

Théorie des langages et de la compilation
Année académique 2007-2008
1ère licence en Informatique et
2ème année du grade de l'ingénieur civil informaticien
Examen de première session

Remarques préliminaires

- On vous demande de répondre à chaque question sur des feuilles séparées (les correcteurs pouvant être différents).
- N'oubliez pas d'inscrire vos nom et prénom sur chacune des feuilles.
- Vous ne pouvez pas utiliser de notes.
- L'examen comporte cinq questions pour un total de 16 points, les projets (4 points) complétant la note.
- Vous disposez de quatre heures. Pensez à gérer correctement votre temps.
- Si rien n'est spécifié pour les formalismes, les conventions utilisées habituellement dans le cours sont prises.

Question 1 (1pts)

1. Donnez la définition d'une grammaire G et de son langage $L(G)$ associé.
2. Définissez la hiérarchie de Chomsky pour les grammaires.

Question 2 (5pts)

Soit $\Sigma = \{0, 1, 2\}$ l'alphabet comprenant les 3 symboles '0', '1' et '2' et le langage sur Σ

$$L = \{0^n 1^n 2^n \mid n \in \mathbb{N}\}$$

1. Montrez que ce langage peut être formellement défini comme $L(G)$ pour la grammaire $G = \langle N, \Sigma, S, P \rangle$ avec
 - $N = \{S, T, B\}$
 - $P =$
 - (a) $S \rightarrow 0SB$
 - (b) $S \rightarrow T$
 - (c) $TB \rightarrow 1T2$
 - (d) $2B \rightarrow B2$
 - (e) $T \rightarrow \epsilon$
2. Parmi les formalismes suivants :
 - (a) les automates finis déterministes
 - (b) les automates à pile
 - (c) les machines de Turing
 - (d) les expressions régulières
 lesquels permettent de définir ce langage L (**celui défini plus haut**) (précisez si nécessaire, mais sans démonstration)?
 Pour chacun des formalismes permettant de définir L , donnez une définition formelle complète pour L dans ce formalisme.
3. Énoncez et démontrez formellement le lemme de pompage (*pumping lemma*) des langages *context-free*.
4. Peut-on définir L avec une grammaire *context-free*? (Dé)montrez.

Question 3 (3pts)

La grammaire ci-dessous est-elle $LL(1)$? Justifiez votre réponse. Si possible, et si nécessaire, transformez la en grammaire $LL(1)$ « nettoyée » et équivalente. Donnez la table des actions de l'analyseur $LL(1)$ qui lui correspond. Veuillez à bien détailler les étapes de calcul.

$$\begin{aligned} S &\rightarrow aAb \mid Bb \mid cB \mid D \\ A &\rightarrow Aa \mid c \\ B &\rightarrow fB \mid eCF \mid \varepsilon \\ C &\rightarrow cDE \mid cSE \\ D &\rightarrow ef \mid eg \mid d \\ E &\rightarrow gC \\ F &\rightarrow f \end{aligned}$$

Question 4 (3pts)

Donnez l'automate canonique $LR(1)$ de la grammaire suivante :

$$\begin{aligned} S &\rightarrow cS \mid T \\ T &\rightarrow aCa \mid bCb \mid aDb \mid bDa \\ C &\rightarrow z \\ D &\rightarrow z \end{aligned}$$

Répondez ensuite aux question suivantes, en justifiant vos réponses à l'aide de l'automate : cette grammaire est-elle $LR(0)$? Est-elle $LALR(1)$?

Finalement, donnez les tables de l'analyseur $LR(1)$ pour cette grammaire, et simulez en le fonctionnement sur la chaîne `cccazb`.

Question 5 (4pts)

Expliquez de façon succincte (en restant au niveau des principes) les différentes étapes de la compilation en expliquant les outils étudiés dans le cours. Illustrez *brièvement* ces différentes étapes en vous aidant du programme ci-dessous.

```
int fact(int z)
{
    if (z>0)
        return z*fact(z-1);
    else
        return 1;
}
main()
{
    int x = 5;
    int y;
    y=fact(x);
}
```