CS-302
HW 4
Page-1

Görkem YAK
27970 ~~Görkem~~

1)
a) CFG = (V, T, P, S)

V = { S, E, F, I, D, X, Y, Z }

T = { +, -, *, /, (,), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, . }

S = { S)

P: $S \rightarrow E \mid S+E \mid S-E \mid -S$

$E \rightarrow F \mid E*F \mid E/F$

$F \rightarrow I \mid (S)$

$I \rightarrow D \mid D.Z$

$D \rightarrow Y \mid XZ$

$X \rightarrow 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

$Y \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

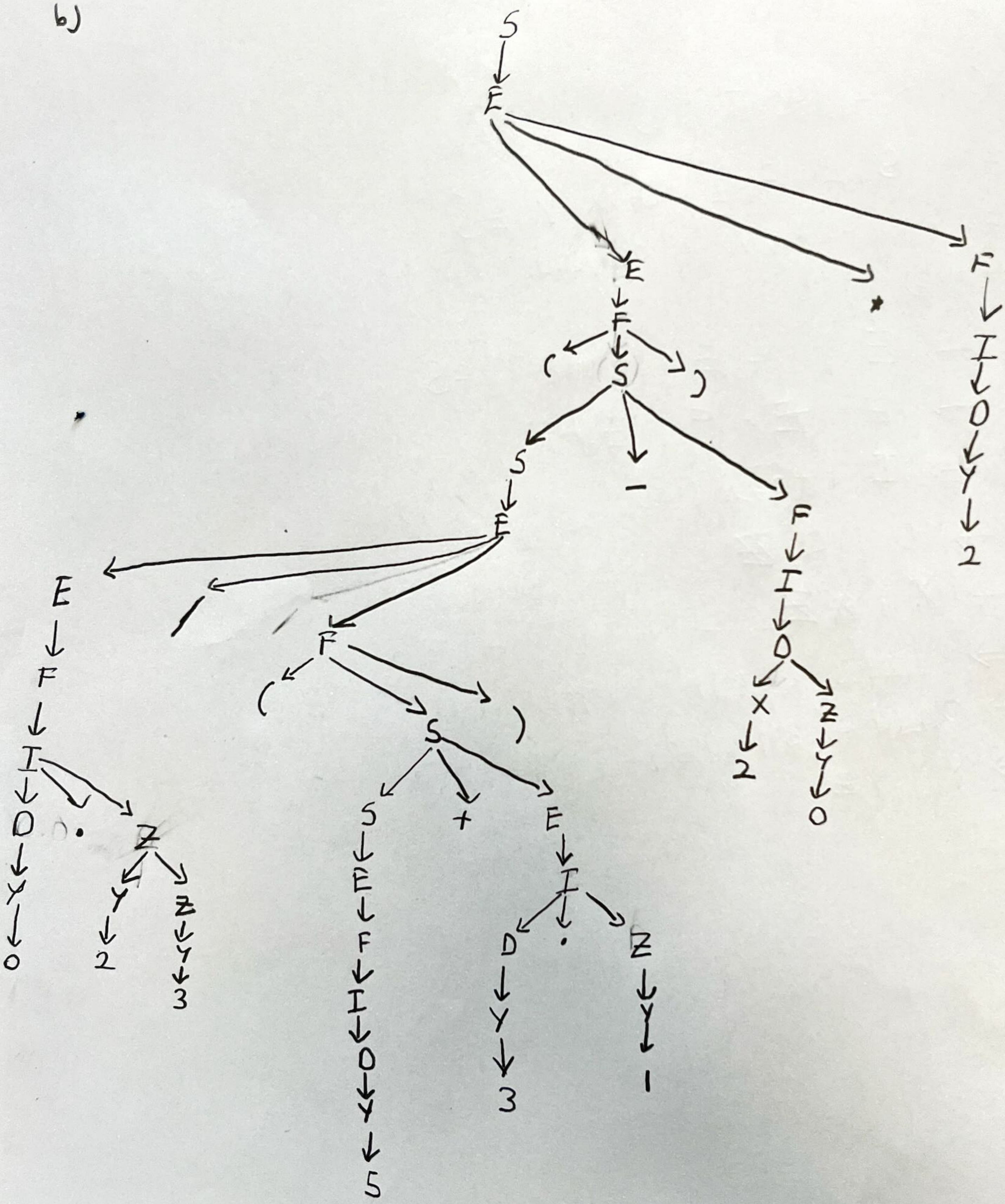
$Z \rightarrow Y \mid YZ$

CS-302 HW4
Page 2

Gorkem YAR
27970

Question -1

b)



Görkem YAK
27970 ~~Görkem~~

Question - 2

$$CFG = (V, T, P, S)$$

$V = \{S, \lambda\}$

$T = \{x, y, z, T, F\}$

$S = \{S\}$

This to break ambiguity..

$P: S \to (S) \mid \neg S \mid \lambda \vee S \mid \lambda \wedge S \mid \lambda$

$P: \lambda \to x \mid y \mid z \mid T \mid F$

To derive $E = x \vee y \wedge (T \wedge \neg z)$

$S \to \lambda \vee S \to x \vee S \to x \vee \lambda \wedge S \to$
$x \vee y \wedge S \to x \vee y \wedge (S) \to$
$x \vee y \wedge (\lambda \wedge S) \to x \vee y \wedge (T \wedge S) \to$
$x \vee y \wedge (T \wedge \neg S) \to x \vee y \wedge (T \wedge \neg \lambda)$
$\to x \vee y \wedge (T \wedge \neg z)$

Gökten YAR
27970

Question-3

Exercise 5.1.2

b) leftmost

$S \xrightarrow[lm]{} A1B \xrightarrow[lm]{} 1B \xrightarrow[lm]{} 10B \xrightarrow[lm]{} 100B \xrightarrow[lm]{} 1001B \xrightarrow[lm]{} 1001$

rightmost

$S \xrightarrow[rm]{} A1B \xrightarrow[rm]{} A10B \xrightarrow[rm]{} A100B \xrightarrow[rm]{} A1001B \xrightarrow[rm]{} A1001 \xrightarrow[rm]{} 1001$

c) leftmost

$S \xrightarrow[lm]{} A1B \xrightarrow[lm]{} 0A1B \xrightarrow[lm]{} 00A1B \xrightarrow[lm]{} 000A1B \xrightarrow[lm]{} 0001B$
$\xrightarrow[lm]{} 00011B \xrightarrow[lm]{} 00011$

rightmost

$S \xrightarrow[rm]{} A1B \xrightarrow[rm]{} A11B \xrightarrow[rm]{} A11 \xrightarrow[rm]{} 0A11 \xrightarrow[rm]{} 00A11 \xrightarrow[rm]{} 000A11$
$\xrightarrow[rm]{} 00011$

Gorkem YAR
27970
Gorkem

Problem 3:

Exercise 5.1.3: By definition a regular language can be represented by an DFA or NFA. We know that we can get Regular expressions from DFA's using main theorem or state elimination. So every regular language can be described by a regular expression. We need to show that we can describe every regular expressions using CFG.

Regular Expressions (E) can do the following operations if $\Sigma = \{6_1, 6_2 \dots 6_n\}$

$$E \rightarrow 6_1 \mid 6_2 \mid \dots \mid 6_n \mid e \mid \phi \mid E+E \mid E.E \mid E^* \mid (E)$$

Now, we need to define a CFG that can do these operations.

$G = (V, T, P, S)$

$V = \{E\}$

$T = \Sigma \cup \{e, \phi, +, ., *, (,)\}$

P = the $n+6$ productions above

$S = E$

Assumption

This CFG can accept all regex.

Proof by Induction:

Base case: Single character in the Σ

Inductive step: For any regex with $< n$ operators and CFG we can construct a CFG s.t. it accepts regex with n operators.

In our language + (union), . (concatenation), * (star) operations.

$R_1 + R_2$ — $n-1$, $n-1$ operators

$R_1 . R_2$ — $n-1$ operators

R_1^* — $n-1$ operators

These production rules defined above. With them we can define any regex with more than n operators.

Thus every RE can be described by a CFG. So, every Regular language is a context free language.

Gorkem YAR
27910

Problem : 3

Exercise 5.1.4

a) if a CFG is "right linear" every production rule can consist only one non-terminal variable in the very right end.

For instance $A \rightarrow wB \mid w$

We need to show that the language $L(G)$ generated by $\underbrace{G(V,T,P,S)}_{CFG}$ is regular. We will use e-NFA to show this

Let's say N_e is an ϵ-NFA s.t $N_e = (Q, \Sigma, \delta_{Ne}, Q_0, F)$.
We can build N_e with the following properties.

$Q = V \cup \{F\}$ There should be exactly one final

$\Sigma = w_i$ set of terminal symbols

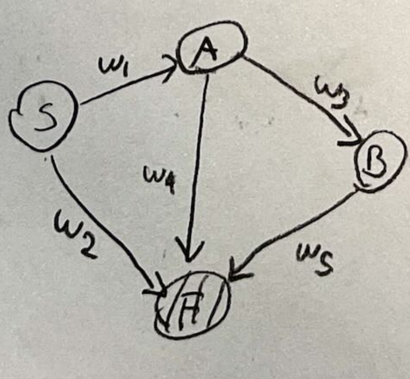
$\delta_{NE} : Q \times (\Sigma \cup \epsilon) \rightarrow 2^Q$

Starts variable in CFG is the start state in N_e.

For each production $A \rightarrow wb$ we add a transition from state A to state B via input w. W can be e which lead to an ϵ transition.

For each transition from $A \rightarrow w$, we add a transition to final state F with input w. no non terminal

$$\left.\begin{aligned} S &\rightarrow w_1 A \mid w_2 \\ A &\rightarrow w_3 B \mid w_4 \\ B &\rightarrow w_5 \end{aligned}\right\}$$



CS-302
Homework-4
Page-7

Gördem YAR
27970 Gördem

5.1.4

b) if a languge regular we can assume that there is a DFA that accepts this language.

DFA $D = (Q, \Sigma, \delta, q_0, F)$

We can create right linear grammar $G(V, T, P, S)$

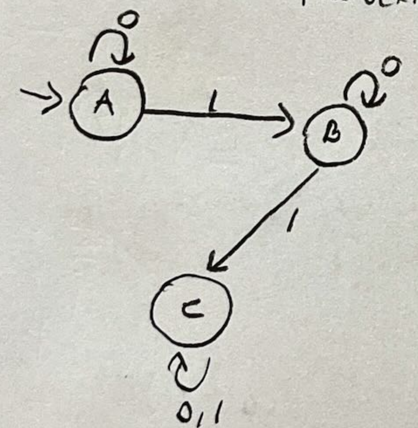
$V = Q - F$ — Q_i that not in F correspond to a non terminal Variable in G.

$T = \Sigma$

$A_0 = Q_0$

For each transition $\delta(q_i, \sigma) = q_j$ if $q_j \notin F$
add a production rule s.t. $V_i \rightarrow a V_j$ in G

if the transition $\delta(q_i \sigma) = q_j \in F$
add a production rule $V_i \rightarrow \sigma$



$A \rightarrow 0A \mid 1B$

$B \rightarrow 0B \mid 1C$

$C \rightarrow 0C \mid 1C \mid 0 \mid 1$

Görkem YAR
27970

Problem 3.

Exercise 5.1.7

a)

Assumption : There is no "ba" as a substring in any string generated by $L(G)$

Proof by Induction:

Base step: string 'a' or 'b' does not include 'ba'

Inductive step: Assume that every string that length $\leq k$ does not include 'ba' as a substring. Prove that the string with $k+1$ length cannot include 'ba' as a substring.

We can interpret the string with $k+1$ element in two different ways.

$w' = aw$

len k that is an accepted string

adding an 'a' in the beginning cannot form 'ba'

therefore $w' \in L(G)$

$w' = wb$

len k that is an accepted string.

Adding a 'b' at the end cannot form 'ba'

therefore $w' \in L(G)$

By the induction Hypothesis, the string w' has no 'ba' as substring. Therefore, no string in $L(G)$ contain "ba" as substring.

b) $L(G)$ is the language generated by CFG G.

The set of strings of $\{a, b\}$ such that all of the b's should come after all of the a's. Since there is no 'ba' in the language.