

# Correction du Partiel THL

## THÉORIE DES LANGAGES

EPITA – Promo 2015 – Sans document ni machine

Janvier 2013 (1h30)

**Correction:** Le sujet et sa correction ont été écrits par Akim Demaille.

**Barème:** Toutes les questions rédactionnelles sont notées sur 4, et c'est ainsi qu'il faut lire les barèmes qui suivent. Se reporter à la feuille de calcul pour voir les coefficients donnés aux questions.

**Best-of:** Le best-of est tiré des copies des étudiants. Les erreurs, en particulier d'orthographe, sont les leurs ! Les commentaires du correcteur sont notés [ainsi].

Bien lire les questions, chaque mot est important.

Répondre aux questions à choix multiples (numérotées Q.1, Q.2, etc.) sur les formulaires de QCM ; aucune réponse manuscrite ne sera corrigée. Renseigner les champs d'identité. Sauf mention contraire, il y a exactement une et une seule réponse juste pour ces questions. Si plusieurs réponses sont valides, sélectionner la plus restrictive. Par exemple s'il est demandé si 0 est *nul*, *non nul*, *positif*, ou *négalif*, cocher *nul* qui est plus restrictif que *positif* et *négalif*, tous deux vrais.

## 1 Incontournables

Chaque erreur (ou non réponse) aux trois questions suivantes retire 1/6 de la note finale. Avoir tout faux divise donc la note par 2.

Q.1 Si  $\{a^n b^n \mid n \in \mathbb{N}\} \subseteq L$ , alors  $L$  n'est pas rationnel.

vrai  faux

**Correction:** Comme toujours,  $l = \Sigma^*$  est un bon contre-exemple.

Q.2 Combien existe-t-il de sous-ensembles de  $\{1, 2, \dots, n\}$  ?

$n!$    $2^n$    $\frac{n(n+1)}{2}$    $n^2$    $\frac{n(n-1)}{2}$

Q.3 Si une grammaire n'est pas LR(1), alors elle est ambiguë.

vrai  faux

**Correction:** La grammaire de Yacc, sans terminateur, est un bon exemple.

## 2 Contrôle

Pour les questions suivantes, une réponse fautive entraîne une pénalité. Pas de réponse donne 0.

Q.4 Quelle chaîne n'engendre pas l'expression rationnelle étendue suivante ?

$[ - + ] ? 0 [ x X ] [ a - f A - F 0 - 9 ] * [ a - f A - F 0 - 9 ] ( \setminus . [ a - f A - F 0 - 9 ] [ a - f A - F 0 - 9 ] * ) ? ( [ p P ] [ - + ] ? [ a - f A - F 0 - 9 ] [ 0 - 9 A - F a - f ] * ) ?$

0x3.243\_F6a\_888\_5a3\_08D\_313\_19

- ✗ -0X\_Dead\_BeeF\_\_\_1.2P-Deaf\_Bee
- ✗ 0x11.0010\_0100001111110\_1101010100010001000010110100011\_\_
- ✓ +0x\_3\_.14\_15\_92\_P01

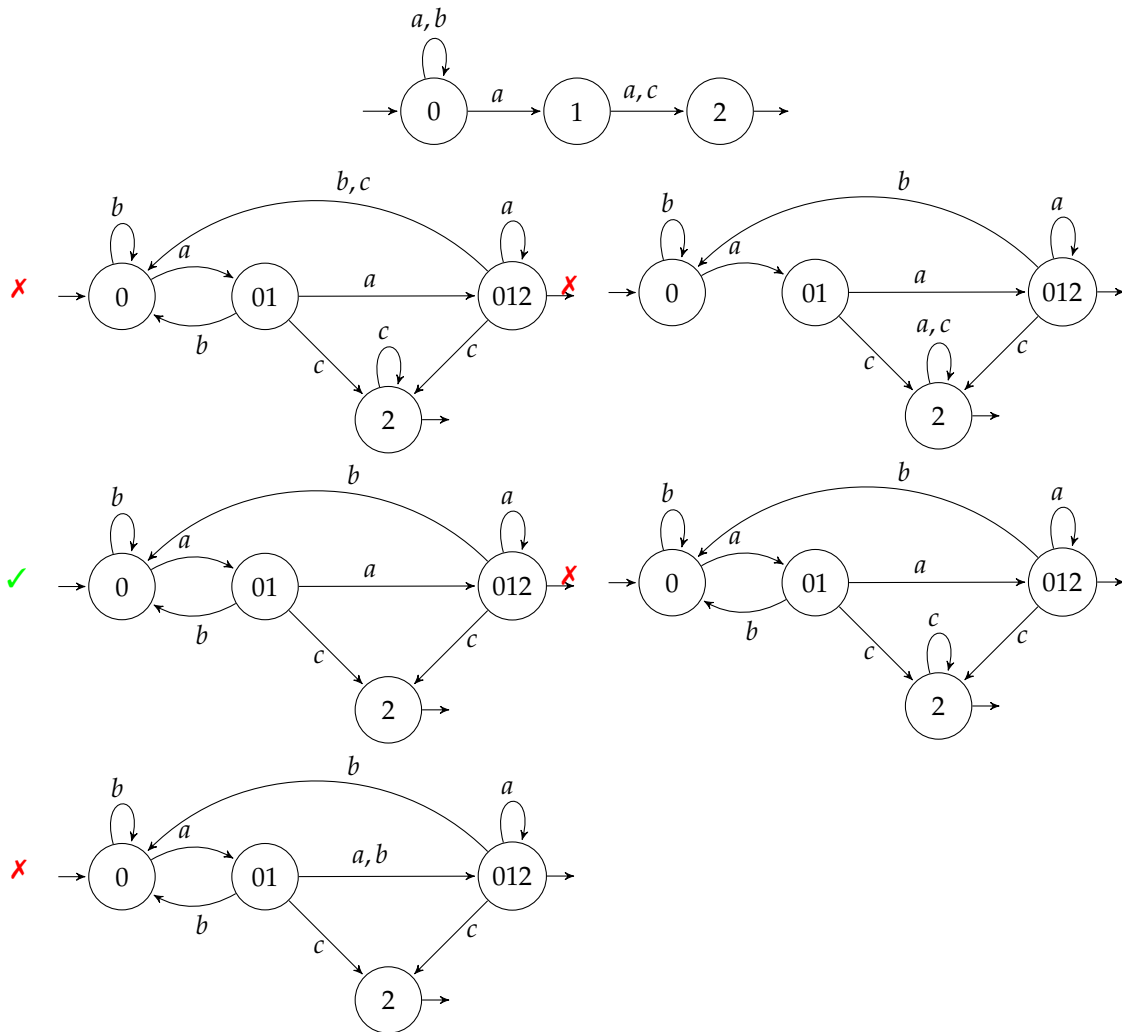
**Correction:** Il ne peut pas y avoir de '\_' devant '.'.

Q.5 Combien d'états compte l'automate de Thompson de  $(a + b)^* a^4 (a + b)$  ?

- ✗ 18
- ✗ 20
- ✓ 22
- ✗ 27
- ✗ 28

**Correction:** Il y a 11 symboles qui comptent  $(a + b * a a a a + b)$ .

Q.6 Quel automate obtient-on en déterminisant l'automate suivant (sans forcer la complétude) ?



Q.7 Quelle est la classe de Chomsky de la grammaire suivante ?

- |                 |                        |                     |                     |
|-----------------|------------------------|---------------------|---------------------|
|                 | $A \rightarrow aABC$   | $CB \rightarrow BC$ | $bC \rightarrow bc$ |
| ✗ Rationnelle   | $A \rightarrow abc$    | $bB \rightarrow bb$ | $cC \rightarrow cC$ |
| ✗ Hors contexte | ✓ Sensible au contexte | ✗ Générale          |                     |
|                 | ✓ Monotone             |                     |                     |

Q.8 Quelle est la classe du langage qu'elle engendre ?

- ✗ Rationnelle
- ✓ Sensible au contexte
- ✗ Hors contexte
- ✗ Générale

**Correction:** C'est  $\{a^n b^n c^n | n \in \mathbb{N}\}$ .

Q.9 Quelle est la classe du langage engendré par la grammaire suivante ?

- ✓ Rationnelle  $E \rightarrow E + T \mid E \times T \mid T, T \rightarrow T + E \mid T - F \mid F \quad F \rightarrow -F \mid n$
- ✗ Hors contexte
- ✗ Générale

**Correction:** C'est  $-^*n((+|-|*|/)-^*n)^*$ .

Q.10 Dans une analyse classique en utilisant Lex et Yacc :

- ✗ on fait plusieurs fois l'appel `yyparse(yylex())`
- ✗ on appelle la fonction `yyparse` plusieurs fois, elle appelle la fonction `yylex` chaque fois
- ✓ on appelle la fonction `yyparse` une fois, elle appelle la fonction `yylex` plusieurs fois
- ✗ on appelle la fonction `yylex` une fois, elle appelle la fonction `yyparse` plusieurs fois
- ✗ on appelle la fonction `yylex` plusieurs fois, puis la fonction `yyparse` une fois

Q.11 Avec la grammaire suivante, quel état atteint l'automate LR(1) après une transition sur E puis sur '?' ?

- |  |   |  |
|--|---|--|
| $S \rightarrow E \$$   |   |  |
| $E \rightarrow E ? E : E \mid E + E \mid 0$  |   |  |
| <ul style="list-style-type: none"> <li>✗ <math>E \rightarrow E ? \bullet E : E \quad [\\$?+]</math></li> <li><math>E \rightarrow \bullet E ? E : E \quad [\\$?+ :]</math></li> <li><math>E \rightarrow \bullet E + E \quad [\\$?+ :]</math></li> <li><math>E \rightarrow \bullet 0 \quad [\\$?+ :]</math></li> </ul>   | <ul style="list-style-type: none"> <li>✓ <math>E \rightarrow E ? \bullet E : E \quad [\\$?+]</math></li> <li><math>E \rightarrow \bullet E ? E : E \quad [?+ :]</math></li> <li><math>E \rightarrow \bullet E + E \quad [?+ :]</math></li> <li><math>E \rightarrow \bullet 0 \quad [?+ :]</math></li> </ul> | <ul style="list-style-type: none"> <li>✗ <math>S \rightarrow E \bullet \\$ \quad [\\$]</math></li> <li><math>E \rightarrow E \bullet ? E : E \quad [\\$?+]</math></li> <li><math>E \rightarrow E \bullet + E \quad [\\$?+]</math></li> <li>✗ <math>S \rightarrow E \bullet \\$ \quad [\\$]</math></li> <li><math>E \rightarrow E \bullet ? E : E \quad [\\$?+ :]</math></li> <li><math>E \rightarrow E \bullet + E \quad [\\$?+ :]</math></li> </ul> |
| <ul style="list-style-type: none"> <li>✗ <math>E \rightarrow E ? \bullet E : E \quad [\\$?+]</math></li> <li><math>S \rightarrow \bullet E \\$ \quad [\\$]</math></li> <li><math>E \rightarrow \bullet E ? E : E \quad [\\$?+ :]</math></li> <li><math>E \rightarrow \bullet E + E \quad [\\$?+ :]</math></li> <li><math>E \rightarrow \bullet 0 \quad [\\$?+ :]</math></li> </ul> |   |  |

### 3 Logique Propositionnelle

Soit le langage de la logique propositionnelle, composé de deux symboles  $\top$  (vrai) et  $\perp$  (faux), de l'opération unaire  $\neg$  (non), des opérations binaires  $\vee$  (ou) et  $\wedge$  (et), et des parenthèses notées  $[, ]$ . Ce langage inclut des mots tels que  $\perp \wedge \perp$ ,  $\top \vee \perp$  et  $\neg\neg[\top \wedge \top] \vee [\perp \wedge \perp]$ .

Q.12 Que dire de la grammaire suivante ?

- ✗ rationnelle  $S \rightarrow S \wedge S \mid S \vee S \mid \neg S \mid [S] \mid \top \mid \perp$
- ✗ non ambiguë
- ✗ infiniment ambiguë

Q.13 Dans la grammaire suivante, quelles sont les priorités/associativités des opérateurs ?

- ✓  $\wedge$  et  $\vee$  associatives à gauche, priorités croissantes :  $\vee < \wedge < \neg$
- ✗  $\wedge$  et  $\vee$  associatives à droite, priorités croissantes :  $\vee < \wedge < \neg$
- ✗  $\wedge$  et  $\vee$  associatives à gauche, priorités croissantes :  $\neg < \wedge < \vee$
- ✗  $\wedge$  et  $\vee$  associatives à droite, priorités croissantes :  $\neg < \wedge < \vee$

Q.14 Que dire de la grammaire de la question 13 ?

- ✗ ambiguë et LL(1)
- ✗ non ambiguë et LL(1)
- ✗ ambiguë et non LL(1)
- ✓ non ambiguë et non LL(1)

Q.15 Que dire de la grammaire suivante par rapport à celle de la question 13 ?

- ✗ même langage, mêmes priorités et associativités, mais LL(1)
- ✗ même langage, mêmes priorités et associativités, pas LL(1)
- ✓ même langage, priorités et/ou associativités différentes, mais LL(1)
- ✗ même langage, priorités et/ou associativités différentes, pas LL(1)

✗ langage différent

Q.16 Quels sont les symboles annulables dans la grammaire de la question 15.

✗  $S, S', T, T', F$    ✗  $S, T, F$    ✗  $S', T', F$    ✓  $S', T'$    ✗  $F$

Q.17 Quels sont les FIRST dans la grammaire de la question 15.

✗	FIRST	✗	FIRST	✓	FIRST	✗	FIRST	✗	FIRST
$S$	$\neg[\top\perp]$	$S$	$\neg[\top\perp]$	$S$	$\neg[\top\perp]$	$S$	$\neg[\top\perp]$	$S$	$T$
$S'$	$\vee\wedge$	$S'$	$\varepsilon\vee\wedge$	$S'$	$\vee$	$S'$	$\varepsilon\vee$	$S'$	$\vee$
$T$	$\neg[\top\perp]$	$T$	$\neg[\top\perp]$	$T$	$\neg[\top\perp]$	$T$	$\neg[\top\perp]$	$T$	$F$
$T'$	$\vee\wedge$	$T'$	$\varepsilon\vee\wedge$	$T'$	$\wedge$	$T'$	$\varepsilon\wedge$	$T'$	$\wedge$
$F$	$\neg[\top\perp]$	$F$	$\neg[\top\perp]$	$F$	$\neg[\top\perp]$	$F$	$\neg[\top\perp]$	$F$	$\neg[\top\perp]$

Q.18 Quels sont les FOLLOW dans la grammaire de la question 15.

✗	FOLL	✗	FOLL	✗	FOLL	✓	FOLL	✗	FOLL
$S$	$]$	$S$	$]$	$S$	$]$	$S$	$]$	$S$	$\varepsilon]$
$S'$	$]$	$S'$	$]$	$S'$	$\vee\wedge]$	$S'$	$]$	$S'$	$\vee\wedge]$
$T$	$\vee]$	$T$	$]$	$T$	$\vee]$	$T$	$\vee]$	$T$	$\vee]$
$T'$	$\vee]$	$T'$	$]$	$T'$	$\vee]$	$T'$	$\vee]$	$T'$	$\vee]$
$F$	$\vee]$	$F$	$]$	$F$	$\wedge\vee]$	$F$	$\wedge\vee]$	$F$	$\wedge\vee]$

Q.19 Que dire de la grammaire étendue suivante par rapport à celle de la question 13 ?

- ✓ même langage, mêmes priorités et associativités, mais LL(1)  $S \rightarrow T(\vee T)^* \quad T \rightarrow F(\wedge F)^* \quad F \rightarrow \neg^F | [S] | \top | \perp$
- ✗ même langage, mêmes priorités et associativités, pas LL(1)
- ✗ même langage, priorités et/ou associativités différentes, mais LL(1)
- ✗ même langage, priorités et/ou associativités différentes, pas LL(1)
- ✗ langage différent

Q.20 Quelle routine parse et calcule correctement S pour la grammaire de la logique booléenne de la question 19? La variable la désigne le lookahead courant, et la routine eat (expect) vérifie que le lookahead actuel est expect puis stocke le suivant dans la.

<p>✗</p> <pre> bool S() {     bool res = T();     while (la == 'v')     {         eat('v');         res  = F();         while (la == '^')         {             eat('^');             res &amp;= F();         }     }     return res; }                 </pre>	<p>✗</p> <pre> bool S() {     bool res = T();     while (la == 'v')     {         res  = T();         eat('v');     }     return res; }                 </pre>	<p>✗</p> <pre> bool S() {     bool res = false;     do     {         eat('v');         res  = T();     }     while (la == 'v');     return res; }                 </pre>
<p>✓</p> <pre> bool S() {     bool res = T();     while (la == 'v')     {         eat('v');         res  = T();     }     return res; }                 </pre>	<p>✗</p> <pre> bool S() {     bool res = true;     do     {         eat('v');         res  = T();     }     while (la == 'v');     return res; }                 </pre>	

Q.21 Terminer la séquence de décalages/réductions suivante pour un parser Yacc/Bison implémentant la grammaire de la question 12 avec des directives précisant correctement priorités et associativités.

<p style="color: red; font-weight: bold; margin: 0;">X</p> <pre style="border: 1px solid black; padding: 5px; background-color: #e0ffe0;"> ┌           T ^ T V T s ⊢ "T"      ^ T V T r ⊢ S        ^ T V T s ⊢ S "^"    T V T s ⊢ S "^" "T" V T r ⊢ S                V T s ⊢ S "V"        T s ⊢ S "V" "T" r ⊢ S s ⊢ S ⊢ accept</pre>	<p style="color: red; font-weight: bold; margin: 0;">X</p> <pre style="border: 1px solid black; padding: 5px; background-color: #e0ffe0;"> ┌           T ^ T V T s ⊢ "T"      ^ T V T r ⊢ S        ^ T V T s ⊢ S "^"    T V T s ⊢ S "^" "T" V T s ⊢ S "^" "T" "V" T s ⊢ S "^" "T" "V" "T" r ⊢ S "^" "T" "V" S r ⊢ S "^" S r ⊢ S s ⊢ S ⊢ accept</pre>	<p style="color: red; font-weight: bold; margin: 0;">X</p> <pre style="border: 1px solid black; padding: 5px; background-color: #e0ffe0;"> ┌           T ^ T V T s ⊢ "T"      ^ T V T r ⊢ S        ^ T V T s ⊢ S "^"    T V T s ⊢ S "^" "T" V T r ⊢ S "^" S  V T s ⊢ S "^" S "V" T s ⊢ S "^" S "V" "T" r ⊢ S "^" S "V" S r ⊢ S "V" S r ⊢ S s ⊢ S ⊢ accept</pre>
<p style="color: green; font-weight: bold; margin: 0;">✓</p> <pre style="border: 1px solid black; padding: 5px; background-color: #e0ffe0;"> ┌           T ^ T V T s ⊢ "T"      ^ T V T r ⊢ S        ^ T V T s ⊢ S "^"    T V T s ⊢ S "^" "T" V T r ⊢ S "^" S  V T r ⊢ S                V T s ⊢ S "V"        T s ⊢ S "V" "T" r ⊢ S "V" S r ⊢ S s ⊢ S ⊢ accept</pre>	<p style="color: red; font-weight: bold; margin: 0;">X</p> <pre style="border: 1px solid black; padding: 5px; background-color: #e0ffe0;"> ┌           T ^ T V T s ⊢ "T"      ^ T V T r ⊢ S        ^ T V T s ⊢ S "^"    T V T s ⊢ S "^" "T" V T r ⊢ S "^" S  V T s ⊢ S "^" S "V" T s ⊢ S "^" S "V" "T" r ⊢ S "^" S "V" S r ⊢ S "^" S r ⊢ S s ⊢ S ⊢ accept</pre>	

## 4 À propos de ce cours

Nous nous engageons à ne pas tenir compte des renseignements ci-dessous pour noter votre copie. Ils ne sont pas anonymes, car nous sommes curieux de confronter vos réponses à votre note. En échange, quelques points seront attribués pour avoir répondu. Merci d'avance.

Répondez sur les formulaires de QCM qui vous sont remis. Vous pouvez cocher plusieurs réponses par question.

Q.22 Prises de notes

- a. Aucune
- b. Sur papier
- c. Sur ordinateur à clavier
- d. Sur ardoise
- e. Sur le journal du jour

Q.23 Travail personnel

- a. Rien
- b. Bachotage récent
- c. Relu les notes entre chaque cours
- d. Fait les annales
- e. Lu d'autres sources

Q.24 Ce cours

- a. Est incompréhensible et j'ai rapidement abandonné
- b. Est difficile à suivre mais j'essaie
- c. Est facile à suivre une fois qu'on a compris le truc
- d. Est trop élémentaire

Q.25 Ce cours

- a. Ne m'a donné aucune satisfaction
- b. N'a aucun intérêt dans ma formation
- c. Est une agréable curiosité
- d. Est nécessaire mais pas intéressant
- e. Je le recommande

Q.26 L'enseignant

- a. N'est pas pédagogue
- b. Parle à des étudiants qui sont au dessus de mon niveau
- c. Me parle
- d. Se répète vraiment trop
- e. Se contente de trop simple et devrait pousser le niveau vers le haut