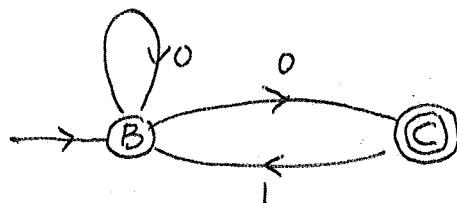


## TEST #2 - Fall 2011

TIME: 75 min.

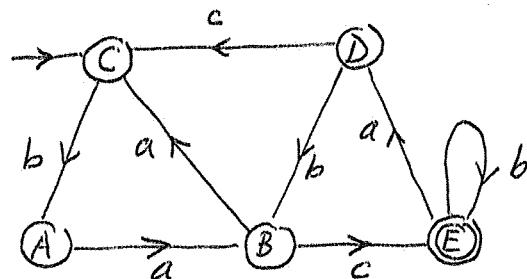
Answer all 6 questions. Provide all reasoning and show all working. An unjustified answer will receive little or no credit. Begin each of the 6 questions on a separate page.

- (15) 1. Let  $L$  be the language accepted by the NFA shown on the right.  
Find NFAs which accept  
(a)  $L^c$       (b)  $(L^c)^R$ .



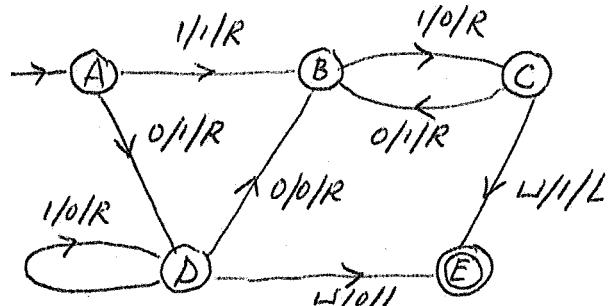
- (15) 2. (a) Find an NFA which is equivalent to the RLG given below.  
G:  $\rightarrow A, A \rightarrow 0A, A \rightarrow 0B, B \rightarrow 01, B \rightarrow 1C, B \rightarrow \lambda, C \rightarrow 0E, C \rightarrow D, D \rightarrow 01D, D \rightarrow 10, D \rightarrow \lambda, E \rightarrow 1A$ .  
(b) Convert the NFA shown below on the right in Qu.#3 into an equivalent RLG.

- (15) 3 (a) Define what is the **Busy beaver function**.  
(b) Find a regular expression for the language accepted by the NFA shown on the right.



- (22) 4 (a) Define what it means for  $f$  to be obtained from  $g$  and  $h$  by **primitive recursion**. Show that  $f(x,y) = 2x+3y+1$  is a primitive recursive function by finding primitive recursive functions  $g$  and  $h$  such that  $f = \text{prim.rec.}(g,h)$   
(b) Define what it means for  $f$  to be obtained from  $g$  by **minimization**. Show that  $f(x) = \sqrt{x+2}$  is a recursive function by finding a primitive recursive function  $g$  such that  $f = \mu[g, 0]$ . [You may use the fact that MONUS, ADD, & MULT are prim. rec. if needed in 4(b), but you can't do so in 4(a).]

- (15) 5 (a) Define what is a Turing-acceptable language on the alphabet  $V$ .  
(b) Show what happens at each step if 00101 is the input for the TM,  $M$  shown on the right.  
(c) Find the language accepted by  $M$ .



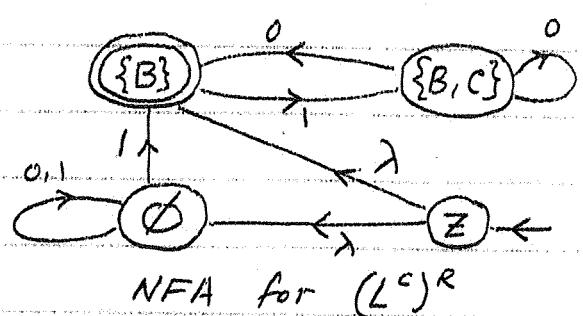
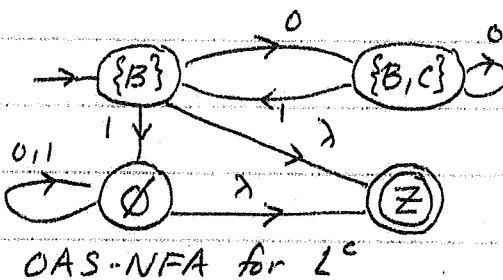
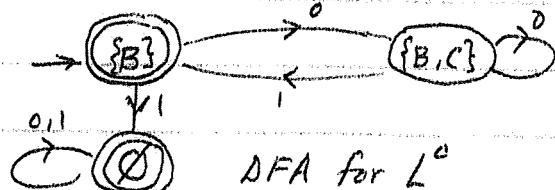
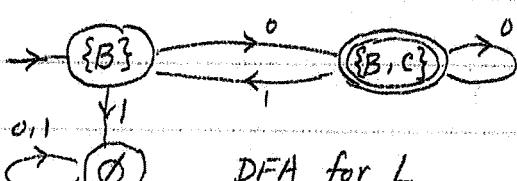
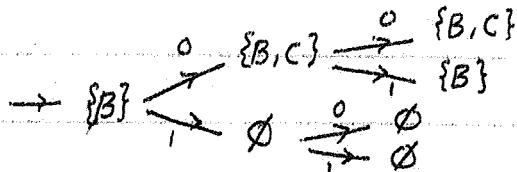
- (18) 6. Determine which of the following languages are regular and which are not.  
(a)  $L_1 = \{a^k.b^n : k \pmod 3 < 2n \pmod 3\}$       (b)  $L_2 = \{b^k.c^n : 2k < n\}$ .

[If you say that it is regular, you must find a regular expression; if you say it is non-regular, you must give a complete proof.]

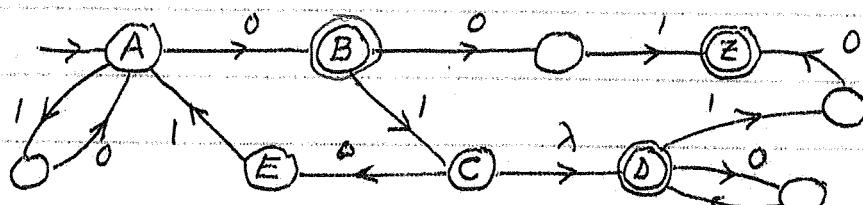
## Solutions to Test # 2

Fall 2011

1.



2(a)



(b)

$\rightarrow C, C \rightarrow bA, A \rightarrow aB, B \rightarrow aC, B \rightarrow cE, E \rightarrow bE, E \rightarrow \lambda, E \rightarrow aD, D \rightarrow cC, D \rightarrow bB$ .

3(a)

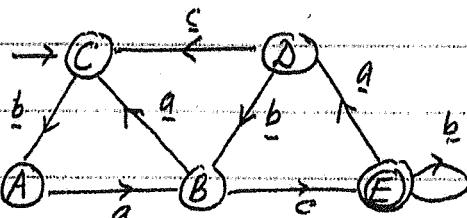
The Busy Beaver function is defined as follows:

$\beta(n) =$  maximum number of 1's a TM in  $\mathcal{H}_n$  can produce from blank tape.

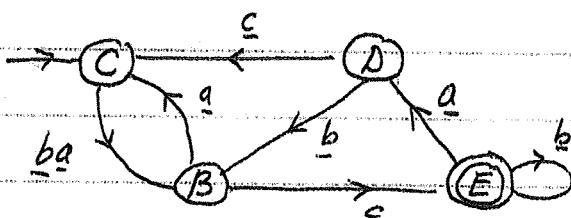
Here  $\mathcal{H}_n =$  set of all TMs with tape alphabet  $\{1, \text{blank}\}$  and  $n$  states, which halts when started on the blank tape

(b)

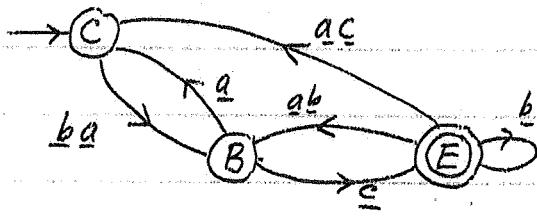
Corresponding GFA:



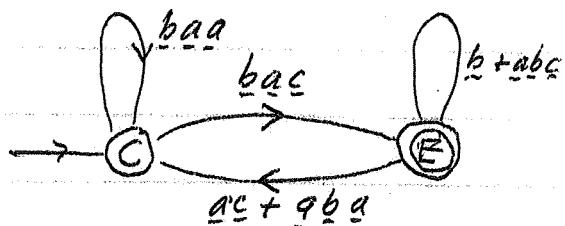
1. Eliminate A



3(b) 2. Eliminate D



3. Eliminate B



$$\begin{aligned} L(M) &= r_1^* r_2 (r_4 + r_3 r_1^* r_2) \\ &= (\underline{baa})^*. \underline{bac} ((b + abc) + (ac + aba).(\underline{baa})^*. \underline{bac})^* \end{aligned}$$

4(a)  $f$  is obtained from  $g$  and  $h$  by primitive recursion if  $f(x, 0) = g(x)$  and  $f(x, y+1) = h(x, y, f(x, y))$ . Here  $\underline{x} = \langle x_1, \dots, x_n \rangle$ ,  $g: \mathbb{N}^n \rightarrow \mathbb{N}$ ,  $f: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$  and  $h: \mathbb{N}^{n+2} \rightarrow \mathbb{N}$ .

$$\begin{aligned} f(x, y) &= 2x + 3y + 1. \text{ So } f(x, 0) = 2x + 1 \Rightarrow g(x) = 2x + 1 \\ \text{Also } f(x, y+1) &= 2x + 3(y+1) + 1 = (2x + 3y + 1) + 3 = f(x, y) + 3 \\ \Rightarrow h(x, y, f(x, y)) &= f(x, y) + 3. \quad \therefore h = s_0 s_0 s_0 I_3^{(2)} \\ \text{Now } g(0) &= 1 = s_0 0, \text{ and } g(y+1) = 2(y+1) + 1 = (2y+1) + 2 \\ \therefore g &= \text{prim. rec. } (s_0 0, s_0 s_0 I_2^{(2)}) \quad = g(y) + 2 \\ \therefore f &= \text{prim. rec. } (g, h) \\ &= \text{prim. rec. } (\text{prim. rec. } (s_0 0, s_0 s_0 I_2^{(2)}), s_0 s_0 s_0 I_3^{(3)}). \end{aligned}$$

and so is a primitive recursive function.

(b)  $f$  is obtained from  $g$  by minimization if  $f: \mathbb{N}^n \rightarrow \mathbb{N}$  and  $f(x) = \begin{cases} \text{smallest } y \text{ such that } g(x, y) = 0 \\ \text{undefined, if } g(x, y) > 0 \text{ for every } y. \end{cases}$

Here  $g: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$  must be a total function.

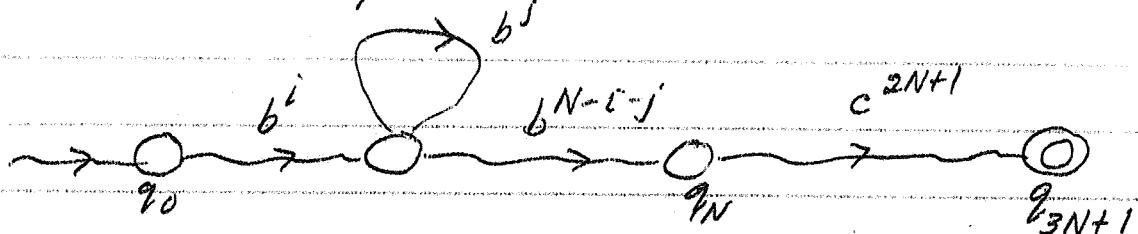
Take  $g(x, y) = x + 2 - y^2$ . Then  $(\mu y)[g(x, y) = 0]$  is just the function  $(\mu y)[x + 2 - y^2 = 0] = \sqrt{x+2}$ . Hence  $f = \mu[g, 0] = \mu[\text{MONUS} \circ [s_0 s_0 I_1^{(2)}, \text{MULT} \circ [I_2^{(2)}, I_2^{(2)}]], 0]$  and so is a recursive function.

5(a) A language  $L$  is Turing-acceptable if we can find a TM,  $M$  with input alphabet  $V$  such that for any input  $w \in V^*$ ,  $M$  will halt in an accepting state, if  $w \in L$ ; and  $M$  will halt in a non-acc. state or will fail to halt, if  $w \notin L$ .

- (b)  $\langle A, 00101 \rangle \vdash \langle D, 10101 \rangle \vdash \langle B, 10101 \rangle \vdash \langle C, 10001 \rangle \vdash \langle B, 10011 \rangle \vdash \langle E, 100101 \rangle \vdash \langle E, 100101 \rangle$ ,
- (c)  $L(M) = 01^* + 01^* 01(01)^* + 11(01)^*$ .

6(a)  $n \equiv 0 \pmod{3} \Rightarrow k \pmod{3} < 0$  which is not possible  
 $n \equiv 1 \pmod{3} \Rightarrow k \pmod{3} < 2(1) \Rightarrow k \equiv 0 \pmod{3}$  or  $1 \pmod{3}$   
 $n \equiv 2 \pmod{3} \Rightarrow k \pmod{3} < 2(2) \pmod{3} = 1 \pmod{3} \Rightarrow k \equiv 0 \pmod{3}$   
 $\therefore L_1 = (\underline{aaa})^* b (\underline{bbb})^* + a (\underline{aaa})^* b (\underline{bbb})^* + (\underline{aaa})^* b b (\underline{bbb})^*$   
Hence  $L_1$  is a regular language.

(b)  $L_2 = \{b^k c^n : 2k < n\}$ . Suppose  $L_2$  was regular. Then we can find an NFA  $M$  such that  $L(M) = L_2$ . Let  $N$  be the number of states in  $M$ . Then  $b^N c^{2N+1} \in L(M)$  because  $2(N) < 2N+1$  - (here  $k=N$  &  $n=2N+1$ ). Since  $M$  has only  $N$  states and it takes  $N+1$  states to process the  $b^N$ , the acceptance track of  $b^N c^{2N+1}$  must have a loop as shown below, with  $j \geq 1$ .



Now if we ride this loop twice we will see that  $M$  accepts  $b^2 b^j b^j b^{N-j} c^{2N+1} = b^{N+j} c^{2N+1} \notin L_2$  because  $2(N+j) \neq 2N+1$ . So this contradicts the fact that  $L(M) = L_2$ . Hence  $L_2$  cannot be regular.