

Examen « Compilation II »
Enseignant : Karim Baïna
Durée = 2H00

Seuls les documents de Cours et de TD sont autorisés !!
Le barème est donné seulement à titre indicatif !!
 Les **réponses directes** et **synthétiques** seront appréciées

Nom :

Prénom :

Exercice I : QCM 5 pts (à rendre avec votre copie !!)

Pour chaque concept/question, remplissez la case de la colonne des choix uniques correspondante par un choix qui soit le plus adéquat :

Concept/Question	Choix unique	Choix possibles
(1) Control Flow Graph	(b)	(a) démontrer qu' « une grammaire est ambiguë » est décidable mais l'inverse est non décidable
(2) bytecode J2EE		(b) Représentation.....
(3) Grammaire attribuée		(c) Analyseur Ascendant
(4) Grammaire LL		(d) Erreur Syntaxique
(5) Acorn RISC Machine-ARM		(e) Représentation.....
(6) select * from * ;		(f) Analyseur Descendant
(7) bytecode	(e)	(g) Erreur Sémantique
(8) select T1.A1 from T2 ;		(h) Erreur Lexicale
(9) Grammaire LR		(i) actions sémantiques
(10) Commentaire C non fermé (/* sans */)		(j) one-address code
(11) semi-décidabilité	(a) « RESOLUE »	(k) three-address code

Exercice II : Analyse Syntaxique / Contextuelle 10 pts

Soit la grammaire LALR G_{pcsh} du langage sous la forme BNF suivante :

```

<SCRIPTPCSH> ::= <HEAD> <INSTLIST>
<HEAD> ::= "#!/bin/pcsh" RC
<INSTLIST> ::= <INST> <INSTLISTAUX>
<INSTLISTAUX> ::= ε | RC <INSTLIST>
<INST> ::= <ECHO> | <ASSIGN> | <IF> | <FOREACH> |
<WHILE> | COMMENT
<ECHO> ::= echo <ECHOAUX>
<ECHOAUX> ::= '$'IDF | <ELT>
<ASSIGN> ::= '@' IDF '=' <EXPNUM> | "set" IDF = STRING
<IF> ::= "if" '(' <EXPBOOL> ')' then RC <INSTLIST> <ELSE> RC
"endif"
<ELSE> ::= ε | RC "else" RC <INSTLIST>
<FOREACH> ::= "foreach" IDF '(' <ELTLIST> ')' RC <INSTLIST> RC
"end"
<WHILE> ::= "while" '(' <EXPBOOL> ')' RC <INSTLIST> RC "end"
<EXPNUM> ::= NUM | '$'IDF | <EXPNUM> OPNUM NUM |
<EXPNUM> OPNUM '$'IDF
<EXPBOOL> ::= '$'IDF | <EXPBOOL> OPBOOL <EXPBOOL> | !
'(<EXPBOOL> ' )' | '$'IDF COMP <ELT>
<ELTLIST> ::= <ELT> <ELTLISTAUX>
<ELTLISTAUX> ::= ε | <ELTLIST>
<ELT> ::= STRING | NUM

```

Où respectivement la description des terminaux est la suivante : RC (retour chariot), COMMENT (commentaire c-shell sur une ligne : toute suite de caractères commençant par une #) IDF (identificateur), OPNUM (opérateur arithmétique : *, +, -, /, %), STRING (chaîne de caractères entre apostrophes ' '), OPBOOL (opérateur logique : &&, ||, !), IDFORNUM (IDF ou NUM), COMP (opérateur relationnel : ==, <=, >=, !=). On utilisera la sémantique usuelle des instructions C-SHELL.

1. Démontrer que la grammaire G_{pcsh} est Ambiguë (a) contre-exemple et (b) causes d'ambiguïté (2 pt)
2. Éliminer l'ambiguïté en se basant sur les mêmes conventions que le cours (2 pts)
3. Éliminer la récursivité gauche de la grammaire G_{pcsh} (2 pt)
4. Rendre la grammaire G_{pcsh} LL(1) (2 pts)
5. Donner trois défauts ou limitations syntaxiques de la grammaire G_{pcsh} et proposer les solutions pour ces trois défauts (2 pts)

Exercice III : Sémantique et Programmation 6 pts¹

1. Rendre la grammaire G_{pcsh} attribuée LL(1) (2 pt)
2. Programmer la grammaire attribuée LL(1) en C (2 pt)
3. Programmer la grammaire attribuée LALR en bison ? (2 pt)

¹ (dont 1 pt optionnel)