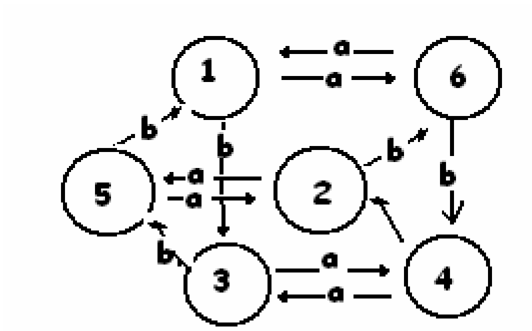
Homework #5

1. True or False:

a) Regular Languages are always Context-Free Languages   True   False
b) Context Free Languages are always Regular Languages   True   False
c) The grammar S→ 0S |0S1S |ε is ambiguous   True   False
d) The language $\{a^n b^n c^n\}$ is regular   True   False
e) The language $\{a^n b^n c^n\}$ is context-free   True   False

2. Minimize the following dfa:



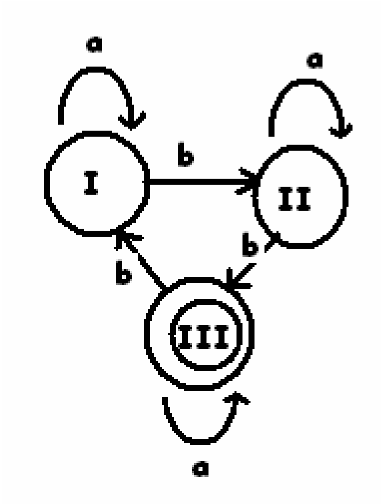
(1) Dividing into Final and Non-Final:

	a	b
1	6	3
2	5	6
5	2	1
6	1	4
3	4	5
4	3	2

(2) In Partition 1, all states "do the same thing" on an *a*. But on a *b*, states 1 and 6 both go to Partition II. We'll move them to their own partition:

	a	b
2	5	6
5	2	1
1	**6**	**3**
6	**1**	**4**
3	4	5
4	3	2

(3) We cannot partition further:

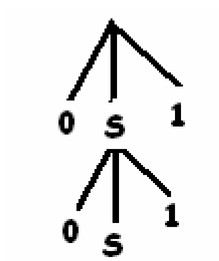


L(M) = (a*ba*ba*)*

#3. a) Create a grammar that generates the set of all strings over {0,1} with an equal number of 0's and 1's Also b) construct a parse tree and c) leftmost derivation of *0011*. d) Is your grammar ambiguous? Why or why not?

a) S→0 S 1, S→1 S 0, S→S S, S→ε

b)



b) c) Yes. There is more than 1 parse tree for ε as well as other strings.

#4. Find the Start symbol for the Java grammar shown at:
http://www.cse.psu.edu/~saraswat/cg428/lecture_notes/LJava2.html

The start symbol is CompilationUnit. It doesn't appear on the left-hand-side. It is good technique to write a programming language grammar so that the Start symbol does not occur on the right-hand-side, and all grammars can be changed to an equivalent grammar having this property (how?)

#5. For the grammar G:

S → X Z Z X
X → **x**
X → ε
Z → **z**
Z → ε

a) What is L(G)?

L(G) is finite so we can just list its strings: { ε, x, z, xz, xx, zz, zx, xzz, xzx. zzx, xzzx }.

Proof:

Let X = { ε, x, z, xz, xx, zz, zx, xzz, xzx. zzx, xzzx }. To show X = L(G) requires 2 proof parts:

1. if w ε X, then w ε L(G)
2. if w ε L(G), then w ε X

1. Given: w ε X
   Prove: w ε L(G)

   To show w ε L(G) means we have to show $S \overset{*}{\rightarrow} w$

   Since the language is finite, we can show this for each string:

   w = ε:
   S⇒XZZX⇒εZZX⇒εεZX⇒εεεX⇒εεεε = ε

   w = x:
   S⇒XZZX⇒xZZX⇒xεZX⇒xεεX⇒xεεε=x

   w = xz:
   S⇒XZZX⇒xZZX⇒xzZX⇒xzεX⇒xzεε=xz

   w = xzz:
   S⇒XZZX⇒xZZX⇒xzZX⇒xzzX⇒xzzε=xzz

   w = xzx:
   S⇒XZZX⇒xZZX⇒xzZX⇒xzX⇒xzx

   w = xzzx:
   S⇒XZZX⇒xZZX⇒xzZX⇒xzzX⇒xzzx=xzzx

   w = xx:
   S⇒XZZX⇒xZZX⇒xεZX⇒xεεX⇒xεεx=xx

   w = z:
   S⇒XZZX⇒εZZX⇒εzZX⇒εzεX⇒εzεε=z

   w = zz:

S⇒XZZX⇒εZZX⇒εzZX⇒εzzX⇒εzzε=zz

w = zx:

S⇒XZZX⇒εZZX⇒εzZX⇒εzεX⇒εzεx=zx

w = zzx:

S⇒XZZX⇒εZZX⇒εzZX⇒εzzX⇒εzzx=zzx

2. if w ε L(G), then w ε X

<u>Derivations of strings of length 0 in L(G):</u>
S⇒XZZX⇒εZZX⇒εεZX⇒εεεX⇒εεεε = ε
and ε is in X

<u>Derivations of strings of length 1 in L(G):</u>

S⇒XZZX⇒xZZX⇒xεZX⇒xεεX⇒xεεε=x (can be derived another way also)
And x is in X

S⇒XZZX⇒εZZX⇒εzZX⇒εzεX⇒εzεε=z (can be derived another way also
And z is in X

No other derivations result in strings of length 1

<u>Derivations of strings of length 2 in L(G):</u>

S⇒XZZX⇒xZZX⇒xzZX⇒xzεX⇒xzεε=xz
S⇒XZZX⇒xZZX⇒xεZX⇒xεεX⇒xεεx=xx
S⇒XZZX⇒εZZX⇒εzZX⇒εzzX⇒εzzε=zz
S⇒XZZX⇒εZZX⇒εzZX⇒εzεX⇒εzεx=zx

<u>Derivations of strings of length 3 in L(G):</u>

S⇒XZZX⇒xZZX⇒xzZX⇒xzzX⇒xzzε=xzz
S⇒XZZX⇒εZZX⇒εzZX⇒εzzX⇒εzzx=zzx

<u>Derivations of strings of length 4 in L(G):</u>
S⇒XZZX⇒xZZX⇒xzZX⇒xzzX⇒xzzx=xzzx

Then we'd have to argue that these are all the possible derivations in G.

I think a slightly better proof here might have been to show all the leftmost derivations and show that each results in a string in X.

Now we can assert that L(G) = { ε, x, xz, xzz, xzzx, xx, z, zz, zx, zzx }