

Answer all 6 questions. No calculators, formula sheets or cell-phones are allowed. Provide all reasoning and show all working. An unjustified answer will receive little or no credit. Begin each of the 6 questions on 6 separate pages.

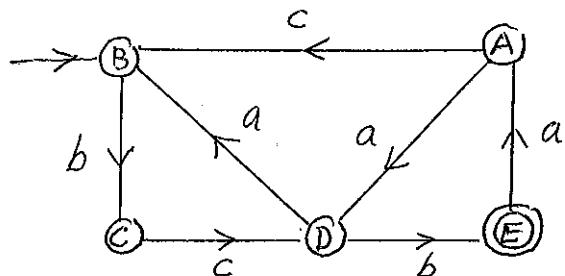
- (15) 1. (a) Find an NFA,  $M$ , which is equivalent to the RLG  $G$  given below.

$$G: \quad \neg E, \quad E \rightarrow 01, \quad E \rightarrow 01B, \quad B \rightarrow 0C, \quad C \rightarrow 11, \\ C \rightarrow 0D, \quad C \rightarrow E, \quad D \rightarrow 1B, \quad D \rightarrow \lambda.$$

- (b) Find an RLG,  $G$ , which is equivalent to the NFA in Problem 2 below.

- (15) 2. (a) Find a regular expression for the language accepted by the NFA shown on the right.

- (b) Define what is the Busy Beaver function.



- (15) 3. (a) Define what it means for  $f$  to be obtained from  $g$  and  $h$  by primitive recursion.

- (b) Show that  $f(x,y) = 2x+3y+3$  is a primitive recursive function by finding primitive recursive functions  $g$  and  $h$  such that  $f = \text{prec}[g,h]$ .

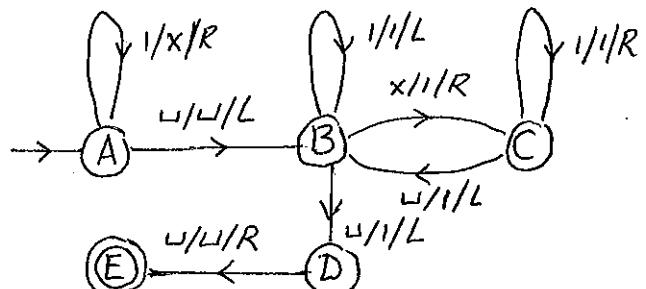
- (20) 4.(a) Define what it means for the function  $f: \mathbb{N}^n \rightarrow \mathbb{N}$  to be obtained from the total function  $g: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$  by minimization.

- (b) Let  $f(x) = \lceil (x/4) \rceil$  and  $h(x)=0$  (if  $x$  is a multiple of 4) &  $h(x)=1$  (otherwise). Show that  $f$  and  $h$  are  $\mu$ -recursive functions.

[You may use the fact that PRED, MONUS, ADD, MULT & SIGN are primitive recursive if needed in #4, but you are not allowed to do so in #3.]

- (15) 5. (a) Define what is a Turing semi-decidable binary relation  $R$  on  $\mathbb{N}$ .

- (b) Show what happens at each step if (i)  $\lambda$  and (ii)  $1$  are the inputs for the TM,  $M$ , shown on the right.



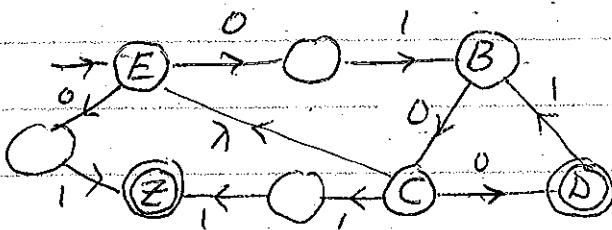
- (20) 6. Determine which of the following languages are regular and which are not.

$$(a) L_1 = \{a^k.b^n : k \pmod{3} > 2 + n^2 \pmod{3}\} \quad (b) L_2 = \{b^k.a^n : k > 2 + n^2\}.$$

[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.]

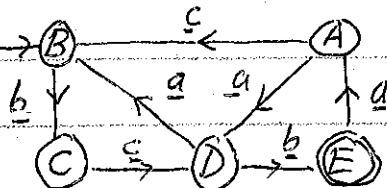
MAD 3512 - Theory of Algorithms Florida International Univ.  
 Solutions to Test #2 Spring 2014.

1(a)

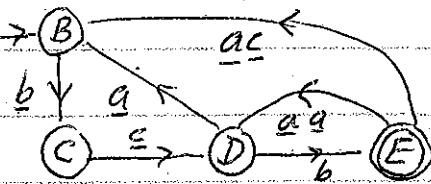


(b)  $\rightarrow B, B \rightarrow bC, C \rightarrow cD,$   
 $D \rightarrow aB, D \rightarrow bE, E \rightarrow aA,$   
 $A \rightarrow aD, A \rightarrow cB, E \rightarrow \lambda.$

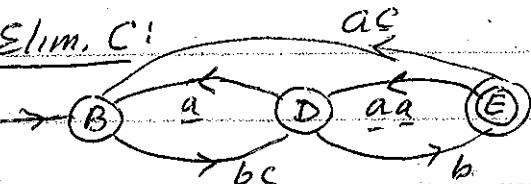
2(a)



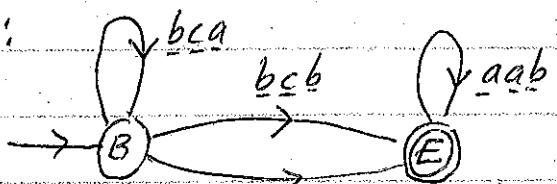
Elim. A:



Elim. C:



Elim. D:



$$L(M) = R_1^* R_2 (R_4 + R_3 R_1 R_2)^*$$

$$= (\underline{bca})^*. \underline{bc} \underline{b}. (\underline{aab} + (\underline{ac} + \underline{aa} \underline{a}). (\underline{bca})^*. \underline{bc} \underline{b})^*$$

(b)  $\beta(n)$  = maximum no. of 1's a TM in  $H_n$  can produce when started on the blank tape. Here  $H_n$  = set of TMs with  $n$  states & with tape alphabet  $\{1, \lambda\}$ , which halts on the blank tape.

3(a)  $f: N^{n+1} \rightarrow N$  is said to be obtained from  $g: N^n \rightarrow N$  &  $h: N \rightarrow N$  by primitive recursion if  $f(x, 0) = g(x)$  &  $f(x, s(y)) = h(x, y, f(x, y))$ . Here  $x = \langle x_1, \dots, x_n \rangle$ .

(b)  $f(x, y) = 2x + 3y + 3$ . So  $f(x, 0) = 2x + 3 \Rightarrow g(x) = 2x + 3$ , &  $f(x, s(y)) = 2x + 3(y+1) + 3 = f(x, y) + 3 \Rightarrow h(x, y, f(x, y)) = f(x, y) + 3$ .

$\therefore f = \text{prec}(g, h)$  &  $h = s_0 s_0 s_0 I_{3,3}$ . Now  $g(y) = 2y + 3$ .

So  $g(0) = 3$  &  $g(s(y)) = 2(y+1) + 3 = g(y) + 2$ . Hence

$g = \text{prec}(s_0 s_0 s_0 0, s_0 s_0 I_{2,2})$ . Thus  $f = \text{prec}(g, h) = \text{prec}(\text{prec}(s_0 s_0 s_0 0, s_0 s_0 I_{2,2}), s_0 s_0 s_0 I_{3,3})$  and so  $f$  is primitive recursive.

4(a)  $f: \mathbb{N}^n \rightarrow \mathbb{N}$  is obtained from the total function  $g: \mathbb{N}^{n+1} \rightarrow \mathbb{N}$  if  
 $f(x) = \begin{cases} \text{smallest value of } y \text{ such that } g(x, y) = 0 \text{ where } x = \langle x_1, \dots, x_n \rangle, \\ \text{undefined, if } g(x, y) \geq 1 \text{ for all } y \in \mathbb{N}. \end{cases}$

(b) Let  $g(x, y) = x - 4y$ . Then  $(\mu y)[g(x, y) = 0] = (\mu y)[x - 4y = 0]$   
 $= \lceil x/4 \rceil = f(x)$ . So  $f = \mu[g, 0] = \mu[\text{MONUS} \circ [I_{1,2} \wedge \text{MULT} \circ [(\text{SOSOSOSOZO} I_{2,2}) \wedge I_{2,2}], 0]$   
 $\therefore f$  is  $\mu$ -recursive.

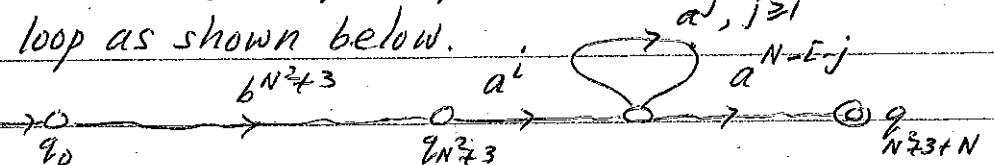
(c)  $h(x) = \text{SIGN}(4 \cdot \lceil x/4 \rceil - x) = \text{SIGN}(4 \cdot f(x) - x)$ . So  
 $h = \text{SIGN} \circ [\text{MONUS} \circ [\text{MULT} \circ [(\text{SOSOSOSOB}) \wedge f], I_{1,1}]]$ . So  $h$  is  $\mu$ -recursive

5(a)  $R$  is Turing semi-decidable if we can find a TM  $M$  such that  $M$  halts on  $\langle m, n \rangle$  in an acc. state, if  $\langle m, n \rangle \in R$  &  $M$  halts on  $\langle m, n \rangle$  in a non-accepting state or fails to halt, if  $\langle m, n \rangle \notin R$ .

(b)  $\langle A, \sqcup \rangle \vdash \langle B, \sqcup \sqcup \rangle \vdash \langle D, \sqcup 1 \rangle \vdash \langle E, 1 \rangle$  halts  
 $\langle A, 1 \rangle \vdash \langle A, x \sqcup \rangle \vdash \langle B, x \sqcup \rangle \vdash \langle C, 1 \sqcup \rangle \vdash \langle B, 11 \rangle \vdash \langle B, \sqcup 11 \rangle$   
 $\vdash \langle D, \sqcup 111 \rangle \vdash \langle E, 111 \rangle$  halts

6(a)  $n \equiv 0 \pmod{3} \Rightarrow k \pmod{3} > (2+0^2) \pmod{3} \Rightarrow$  no possible value of  $k$   
 $n \equiv 1 \pmod{3} \Rightarrow k \pmod{3} > (2+1^2) \pmod{3} \Rightarrow k \equiv 1 \text{ or } 2 \pmod{3}$   
 $n \equiv 2 \pmod{3} \Rightarrow k \pmod{3} > (2+2^2) \pmod{3} \Rightarrow k \equiv 1 \text{ or } 2 \pmod{3}$ , so  
 $(aa\bar{a})(\bar{a}a\bar{a})^* b(b\bar{b}\bar{b})^* + (\bar{a}aa)(\bar{a}aa)^* b\bar{b}(b\bar{b}\bar{b})^*$  describes  $L_1$ . So  $L_1$  is reg.

(b) Suppose  $L_2 = \{b^{k-n}a^n : k > 2+n^2\}$  was regular. Then we can find a  $\lambda$ -free NFA  $M$  with  $N$  states such that  $L(M) = L_2$ . Now  $b^{N^2+3}a^N \in L_2$  because if we take  $k = N^2+3$  &  $n = N$ , then  $k > 2+n^2$ . So  $M$  will accept  $b^{N^2+3}a^N$ . Since it takes  $N+1$  states to process the  $a^N$  part of the string, any acceptance track of  $b^{N^2+3}a^N$  must have a loop as shown below.



Now if we ride the loop twice, we will see that  $M$  accepts the string  $b^{N^2+3}a^{N+j}$ . But  $N^2+3 > 2+(N+j)^2$ , so this contradicts the fact that  $L(M) = L_2$ . Hence  $L_2$  is a non-regular language.