CSci 435: Formal Languages and Automata

Name: Adam Roy

**Home Assignment 4: 55 points + 10 points (optional)**

Q1. [15] For a given language L = {*a2nb****n*** | *n* ≥ 0 is even}.

1. [5] Give a CFG that accepts L.

G = {(S), (a, b), S, (S -> aaSb | ε )}

1. [5] Show the sequence of derivations for the acceptance of *aaaabb* by G in (1).

S -> apply rule S -> aaSb to S

aaSb -> apply rule S -> aaSb to S

aaaaSbb -> apply rule S -> ε to S

aaaaεbb = aaaabb

1. [5] Draw a derivation tree for *aaaabb*.

A diagram of a network

Description automatically generated

Q2. [10] Construct a CFG for the following languages where *n*, *m, k* ≥ 0.

1. [10] L = { *anbm* | *n =* *m –*1, *m* ≥ 1 }

G = {(S, A), (a, b), S, (S -> Ab, A -> aAb | ε )}

1. [10, optional] L2 = { *anbmck* | *n=m* or *m* ≤ *k* }  
   G = {(S, A), (a, b, c), S, (S -> AB | BC, A -> aA | ε, B -> bB | ε, C -> cC | ε)}

Q3. [10] Find an s-grammar for L = {*aaa*\**b* + *ab*\*}.

G = {(S, A, B, C), (a, b), S, (S -> aAb | B, A -> aA | ε, B -> aB | bC, C -> bC | ε)}

Q4. [20] For a grammar G with the productions where G = ( {S, A, B}, {*a, b*}, S, P ) with productions

S → AB | *bbbB*, A → *b* | A*b*, B → *a..*

1. [6] Show that the grammar G is ambiguous.

A diagram of different types of arrows

Description automatically generated

Since there are two parse trees that produce the same string ‘bbba’ I can determine that the grammar is ambiguous.

1. [7] Give language L that is generated by G, L = L(G), in a formal expression (including a regular expression).

L(G) = (b+ + bbb)a

1. [7] Can you construct an unambiguous grammar that is equivalent to G? Otherwise, show that G is inherently ambiguous.

If non-terminal A is swapped into S and the A production is removed all ambiguity of the grammar is removed.

S -> {b | Ab} | bbbB = b+ | bbbB

B -> a

This creates the production rules:

S -> b+ | bbbB

B -> a

This can only create sets of strings with no ambiguity