COMP2805: Solution to assignment # 4

December 2, 2010

Question 1

Prove, using the pumping lemma for CFL, that the following are not context-free:

Solution:

- $L = \{a^N b^{2N} a^{2N} b^N\}$

  Assume that L is context free. Thus, for any word of length greater than or equal to the pumping length p, it can be written as $uvxyz$, where $v \neq \epsilon$, $y \neq \epsilon$, and $|vxy| \leq p$ such that $uv^ixy^iz \in L$ for all $i \geq 1$.

  Consider the word $a^p b^{2p} a^{2p} b^p \in L$. This word clearly has length greater than the pumping length, and thus can be pumped. However, since $|vxy| \leq p$, there are a limited number of cases for where in the string v and y can be. Clearly, if both v and y consist only of the character a or b, when pumped the ratio of a to b will not be consistent with the definition of the language. For the other cases, consider $i = 2$. Three possible cases emerge, depending on where v and y are located in the string: $a^{p+i} b^{2p+i} a^2 p b^p$, $a^p b^{2p+i} a^{2+i} p b^p$, and $a^p b^2 p a^2 p + i b^{p+i}$. Clearly, none of these words are in L, and thus, by contradiction, L is not context-free.

- $L = \{a^N b^{2N} c^{3N}\}$

  By the same reasoning as above, any word in L with length greater than p can be pumped. We consider the word $a^p b^{2p} c^{3p}$.

  As $|vxy| \leq p$, only two of the three characters in the word can appear in vxy, thus, when $k \geq 2$, The ratio between the three characters cannot hold, and thus L is not context-free.

Question 2

Construct (*deterministic or non-deterministic*) pushdown automata that accept the following language:

Solution:

1. $L = \{w | w \in \{0;1\}^*$ *w contains more $0s$ than $1s$* $\}$.

$q0\$ \rightarrow qR\$0$	word starts with a 0; push 0
$q1\$ \rightarrow qR\$1$	word starts with a 1; push 1
$q\square\$ \rightarrow qN\epsilon$	word is accepted and the stack is empty
$q00 \rightarrow qR00$	read 0 push 0 if stack has 0
$q10 \rightarrow qR\epsilon$	read 1 pop 0
$q\square 0 \rightarrow qN\epsilon$	word's finished but need to clear the 0s
$q01 \rightarrow qR\epsilon$	read 0 pop 1
$q11 \rightarrow qR11$	read 1 push 1 if stack has 1
$q\square 1 \rightarrow qN1$	infinite loop 1

2. L=$\{a^N b^{2N} c^M | N, M \geq 0\}$ where $\Sigma = \{a, b, c\}$

$q_1\square\$ \rightarrow q_1N\epsilon$	word is accepted and the word is ϵ
$q_1a\$ \rightarrow q_2NS$	push a S on the stack and moves to q_2
$q_1aS \rightarrow q_2NSS$	push a S on the stack and moves to q_2
$q_1bS \rightarrow q_3R\epsilon$	pop S and moves to q_3
$q_1c\$ \rightarrow q_4RT$	push a T on the stack and moves to q_4
$q_2aS \rightarrow q_1RSS$	push a S on the stack and returns to q_1
$q_3bS \rightarrow q_3R\epsilon$	pop S
$q_3\square\$ \rightarrow q_3N\epsilon$	word is accepted and the stack is empy
$q_3c\$ \rightarrow q_4RT$	push a T on the stack and moves to q_4
$q_4cT \rightarrow q_4RTT$	push a T on the stack
$q_4\square T \rightarrow q_4N\epsilon$	pop T
$q_4\square\$ \rightarrow q_4N\epsilon$	word is accepted and the stack is empy
empty	

3. L=$\{ww^R | w \in \{0;1\}^*, and\ where\ w^R\ is\ the\ reverse\ of\ w\}$.

$q\square\$ \rightarrow qN\epsilon$	word is accepted and the stack is empy
$q0\$ \rightarrow qR0$	0 is read as first symbol, push a 0 on the stack
$q1\$ \rightarrow qR1$	1 is read as first symbol, push a 1 on the stack
$q00 \rightarrow qR00$	0 is read in first half of word, push a 0 on the stack
$q01 \rightarrow qR10$	0 is read in first half of word, push a 0 on the stack
$q00 \rightarrow q'R\epsilon$	read a 0 and second half is reached, pop 0 and move to state q'
$q10 \rightarrow qR01$	1 is read in first half of word, push a 1 on the stack
$q11 \rightarrow qR11$	1 is read in first half of word, push a 1 on the stack
$q11 \rightarrow q'R\epsilon$	read a 1 and second half is reached, pop 1 and move to state q'
$q'00 \rightarrow q'R\epsilon$	0 is read on the second half, pop 0
$q'11 \rightarrow q'R\epsilon$	1 is read on the second half, pop 1
$q'\square\$ \rightarrow q'N\epsilon$	word is accepted and the stack is empty

Question 3

Construct a Turing Machine with one tape that accept $\{a^N b^{2N} : n \geq 0\}$**.**

Solution:

The Turing machine will do the following:

Stage 1: Delete the leftmost symbol (if it is an a).
Stage 2: Walk to the rightmost position and delete the last two symbols (if they are b's).

Stage 3: Walk to the leftmost symbol.
Repeat stages 1-3 until the string is empty. If nothing strange happens, accept; otherwise, reject.

We use the following states:

- q_0: start state.
- q_1: leftmost symbol was an a; it has been deleted; we walk to the rightmost symbol.
- q_2: we are at the rightmost symbol.
- q_3: rightmost symbol was a b; it has been deleted.
- q_4: second symbol from the right was a b; it has been deleted; we walk back to the leftmost symbol.
- q_{accept}
- q_{reject}

Here are the instructions:
$q_0 a \rightarrow q_1 \Box R$
$q_0 b \rightarrow q_{reject}$
$q_0 \Box \rightarrow q_{accept}$

$q_1 a \rightarrow q_1 a R$
$q_1 b \rightarrow q_1 b R$
$q_1 \Box \rightarrow q_2 \Box L$

$q_2 a \rightarrow q_{reject}$
$q_2 b \rightarrow q_3 \Box L$
$q_2 \Box \rightarrow q_{reject}$

$q_3 a \rightarrow q_{reject}$
$q_3 b \rightarrow q_4 \Box L$
$q_3 \Box \rightarrow q_{reject}$

$q_4 a \rightarrow q_4 a L$
$q_4 b \rightarrow q_4 b L$
$q_4 \Box \rightarrow q_0 \Box R$

Question 4

Construct a one-tape Turing machine that accepts the language $L = \{w | w$ has twice the number of 1s than 0s$\}$. You have to give the informal description, describing the states you are going to use, their meaning and what kind of transitions it will have. This will give you partial credit. But you have to also work out all the details of the actual transitions. You can assume that the head of the machine is at the start of the string.

Solution:

The Turing Machine will do the following:

Step 1: Walk until it finds a 0. If it finds no characters, or only the special character \$ $thestringisempty$ accept. If it finds a 1, shift to a state that will not accept until the 1's are cancelled out.
Step 2: Replace the 0 with a special character \$. Return to the start of the string.
Step 3: Walk until it has found and replaced two 1's with the special character \$.
Repeat steps 1-3 until the string is accepted or the end of the string is reached at a non-accepting state, in which case it is rejected.

The following states are used:

- q_0: Start state
- q_1: Analogous to the start state, but does not accept if the end of the string is reached. Reached by encountering a 1 while looking for a 0
- q_2: Backtrack state, used to backtrack to the start of the string when a 0 is found.
- q_3: The machine is looking for the first 1
- q_4: The machine is looking for the second 1
- q_5: Backtrack state, used to backtrack to the start of the string when two 1s have been found and replaced to cancel out a 0
- q_{accept}: The accept state
- q_{reject}: The reject state

The instructions are as follows:

$q_0 0 \rightarrow q_2 \$ L$
$q_0 1 \rightarrow q_1 1 R$
$q_0 \square \rightarrow q_{accept}$
$q_0 \$ \rightarrow q_0 \$ R$

$q_1 0 \rightarrow q_2 \$ L$
$q_1 1 \rightarrow q_1 1 R$
$q_1 \square \rightarrow q_{reject}$
$q_1 \$ \rightarrow q_2 \$ R$

$q_2 0 \rightarrow q_2 0 L$
$q_2 1 \rightarrow q_2 1 L$
$q_2 \square \rightarrow q_3 \square R$
$q_2 \$ \rightarrow q_2 \$ L$

$q_3 0 \rightarrow q_3 0 R$
$q_3 1 \rightarrow q_4 \$ R$
$q_3 \Box \rightarrow q_{reject}$
$q_3 \$ \rightarrow q_3 \$ R$

$q_4 0 \rightarrow q_4 0 R$
$q_4 1 \rightarrow q_5 \$ L$
$q_4 \Box \rightarrow q_{reject}$
$q_4 \$ \rightarrow q_4 \$ R$

$q_5 0 \rightarrow q_5 0 L$
$q_5 1 \rightarrow q_5 1 L$
$q_5 \Box \rightarrow q_0 \Box R$
$q_5 \$ \rightarrow q_5 \$ L$

Question 5

Construct a two-tape Turing machine that accepts the language $L = \{a^N b^{2N} | n \geq 0\}$.

Solution:

The Turing machine will do the following:

1. Copy the a's on tape 2 and erase them from tape 1;
2. For each a read and erased on tape 2, the machine read and erase two b's on tape 1.
3. The machine accepts the input if and only if at the end of step 2, both tapes are empty.

We use the following states:

state	description
q_s	start state
q_a	step 1 described above
q_{1b}	read the first b for step 2 described above
q_{2b}	read the second b for step 2 described above
q_{accept}	the machine accepts the input
q_{reject}	the machine rejects the input

Here are the instructions:

instruction	description
$q_s\square\square \to q_{accept}\square\square NN$	**accept**, the input is empty
$q_s a\square \to q_a a\square NN$	move to q_a
$q_s b\square \to q_{reject}\square\square NN$	reject, start with a b
$q_a\square\square \to q_{reject}\square\square NN$	reject, no b to read
$q_a a\square \to q_a\square aRR$	copy an a to tape 2 and erase it from tape 1
$q_a b\square \to q_{1b} b\square NL$	tape 1 on the first b, tape 2 move left to rightmost a
$q_{1b}\square\square \to q_{accept}\square\square NN$	**accept**, the input is valid and both tapes are empty
$q_{qb}\square a \to q_{reject}\square\square NN$	reject, no more b's to read on tape 1
$q_{1b} a\square \to q_{reject}\square\square NN$	reject, read an a on tape 1
$q_{1b} aa \to q_{reject}\square\square NN$	reject, read an a on tape 1
$q_{1b} b\square \to q_{reject}\square\square NN$	reject, no more a on tape 2
$q_{1b} ba \to q_{2b}\square aRN$	erase the first b and prepare to read the second one
$q_{2b}\square a \to q_{reject}\square\square NN$	reject, needs two b's to clear on a
$q_{2b} aa \to q_{reject}\square\square NN$	reject, read an a on tape 1
$q_{2b} ba \to q_{1b}\square\square RL$	erase the second b, erase the corresponding a and go back to q_{1b}

NOTE: Unmention transition such as $q_s\square a \to$ or $q_a\square a \to$ are simply unreachable.

Here's a trace for the input $aabbbb$, where the character between parenthesis is currently the one checked by the machine:

state	tape 1	tape 2	transition to next step
q_s	$\square(a)abbbb\square$	$\square(\square)\square\square\square\square\square\square$	$q_s a\square \to q_a a\square NN$
q_a	$\square(a)abbbb\square$	$\square(\square)\square\square\square\square\square\square$	$q_a a\square \to q_a\square aRR$
q_a	$\square\square(a)bbbb\square$	$\square a(\square)\square\square\square\square\square$	$q_a a\square \to q_a\square aRR$
q_a	$\square\square\square(b)bbb\square$	$\square aa(\square)\square\square\square\square$	$q_a b\square \to q_{1b} b\square NL$
q_{1b}	$\square\square\square(b)bbb\square$	$\square a(a)\square\square\square\square\square$	$q_{1b} ba \to q_{2b}\square aRN$
q_{2b}	$\square\square\square\square(b)bb\square$	$\square a(a)\square\square\square\square\square$	$q_{2b} ba \to q_{1b}\square\square RL$
q_{1b}	$\square\square\square\square\square(b)b\square$	$\square(a)\square\square\square\square\square\square$	$q_{1b} ba \to q_{2b}\square aRN$
q_{2b}	$\square\square\square\square\square\square(b)\square$	$\square(a)\square\square\square\square\square\square$	$q_{2b} ba \to q_{1b}\square\square RL$
q_{1b}	$\square\square\square\square\square\square\square(\square)$	$(\square)\square\square\square\square\square\square\square$	$q_{1b}\square\square \to q_{accept}\square\square NN$
q_{accept}	$\square\square\square\square\square\square\square(\square)$	$(\square)\square\square\square\square\square\square\square$	input **accepted**