp̄cialité du médecin

infirmier (Inum, Snum, rotation, salaire)

Un employé est soit infirmier soit docteur (Inum et Dnum sont référence à Enum).

patient (Pnum, Snum, SANum, lit, nom, adr, tél, mutuelle, pc)

L attribut pc est la prise en charge par la mutuelle

acte (Lnum, Pnum, date, description, coef)

coef est le coefficient de l'acte médical

### Question 1

exprimer en SQL la question suivante : "Donner le nom des cardiologues qui ont traité un ou plusieurs patients hospitalisés dans un service de gérontologie."

#### épartition des données

La base est répartie sur trois sites informatiques, "Douala", "Ebolowa" et "Régional", correspondant aux valeurs "aquintinie", "Hôpital Régional" et "autre" de l'attribut hôpital de Service.

### Question 2

Proposer (et justifier) une bonne décomposition de la base hospitalière sur ces trois sites. On pourra utiliser la agmentation horizontale et/ou verticale ainsi que la réPLICATION des données, en se basant sur les hypothèses suivantes H1 à H5)

H1: Les sites Douala et Ebolowa ne gèrent que les hôpitaux correspondants.

H2 : Les infirmiers sont employés dans un service donné.

H3 : Les docteurs travaillent le plus souvent sur plusieurs hôpitaux (ou cliniques).

H4 : La gestion des lits d'hôpitaux est locale à chaque hôpital.

H5 : On désire regrouper la gestion des frais d'hospitalisation au centre régional.

Sur chaque fragment, on donnera sa définition en algèbre relationnelle à partir du schéma global.

**REPUBLIQUE DU CAMEROUN**  
Paix - Travail - Patrie



**UNIVERSITÉ DE YAOUNDÉ I**  
**Faculté des Sciences**  
**Département d'Informatique**  
B.P. 812 Yaoundé

**CC de Base de données distribuées**  
**INF4038**

**Exercice 1 (10 points)**

1. Qu'est-ce qu'une base de données distribuées et quelles sont les motivations d'avoir un tel type de base de données
2. Donner la différence entre une base données distribuée et une base de données parallèle
3. Donner les avantages et les inconvénients d'une base de données distribuées
4. Qu'est-ce qu'une multibase ? Donnez en une architecture de référence

**Exercice 2 (10 points)**

La base de données d'une entreprise d'informatique, InfoNet, a le schéma global suivant:

SERVICE (#service, nom, chef, site)  
PROJET (#projet, #service, nom, chef, budget, durée, date-début)  
EMPLOYE (#emp, #service, nom, prénom, fonction, salaire, prime-annuelle, date-naiss, adresse, #tel)  
InfoNet est localisé sur trois sites à Yaoundé : Yaoundé Centre, Yaoundé Sud, et Yaoundé Nord. Le site de Yaoundé centre tient également lieu de siège pour l'entreprise.

**Question 1 (7pts)**

En supposant que la base est répartie sur les trois sites informatiques du Centre, du Sud, et du Nord, proposer une bonne décomposition de la base InfoNet sur ces trois sites en se basant sur les hypothèses suivantes.

- a) L'attribut site de SERVICE prend une des valeurs suivantes: "Centre", "Sud", et "Nord".
  - b) L'attribut nom de SERVICE prend une des valeurs suivantes: "Commercial", "Financier", "Technique", "Maintenance", "Recherche et Développement", etc.
  - c) #service (resp. #emp) est clé primaire de SERVICE (resp. EMPLOYEE).
  - d) #projet est local à un service.
  - e) Le chef d'un service ou d'un projet est un employé désigné par son numéro.
  - f) Les employés sont affectés à un site donné, sauf pour les employés du service de maintenance qui interviennent dans tous les sites.
  - g) Les données concernant la rémunération ainsi que les informations personnelles des employés sont regroupées et centralisées au siège de l'entreprise.
  - h) Les données concernant la rémunération ainsi que les informations personnelles des employés sont regroupées et centralisées au siège de l'entreprise.
- Donner la définition des différents fragments en utilisant les opérateurs de l'algèbre relationnelle.

**Question 2. (3pts)**

Pour chacune des décompositions, donnez les requêtes de reconstruction

**REPUBLIC OF CAMEROON**  
Peace - Work - Fatherland

**UNIVERSITY OF YAOUNDÉ I**  
**Faculté des Sciences**  
**Department of Computer Science**  
P.O.Box 812 Yaoundé

**Durée 2h (2019/2020)**



Département d'Informatique

## Compilation II : Correction du Contrôle Continu du 2<sup>ème</sup> semestre (2h00) 25 Mai 2018

Etienne Kouokam, Gérard Nzebop

### Exercice 1: Généralités (6 pts)

1.1 Quel est le d'un langage L (1 pt)

**Corrigé:** L'analyse lexicale détecte les unités lexicales (les mots) et les passe en paramètres pour la reconnaissance des grammaires (instructions), ce qu'est censée faire l'analyse syntaxique qui en construit un arbre de dérivation. □

1.2 Donner la définition formelle d'une grammaire. (1 pt)

**Corrigé:** Il s'agit d'un 4-uplet  $G = (V_n, V_t, S, P)$  avec  $\Sigma = V_n \cup V_t$  où :

- $V_t$  = Alphabet des symboles terminaux : Les éléments du langage (variable, identificateur, ...).
- $V_n$  = Symbols auxiliaires dénotant les types de construction (boucle, expression booléenne, ...).
- $S$  = Le but (symbole de départ) appelé axiome : dénote n'importe quelle phrase.
- $P$  = Productions : Les règles de réécriture utilisées pour reconstruire et générer des phrases. Elles sont de la forme
  - $\alpha \rightarrow \beta$  avec  $\alpha \in \Sigma^* V_n \Sigma^*$  et  $\beta \in \Sigma^*$
  - $(\alpha, \beta)$ , où  $\alpha \in \Sigma^*$  et  $\beta \in \Sigma^*$

1.3 Soit la grammaire  $G_0$  dont les règles de production sont :

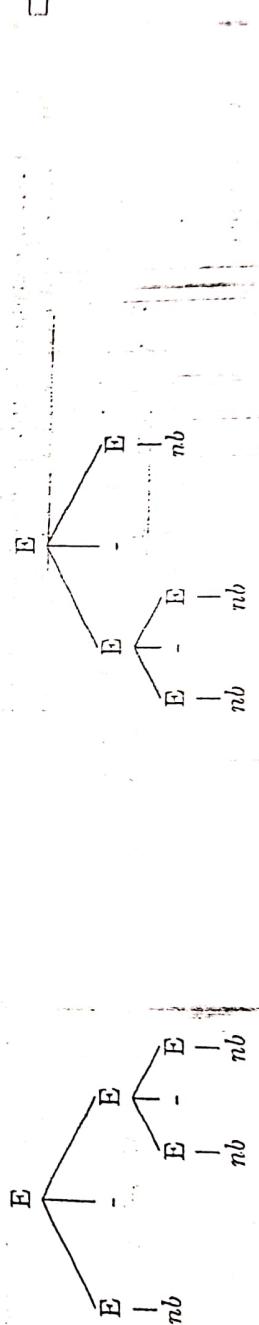
$$E \rightarrow E + E \mid E - E \mid nb$$

1.3.1 Quand dit-on qu'une grammaire est ambiguë ? (1 pt)

**Corrigé:** Une phrase est dite ambiguë s'il existe différentes dérivations gauches pour une même chaîne de terminaux. □

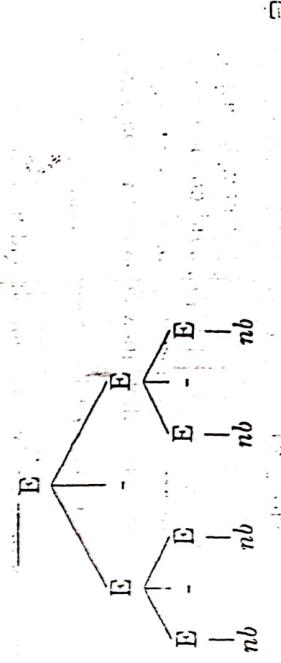
1.3.2 La grammaire  $G_0$  est-elle ambiguë. Justifier (1 pt)

**Corrigé:** Oui, elle l'est. En effet : on peut construire les arbres de dérivations suivants pour la chaîne nb-nb-nb, comme suit :



1.3.3 Soit le mot  $w = nb - nb - nb - nb$ . Construire un arbre abstrait correspondant. (1 pt)

**Corrigé:** L'arbre étant ambigu on peut en construire plusieurs dont celui présenté ci-dessous :



- 1.3.4 Énoncer les règles pour désambiguer cette grammaire puis les appliquer (1 pt.)

*Corrigé:* Elimination de la récursivité gauche + factorisation gauchie ...

- Exercice 2:** Formes normales de Chomsky (CNF) et Greibach (GNF) (5 pts)
- On considère différentes grammaires dont les productions sont les suivantes :

$$G_1 \left\{ \begin{array}{l} S \rightarrow aAa \\ A \rightarrow Sb | bBb \\ B \rightarrow abb | aC \\ C \rightarrow aCA \end{array} \right. \quad G_2 \left\{ \begin{array}{l} S \rightarrow AB | C \\ A \rightarrow aAb | ab \\ B \rightarrow cDd | cd \\ C \rightarrow aCd | aDd \\ D \rightarrow bDc | bc \end{array} \right.$$

2.1 Quel est le langage généré par chacune de ces grammaires ? (2 pts)

- Grammaire  $G_1$  : On peut commencer par simplifier cette grammaire en remarquant que le non terminal  $C$  n'est pas utile. Une fois que ses productions sont éliminées, on obtient que  $L(G) = \{a^n(ababbabb)(ba)^n\}_{n \geq 0}$
  - Grammaire  $G_2$  :
    - Si on part avec la règle  $S \rightarrow AB$ , on génère d'abord autant de  $a$  que de  $b$ , puis autant de  $c$  que de  $d$ ; le langage est donc  $\{a^n b^n c^m d^m\}$ .
    - Si on part avec la règle  $S \rightarrow C$ , on génère d'abord autant de  $a$  que de  $d$ , puis autant de  $b$  que de  $c$ ; le langage est donc  $\{a^n b^n c^m d^m\}$ .
- Finalement, le langage reconnu est  $\{a^n b^n c^m d^m\} \cup \{a^n b^n c^m d^n\}$  pour  $n, m \geq 1$ . □

2.2 Sont-elles sous forme normale de Chomsky ? De Greibach ? Sinon la/les mettre sous ces différentes formes. (3 pts)

- Corrigé:* Aucune des ces grammaires n'est sous forme normale de Chomsky. Elle témoigne la production  $S \rightarrow aAa$  sous forme normale de Chomsky.
- Mise sous forme normale de Chomsky
  - Grammaire  $G$  : Pour la mettre sous forme normale de Chomsky, on introduit les variables  $X_a$  et  $X_b$  et les productions  $X_a \rightarrow a$  et  $X_b \rightarrow b$ . On traite ensuite les productions dont le membre de droite est de longueur supérieur à 2. La grammaire  $G'_1$  que l'on obtient finalement ci-dessous est sous forme normale de Chomsky :

$$G'_1 \left\{ \begin{array}{l} S \rightarrow X_a Y_1, \quad X_a \rightarrow a, \\ X_b \rightarrow b, \quad Y_1 \rightarrow AX_a \\ B \rightarrow X_a Y_2 | X_a C, \quad Y_2 \rightarrow X_b X_b \\ C \rightarrow X_a Y_3, \quad Y_3 \rightarrow CA \\ A \rightarrow SX_b | X_b Y_4, \quad Y_4 \rightarrow BB \end{array} \right.$$

- Grammaire  $G_2$

- Mise sous forme normale de Greibach
- Pour mettre une grammaire sous forme normale de Greibach, on applique l'algorithme vu en cours sur chacune d'elle.
- Pour la grammaire  $G_1$ , on obtient alors la grammaire  $G''_1$  :

$$G''_1 \left\{ \begin{array}{l} S \rightarrow aY_1, \quad X_a \rightarrow a, \\ A \rightarrow aY_1 X_b | bY_4, \quad Y_1 \rightarrow Y_1 X_b X_a | bY_4 X_a, \quad X_b \rightarrow b \\ Y_4 \rightarrow aY_2 B | aCB, \quad Y_2 \rightarrow bX_b \quad R \rightarrow aY_2 | aC \end{array} \right.$$

- Grammaire  $G_2$

- Exercice 3: Automates à Pile (5 pts)**  
Donner l'automate à pile permettant de reconnaître chacun des langages suivants :

3.1  $L_1 = \{a^n b^m c^{2(n+m)} \mid n, m \geq 0\}$  (2.5 pts)

- Corrigé:* Comme on veut deux fois plus de  $c$ , on met deux symboles sur la pile pour chaque  $a$  et chaque  $b$  et on enlève un à chaque  $c$  rencontré. On a besoin de trois états car, quand on commence à lire les  $b$ , on ne peut plus lire de  $a$ . Alors on prend  $Q = \{\rho_0, q_1, q_2\}$ ,  $F = \emptyset$ ,  $\Sigma = \{a, b\}$ ,  $\Gamma = \{A, Z_0\}$  et la fonction de transition est celle définie par le tableau suivant :

3.2  $L_2 = \{u \in \{a, b\}^* \mid u|_a = 2, |u|_b\}$  (2.5 pts)



Département d'Informatique

Rattrapage INF4048, Compilation, Septembre 2020

Dr Etienne KOUOKAM, Dr Thomas MESSI

Questions de cours. (6 points)

1. Automates à pile (3 points)

- (a) Rappeler le Lemme de Pompage pour les langages algébriques puis montrer que le langage  $D = \{ww \mid w \in \{0,1\}^*\}$  n'est pas algébrique. (2 points)

- (b) L'automate des items LR est-il un automate à pile ? Justifier votre réponse. (1 point)

2. Forme Normale de Chomsky (FNC), Forme Normale de Greibach (FNG) (3 points)

- (a) Quand dit-on qu'une grammaire est en FNC ? Rappeler les étapes de mise en FNC. (1.5 points)

- (b) Quand dit-on qu'une grammaire est en FNG ? Rappeler les étapes de mise en FNG. (1.5 points)

Analyseurs syntaxiques. (15 points)

Considérer les grammaires suivantes :

$$(G_1) \begin{cases} A \rightarrow +AA \\ A \rightarrow *AA \\ A \rightarrow b \end{cases} \quad (G_2) \begin{cases} A \rightarrow T \mid T + A \mid T - A \\ T \rightarrow F \mid F * T \\ F \rightarrow i \mid (A) \end{cases}$$

1. Analyseur LL(1) (4 points)

- (a) Construire la table d'analyse LL(1) pour la grammaire  $(G_2)$ . (2 points)

- (b) La grammaire  $(G_2)$  est elle LL(1) ? Justifier votre réponse. (1 point)

- (c) Reconnaître le mot  $i * (i + i)$  pour la grammaire  $(G_2)$  (1 points)

2. Analyseur LR(0) et SLR(1) (5 points)

- (a) Construire l'automate des items LR(0) pour la grammaire  $(G_1)$ . (1 point)

- (b) En déduire les tables d'analyse LR(0) et SLR(1) pour la grammaire  $(G_1)$ . (2 points)

- (c) La grammaire  $(G_1)$  est elle :

i. LR(0) ?

ii. SLR(1) ?

Justifier votre réponse. (1 point)

- (d) Reconnaître le mot  $*b + bb$  pour la grammaire  $(G_1)$  (1 point)

3. Analyseur CLR(1) et LALR(1) (6 points)

- (a) Construire l'automate des items LR(1) pour la grammaire  $(G_1)$ . (2 points)

- (b) En déduire les tables d'analyse CLR(1) et LALR(1) pour la grammaire  $(G_1)$ . (3 points)

- (c) La grammaire  $(G_1)$  est elle :

i. CLR(1) ?

ii. LALR(1) ?

Justifier votre réponse. (1 point)

..... Bon Courage! .....



## Département d'Informatique

INF4048, Analyseurs Syntaxiques, Juillet 2020

Dr Thomas MESSI NGUELE, Assistant Lecturer

N.B. : Utiliser le cours <https://cel.archives-ouvertes.fr/cel-00373150v2/document> pour répondre aux questions suivantes.

### Généralités.

1. Faites un schéma dans lequel vous situer l'analyse syntaxique parmi les étapes de la compilation.
2. Définir l'analyse syntaxique.
3. Définir les notions d'analyse ascendante, analyse descendante. Donner un exemple à chaque fois.
4. Énumérer les générateurs d'analyseurs syntaxiques les plus célèbres et donner le type d'analyseurs correspondant (LL, LR, SLR, LALR).

### Analyseurs LL.

1. Soit  $X$  un symbole d'une grammaire.
  - (a) Définir l'ensemble  $\text{premier}(X)$ , puis donner un algorithme permettant de calculer cet ensemble.
  - (b) Définir l'ensemble  $\text{suivant}(X)$ , puis donner un algorithme permettant de calculer cet ensemble.
2. Considérer les grammaires suivantes :

$$(G_1) \quad \begin{cases} S \rightarrow ABCD \\ A \rightarrow a|\epsilon \\ B \rightarrow CD|b \\ C \rightarrow C|\epsilon \\ D \rightarrow Aa|d|\epsilon \end{cases} \quad (G_2) \quad \begin{cases} S \rightarrow F \\ S \rightarrow (S + F) \\ F \rightarrow 1 \end{cases} \quad (G_3) \quad \begin{cases} E \rightarrow TE' \\ E' \rightarrow +TE'|\epsilon \\ T \rightarrow FT' \\ T' \rightarrow *FT'|\epsilon \\ F \rightarrow i|(E) \end{cases}$$

- (a) Calculer les ensembles  $\text{premier}(X)$  et  $\text{suivant}(X)$  où  $X$  est un symbole de la grammaire  $G_1$ .
- (b) Calculer les ensembles  $\text{premier}(X)$  et  $\text{suivant}(X)$  où  $X$  est un symbole de la grammaire  $G_2$ .
3. Décrire la table d'analyse LL puis dire comment elle est construite.
4. Quand dit-on qu'une grammaire est LL(1) ?
5. La grammaire  $G_3$  est-elle LL(1) ? Justifier votre réponse.
6. Donner l'algorithme de reconnaissance d'un mot à partir d'une table d'analyse LL.
7. Reconnaître le mot  $i + i + i$  pour la grammaire  $(G_3)$

..... Bon Courage ! .....



Département d'Informatique

**INFO416 (Compilation) : Contrôle Continu (2h00)**

25 Avril 2019

Etienne Kouokam

**N.B.** Il ne suffit pas d'affirmer, il faut justifier pour avoir la totalité des points

**EXERCICE 1 [QCM (7 pts = 0.5\*9+2.5)]**

- 1.1 La production  $A \rightarrow aB$  fait référence à une grammaire sous forme normale
  - a. De Chomsky
  - b. De Greibach
  - c. De Boyce-Codd
  - d. Aucune d'elles
  - e. Je ne sais pas
- 1.2 Parmis les formes normales suivantes, quelles sont les grammaires ne contenant aucune récursion gauche
  - a. Chomsky
  - b. Greibach
  - c. Boyce-Codd
  - d. Aucune d'elles
  - e. Je ne sais pas
- 1.3 Toute grammaire sous forme normale de Chomsky est :
  - a. Régulière
  - b. Sensible au contexte
  - c. Hors-contexte
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.4 Quelles sont les productions admissibles pour une grammaire mise sous forme normale de Chomsky ?
  - a.  $A \rightarrow BC$
  - b.  $A \rightarrow a$
  - c.  $S \rightarrow \epsilon$
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.5 Etant donnée la grammaire définie par les productions 1.  $S \rightarrow AS$ , 2.  $S \rightarrow AAS$ , 3.  $A \rightarrow SA$ , 4.  $A \rightarrow aa$ , quelles sont les productions ne respectant pas la forme normale de Chomsky ?
  - a. 2,4
  - b. 1,3
  - c. 1,2,3,4
  - d. 2,3,4
  - e. Je ne sais pas
- 1.6 A quelle catégorie de grammaires appartient la règle de production  $aAb \rightarrow agb$  ?
  - a. Grammaire régulière
  - b. Grammaire hors-contexte
  - c. Grammaire sensible au contexte
  - d. Grammaire récursivement énumérable
  - e. Je ne sais pas
- 1.6 Quels sont les énoncés erronés parmi les suivants :
  - a. Les langages contextuels contiennent les langages hors-contextes
  - b. Tout langage régulier est contenu dans un langage contextuel
  - c. Les langages récursivement énumérables contiennent les langages réguliers
  - d. Les langages hors-contextes contiennent les langages contextuels
  - e. Je ne sais pas
- 1.7 Considérant la production  $S \rightarrow 0S1|\epsilon$  où  $\Sigma = \{0, 1\}$ , dire parmi les langages suivants ceux qui ne correspondent pas à la grammaire
  - a. Grammaire non régulière
  - b.  $\{0^n 1^n \mid n \geq 0\}$
  - c.  $\{0^n 1^n \mid n \geq 1\}$
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.8 Les grammaires ambiguës sont-elles hors-contextes ?
  - a. Oui
  - b. Non
  - c. Je ne sais pas
- 1.9 Un langage reconnu par un automate à pile est de
  - a. Type 0
  - b. Type 1
  - c. Type 2
  - d. Type 3
  - e. Je ne sais pas
- 1.10 On considère la grammaire définie par les règles de production suivantes :  $S \rightarrow aB|bA$ ,  $B \rightarrow aBB|bS|b$ .  
 $bAA|aS|a$ . Parmi les mots suivants, lesquels peuvent être générés par une telle grammaire ?
  - a. aaaabb
  - b. aabbbb
  - c. aabbab
  - d. abbba
  - e. Je ne sais pas

---

## EXERCICE 2 [Melting pot ... (14 pts)]

On considère différentes grammaires dont les productions sont les suivantes :

$$G_1 : \begin{cases} S \rightarrow XA|BB \\ B \rightarrow SB|b \\ X \rightarrow b \\ A \rightarrow a \end{cases} \quad G_2 : \begin{cases} S \rightarrow A \\ A \rightarrow aBa|a \\ B \rightarrow bAb|b \end{cases} \quad G_3 : \begin{cases} S \rightarrow aMb \\ M \rightarrow aM\beta|\epsilon \end{cases} \quad G_4 : \begin{cases} S \rightarrow ASA|aB \\ A \rightarrow B|S \\ B \rightarrow b|\epsilon \end{cases}$$

- 2.1 Quel est le langage généré par chacune de ces grammaires? (0.5\*4 pts)
- 2.2 Transformer chacune des grammaires  $G_i, 1 \leq i \leq 4$  en une grammaire équivalente  $G'_i, 1 \leq i \leq 4$  (si possible sans  $\epsilon$ ) sans symboles inutiles, sans  $\epsilon$ -productions et sans règles unitaires (1\*4 pts)
- 2.3 Les grammaires  $G'_i, 1 \leq i \leq 4$  ainsi obtenues sont-elles de Chomsky? De Greibach? Sinon la/les mettre sous CNF/GNF. (4 pts)
- 2.4 On reconside les grammaires  $G_i, 1 \leq i \leq 4$
- 2.4.1 Y en a-t-il qui soient récursives à gauche? Si oui, lesquelles? Justifier. (2 pts)
- 2.4.2 S'il y a lieu, transformer lesdites grammaires de façon à supprimer la récursivité gauche (2 pts)

Bon courage!!!



Département d'Informatique

Examen INF4048, Compilation, Août 2020

Dr Etienne KOUOKAM, Dr Thomas MESSI

Questions de cours. (4 points)

1. Expliquer pourquoi un automate fini peut reconnaître le langage  $a^n b^m, n, m \geq 0$  mais pas le langage  $a^n b^n, n \geq 0$  (1 point)
2. Donner le schéma simple d'un automate à pile (ruban, pile et cerveau) (1 point)
3. Donner la différence entre les analyseurs LL et LR. (1 point)
4. Citer deux générateurs d'analyseurs syntaxiques célèbres puis donner le type d'analyseur correspondant (LL, LR, SLR, LALR). (1 point)

Analyseurs syntaxiques. (18 points)

Considérer les grammaires suivantes :

$$(G_1) \begin{cases} E \rightarrow T \\ E \rightarrow (E * T) \\ T \rightarrow 0 \end{cases} \quad (G_2) \begin{cases} A \rightarrow TA' \\ A' \rightarrow +TA'|\epsilon \\ T \rightarrow FT' \\ T' \rightarrow *FT'|\epsilon \\ F \rightarrow e|(A) \end{cases}$$

1. Analyseur LL(1) (5 points)

- (a) Construire la table d'analyse LL(1) pour la grammaire  $(G_2)$ . (2.5 points)
- (b) La grammaire  $(G_2)$  est-elle LL(1)? Justifier votre réponse. (1 point)
- (c) Reconnaître le mot  $e * (e + e)$  pour la grammaire  $(G_2)$  (1.5 points)

2. Analyseur LR(0) et SLR(1) (7 points)

- (a) Construire l'automate des items-LR(0) pour la grammaire  $(G_1)$ . (2 points)
- (b) En déduire les tables d'analyse LR(0) et SLR(1) pour la grammaire  $(G_1)$ . (1.5 \* 2 points)
- (c) La grammaire  $(G_1)$  est-elle :
  - i. LR(0) ?
  - ii. SLR(1) ?

Justifier votre réponse. (1 point)

- (d) Reconnaître le mot  $(0 * 0)$  pour la grammaire  $(G_1)$  (1 point)

3. Analyseur CLR(1) et LALR(1) (6 points)

- (a) Construire l'automate des items LR(1) pour la grammaire  $(G_1)$ . (2 points)
- (b) En déduire les tables d'analyse CLR(1) et LALR(1) pour la grammaire  $(G_1)$ . (3 points)
- (c) La grammaire  $(G_1)$  est-elle :
  - i. CLR(1) ?
  - ii. LALR(1) ?

Justifier votre réponse. (1 point)

..... Bon Courage! .....



Département d'Informatique

## INF4048 (Compilation) : Contrôle Continu (2h00)

Août 2020

Etième Kouokam, Thomas Messi

N.B. Il ne suffit pas d'affirmer, il faut justifier pour avoir la totalité des points

- Faites attention à ne sélectionner que les réponses correctes. Autrement, ça pourrait vous coûter des points
- Pour chaque question posée, il peut y en avoir zéro, une ou plusieurs réponses justes.
- Portez vos noms et prénoms sur chacune de vos feuilles (même de brouillon) avant toute chose

EXERCICE 1 [QCM (6 pts = 0.5\*10-1)]

- 1.1. Etant donnée la grammaire définie par les productions 1.  $S \rightarrow AS$ , 2.  $S \rightarrow AAS$ , 3.  $A \rightarrow SA$ , 4.  $A \rightarrow aa$ , quelles sont les productions ne respectant pas la forme normale de Chomsky ?
  - a. 2,3,4
  - b. 1,3
  - c. 1,2,3,4
  - d. 2,4
  - e. Je ne sais pas
- 1.2. Considérez la production  $S \rightarrow 0S1|\epsilon$  où  $\Sigma = \{0,1\}$ , dire parmi les langages suivants ceux qui ne correspondent pas à la grammaire
  - a. Grammaire non régulière
  - b.  $\{0^n1^n \mid n \geq 1\}$
  - c.  $\{0^nr1^r \mid n \geq 0\}$
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.3. Parmis les formes normales suivantes, quelles sont les grammaires ne contenant aucune récursion gauche ?
  - a. Chomsky
  - b. Boyce-Codd
  - c. Greibach
  - d. Aucune d'elles
  - e. Je ne sais pas
- 1.4. La production  $A \rightarrow aB$  fait référence à une grammaire sous forme normale
  - a. De Chomsky
  - b. De Boyce-Codd
  - c. De Greibach
  - d. Aucune d'elles
  - e. Je ne sais pas
- 1.5. A quelle catégorie de grammaires appartient la règle de production  $aAb \rightarrow agb$  :
  - a. Grammaire sensible au contexte
  - b. Grammaire hors-contexte
  - c. Grammaire régulière
  - d. Grammaire récursivement énumérable
  - e. Je ne sais pas
- 1.6. Tout grammaire sous forme normale de Chomsky est :
  - a. Hors-contexte
  - b. Sensible au contexte
  - c. Régulière
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.7. Quelles sont les productions admissibles pour une grammaire mise sous forme normale de Chomsky ?
  - a.  $A \rightarrow BC$
  - b.  $A \rightarrow a$
  - c.  $S \rightarrow \epsilon$
  - d. Rien de tout ça
  - e. Je ne sais pas
- 1.8. Un langage reconnu par un automate à pile est de
  - a. Type 3
  - b. Type 2
  - c. Type 1
  - d. Type 0
  - e. Je ne sais pas
- 1.9. Les grammaires ambiguës sont-elles hors-contextes ?
  - a. Non
  - b. Oui
  - c. Je ne sais pas
- 1.10. Quels sont les énoncés erronés parmi les suivants :
  - a. Les langages contextuels contiennent les langages hors-contextes
  - b. Les langages hors-contextes contiennent les langages contextuels
  - c. Les langages récursivement énumérables contiennent les langages réguliers
  - d. Tout langage régulier est contenu dans un langage contextuel
  - e. Je ne sais pas
- 1.11. On considère la grammaire définie par les règles de production suivantes :  $S \rightarrow aB|bA$ ,  $B \rightarrow aBB|bS|b$ ,  $A \rightarrow bAA|aS|a$ . Parmi les mots suivants, lesquels peuvent être générés par une telle grammaire ?
  - a. aaaabb
  - b. aabbba
  - c. aabbab
  - d. abbbba
  - e. Je ne sais pas

---

**EXERCICE 2 [Melting pot... (14 pts)]**

On considère différentes grammaires dont les productions sont les suivantes :

$$G_1 : \begin{cases} S & \rightarrow XA|BB \\ B & \rightarrow SB|b \\ X & \rightarrow b \\ A & \rightarrow a \end{cases} \quad G_2 : \begin{cases} S & \rightarrow ABa \\ A & \rightarrow aab \\ B & \rightarrow Ab \end{cases} \quad G_3 : \begin{cases} S & \rightarrow ahSb \\ S & \rightarrow aa \end{cases} \quad G_4 : \begin{cases} S & \rightarrow aS \\ S & \rightarrow aSbS \\ S & \rightarrow \epsilon \end{cases}$$

2.1 Quel est le langage généré par chacune des grammaires  $G_1$  et  $G_3$  ?

2.2 Y en a-t-il qui soient ambiguës ? Si oui, précisez lesquelles en justifiant

2.3 Transformer chacune des grammaires  $G_i$ ,  $1 \leq i \leq 4$  en une grammaire équivalente  $G'_i$ ,  $1 \leq i \leq 4$  (si possible sans  $\epsilon$ ) sans symboles inutiles, sans  $\epsilon$ -productions et sans règles unitaires

2.4 Les grammaires  $G'_i$ ,  $1 \leq i \leq 4$  ainsi obtenues Sont-elles de Chomsky? De Greibach? Simplement la/les mettre sous CNF/GNF.

2.5 On reconside<sup>r</sup>e les grammaires  $G_i$ ,  $1 \leq i \leq 4$

2.5.1 Y en a-t-il qui soient récursives à gauche ? Si oui, lesquelles ? Justifier.

2.5.2 S'il y a lieu, transformer lesdites grammaires de façon à supprimer la récursivité gauche (1.5 pts)

**EXERCICE 3 [Automates à piles ... (5 pts)]**

Sur  $\Sigma = \{a, b, c\}$ , on considère le langage  $L = \{wccw' \mid w, w' \in \{a, b\}^* \wedge |w| = |w'|\}$ .

3.1 Montrer que ce langage n'est pas reconnaissable par un automate fini.

3.2 Donner un automate à pile reconnaissant ce langage.

3.3 Donner une grammaire pour ce langage.

Bon courage!!!

(1.5 pts)

(2.5 pts)

(1 pt)

(0.5\*2 pts)

(2 pts)

(3 pts)

(3 pts)



Département d'Informatique

INF4048 (Compilation II) : Fiche de TD N°1

20 mars 2020

Etienne Kouokam

EXERCICE 1 [Généralités sur les grammaires]

1.1 Construire une grammaire pour chacun des langages suivants :

1.1.1  $L_1 = \text{Le langage des palindromes sur l'alphabet } \{0, 1\}$ .

1.1.2  $L_2 = \{w \in \{0, 1\}^* \mid w = w^R \text{ et } w \text{ est pair}\}$

1.1.3  $L_3 = \{w \in \{0, 1\}^* \mid w \text{ contient au moins trois } 1s\}$

1.1.4  $L_4 = \{a^i b^j c^k \mid i, j, k > 0 \text{ et } i + j = k\}$

1.2 Etant donnée la grammaire  $G_1 \left\{ \begin{array}{l} S \rightarrow (L) \mid a \\ L \rightarrow L, S \mid a \end{array} \right.$

1.2.1 Quels sont les symboles terminaux et non terminaux ?

1.2.2 Donner les arbres de dérivation pour chacun des cas suivants : (a,a); (a,(a,a)); (a,((a,a),(a,a)))

1.2.3 Construire une dérivation à gauche et une dérivation à droite pour chacune des phrases de la question précédente.

1.3 Etant donnée la grammaire  $G_2 \left\{ \begin{array}{l} S \rightarrow aB \mid bA \\ A \rightarrow a \mid aS \mid bAA \\ B \rightarrow b \mid bS \mid aBB \end{array} \right.$

1.3.1 Trouver pour le mot aaabbabbba une dérivation à gauche, une dérivation à droite et un arbre de dérivation.

1.3.2 Montrer par récurrence sur  $|w|$  que  $L(G)$  est l'ensemble des mots de longueur non nulle qui contiennent autant de a que de b.

1.3.3 Construire une dérivation à gauche et une dérivation à droite pour chacune des phrases de la question précédente.

EXERCICE 2 [Simplification dans les grammaires]

On considère différentes grammaires dont les productions sont les suivantes :

$$G_1 \left\{ \begin{array}{l} S \rightarrow aAAB \\ S \rightarrow CC \mid cA \\ A \rightarrow uA \mid a \\ C \rightarrow cC \end{array} \right. \quad G_2 \left\{ \begin{array}{l} S \rightarrow AB \\ A \rightarrow aA \mid c \\ B \rightarrow b \mid \epsilon \end{array} \right. \quad G_3 \left\{ \begin{array}{l} S \rightarrow ABC \\ A \rightarrow b \\ B \rightarrow b \mid c \\ C \rightarrow E \mid \epsilon \end{array} \right. \quad G_4 \left\{ \begin{array}{l} S \rightarrow AB \mid CA \\ A \rightarrow a \mid b \mid \epsilon \\ B \rightarrow BC \mid DB \\ C \rightarrow E \mid \epsilon \\ D \rightarrow a \mid d \\ E \rightarrow ab \mid c \mid d \mid \epsilon \end{array} \right.$$

2.1 Supprimer (en expliquant) les symboles inutiles de la grammaire  $G_1$

2.2 Supprimer (en expliquant) les  $c$ -productions de la grammaire  $G_2$

2.3 Supprimer (en expliquant) les règles unitaires de la grammaire  $G_3$ .

2.4 Transformer la grammaire  $G_1$  en une grammaire équivalente (sans  $c$ ) sans symboles inutiles, sans  $c$ -productions et sans règles unitaires

### EXERCICE 3 [Formes normales de Chomsky (CNF) et Greibach (GNF)]

On considère différentes grammaires dont les productions sont les suivantes :

$$G_1 : S \rightarrow aSb \mid c \quad G_2 \left\{ \begin{array}{l} S \rightarrow ASB \mid c \\ B \rightarrow b \end{array} \right. \quad G_3 \left\{ \begin{array}{l} S \rightarrow aB \mid bA \\ B \rightarrow a \mid aS \mid bAA \\ A \rightarrow b \mid bS \mid aBB \end{array} \right. \quad G_4 \left\{ \begin{array}{l} S \rightarrow AB \mid C \\ A \rightarrow aAb \mid ab \\ B \rightarrow cBd \mid cd \\ C \rightarrow aCd \mid aDd \\ D \rightarrow bDc \mid bc \end{array} \right.$$

3.1 Quel est le langage généré par chacune de ces grammaires? Qu'en déduit-on?

3.2 Sont-elles de Chomsky? De Greibach? Sinon la/les mettre sous CNF/GNF.

3.4 Mettre la grammaire suivante sous forme normale de Greibach

$$G_4 \left\{ \begin{array}{l} S \rightarrow XA \mid BB \\ B \rightarrow b \mid SB \\ X \rightarrow b \\ A \rightarrow a \end{array} \right.$$

### EXERCICE 4 [Opération de clôture & Lemme de pompage]

En vous servant du lemme de pompage adapté aux langages hors-contexte, montrer que les langages suivant ne sont pas hors-contexte :

4.1 Montrer que les langages hors-contexte sont clos pour l'Union, la concaténation et l'étoile

4.2  $L_2 = \{a^n b^n c^i \mid i \leq n\}$ .

4.3  $L_2 = \{0^p \mid p \text{ est premier}\}$

4.4  $L_2 = \{w \mid |w|_a < |w|_b < |w|_c\}$