PLD
Assignment 3

3.1, 2, 3, 4, 10, 16

Kurt Medley

1

3.1 Rewrite the productions for each of the following nonterminals as right regular grammars: Identifier, Float

Identifier → Letter {Letter | Digit}
Letter → a|b|...|z|A|B|...|Z
Digit → 0|1|2|...|9

Integer → 0 Int | ... | 9 Int |
→ 0 9 Int
→ 0 9 6 Int

RGrammar

Identifier → a {0|...|9}* identifier | b {0|...|9}* identifier | ... | z {0|...|9}* identifier |
A {0|...|9}* identifier | B {0|...|9}* identifier | ... | Z {0|...|9}* identifier
aBØ2
0|...|9

→ a identifier
→ a B identifier
→ a B Ø 2 $

identifier → letter IdentRest
identifier → letter IdentRest | digit identRest

Float → Integer . Integer

RG Float → 0 Float | ... | 9 Float | . Float |
0|...|9

42.32 → 4 Float
→ 4 2 Float
→ 4 2 . Float
→ 4 2 . 3 Float
→ 4 2 . 3 2

Float → digit IntRest
IntRest → . digit DecRest | digit IntRest
DecRest → | digit DecRest

3.2 Rewrite the productions for each of the following nonterminals as left regular grammars: identifier, Float

identifier → identifier (a|...|z|A|...|Z){0|...|9} | λ
0|...|9
ab2 → identifier b 2
→ a b 2

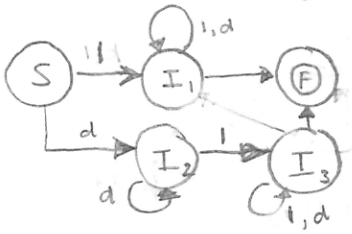
Identifier → letter | Identifier letter
Identifier digit

Float → Float 0 | ... | Float 9 | Float . | λ
0|...|9

Float → IntPart . digit | Float digit
IntPart → digit | IntPart digit

32.4 → Float 4
→ Float . 4
→ Float 2 . 4
→ Float 3 2 . 4
→ 3 2 . 4

3. Draw a DFSA for identifiers that contain only letters and digits, where the identifier must have at least one letter, but it need not be the first character



(S, kurt123$) ⊢ (I1, urt123$) ⊢ (I1, rt123$) ⊢ (I1, t123$)
⊢ (I1, 123$) ⊢ (I1, 23$) ⊢ (I1, 3$) ⊢ (I1, $) ⊢ (F,)

(S, 12a2$) ⊢ (I2, 2a2$) ⊢ (I2, a2$) ⊢ (I3, 2$) ⊢
(I3, $) ⊢ (F,)

4. Try to define the language $\{a^n b^n\}$ using a regular grammar. Discuss why this might not be possible.

n=0
L → aL | bL | abL

A regular grammar cannot compensate for all forms of this expression.

10. SciNO RGrammar

SciNo → [+|-]{01..19}.{01...|9}e{01..19} SciNo | λ

23.42|e23

Float → digit Float | .digit | intRest | Exponent
intRest → digit intRest | Exponent
Exponent → E + Expo | E - Expo | E Expo
Expo → digit Expo | digit

[0-9]*.?[0-9]*E{+1-}?[0-7]+

16. S → |a|(T)
T → T,S|S
S' → S$

First(S) = {a(}
First(T) = {a(}