Anand Mahadevan - SID 862132182 - Disc: 022

CS150 Homework 3

due Wednesday, November 3rd 5:00 PM

Problem 1.
Prove that these aren't regular languages using the Pumping lemma.

a) Let L be the set of strings of balanced parentheses. These are the strings of characters "(" and ")" that can appear in a well formed arithmetic expression.

ANS: Suppose that L is regular.
Pick w = $(^n\,)^n$, which is n opening parentheses followed by n closing parentheses.
Consider xyz = w | $|y| > 0\, and\, |xy| \le n$ so y must be a string of "("s
, and x = $(^i$, y = $(^j$ where j $\geq$ 1 and i + j = n, and z = $)^n$.
Let's pump y with k = 0 so w = $(^i\,)^n \notin$ L because $i < n$, since $j \geq 1$. So L cannot be regular. Q.E.D.

b) L : $\{0^n 1^{2n} \quad | \quad n \geq 1\}$

ANS: Suppose that L is regular.
Pick w = $0^n\,1^{2n}$, which is n zeroes followed by 2n ones.
Consider xyz = w | $|y| > 0\, and\, |xy| \le n$ so y must be a string of "0"s
, and x = 0^i, y = 0^j where j $\geq$ 1 and i + j = n, and z = 1^{2n}.
Let's pump y with k = 0 so w = $0^i\,1^{2n} \notin$ L because the number of zeroes has decreased while the number of ones have not since $i < n$, since $j \geq 1$. So L cannot be regular. Q.E.D.

Problem 2.

Prove that this isn't a regular language using the Pumping lemma.

a) $\{0^n \quad | \quad$ n is a power of 2$\}$

ANS: Suppose that L is regular.

Pick w = 0^{2^n}, which is 2^n zeroes.

Consider xyz = w | $\quad |y| > 0\, and\, |xy| \leq n$ so y must be a string of "0"s with length between 1 to n zeroes.

Let's pump y with k = 2 so the length of w would be $2^n + 1$ to $2^n + n$.

Because the next power of 2 after 2^n would be 2^{n+1}, I know that the length of w would be strictly between 2^n and 2^{n+1} because $2^n + n$ = O(2^{n+1}).

However, this is a contradiction as w can never have a power of 2 zeroes. w $\notin$ L. Q.E.D.

Problem 3.
Which are true?
a) $a(a \setminus L) = L$ ... (1)

No, this is not true.
Counter-example: L = {b}
a $\setminus$ L = ϕ because there are no strings in L that start with the symbol a.
Plugging into (1)...
a $\cdot$ ϕ = L
ϕ = L
This is a contradiction as L = {b}
Q.E.D.

b) $(La)/a = L$ (See 4.2.2 for /)
Yes, this is true.
La represents set of all strings in L appended with an a.
The quotient of this set La with the symbol a results in a set of strings that if appended with the symbol a would result in the set La.
That is clearly the set L, which matches the identity.
Q.E.D.

Problem 4.
Suppose L is a regular language with alphabet Σ.
Give an algorithm to tell whether $L = \Sigma^*$, i.e. all strings over its alphabet.

Strategy: Minimize a DFA by using the TF algorithm and merging equivalent states, if the DFA then only contains 1 state and it is an accepting state, then the language L accepts all strings over its alphabet.
Algorithm:
1) Convert L into a DFA. (Any regular language can be converted into a DFA)
2) Run the Table-Filling algorithm on this DFA, and merge equivalent states for minimization.
3) If the resulting DFA has exactly 1 state, and that state is an accepting state, then $L = \Sigma^*$. L $!= \Sigma^*$ otherwise.

Problem 5.

	0	1
$\rightarrow A$	B	E
B	C	F
$*C$	D	H
D	E	H
E	F	I
$*F$	G	B
G	H	B
H	I	C
$*I$	A	E

a) Draw the table of distinguishablities for this automaton.

	A	B	C	D	E	F	G	H
B	X							
C	X	X						
D		X	X					
E	X		X	X				
F	X	X		X	X			
G		X	X		X	X		
H	X		X	X		X	X	
I	X	X		X	X		X	X

b) Construct the minimum-state equivalent DFA

Minimum DFA		
	0	1
→{A,D,G}	{B,E,H}	{B,E,H}
{B,E,H}	{C,F,I}	{C,F,I}
∗{C,F,I}	{A,D,G}	{B,E,H}