CS3133

HW#3

DUE: Monday, September 15

1.(9 points) Give regular expressions for the following sets of strings over $\{0,1\}$.

- ***a*** All strings for which the number of 0's is divisible by 3.
- ***b*** All strings which contain 00 and it contains 111, though not necessarily in that order.
- ***c*** All strings of odd length which contain 11.

2. (5 points) Which of the following "identities" are true **for all** regular expressions *R*, *S* and *T*? If an "identity" is false, provide a counterexample.

- ***a*** $R(S+T)=RS+RT$
- ***b*** $\left(R^*\right)^*=R^*$
- ***c*** $\left(R^*S^*\right)^*=(R+S)^*$
- ***d*** $RR=R$
- ***e*** $(R+S)^*=R^*+S^*$

3. (10 points) ***a*** Either give a regular expression for $L=\left\{0^l1^m2^n \mid l,m,n>0\right\}$ or prove that it doesn't exist.

b Either give a regular expression for $L=\left\{0^l1^m2^n \mid l>0 \wedge n>m>0\right\}$ or prove that it doesn't exist.

4. (10 points) Define function $\otimes:\Sigma^*\times\Sigma^*\rightarrow 2^{\Sigma^*}$ for any alphabet Σ by

- $\varepsilon\otimes x=\{x\}$,
- $x\otimes\varepsilon=\{x\}$,
- $ax\otimes by=\{a\}\cdot\{x\otimes by\}\cup\{b\}\cdot\{ax\otimes y\}$

for all $a,b\in\Sigma, x,y\in\Sigma^*$. For example, $01\otimes ab=\{01ab,0a1b,0ab1,ab01,a0b1,a01b\}$. We extend the definition to languages by $L_0\otimes L_1=\bigcup_{\substack{x\in L_0\\ y\in L_1}} x\otimes y$. For example,

$$\{01,0\}\otimes\{ab\}=\{01ab,0a1b,0ab1,ab01,a0b1,a01b,0ab,a0b,ab0\}$$

Prove that for any regular languages *A* and *B*, language $A\otimes B$ must be regular.

CS3133

Solutions to HW#3

1. ***a*** $\left(1^*01^*01^*01^*\right)^*$

b We can treat this as the union of two cases, where either 00 appears before 111 or 111 appears before 00.

$$\left((0+1)^* 00(0+1)^* 111(0+1)^*\right)+\left((0+1)^* 111(0+1)^* 00(0+1)^*\right)$$

c We can build the regular expression around 11. We know that either it is preceded by a string of odd length and followed by a string of even length or that it is preceded by a string of even length and followed by a string of odd length. The strings of even length are described by $(00+01+10+11)^*$ and the strings of odd length are described by $(00+01+10+11)^*(0+1)$. So an answer is

$$(00+01+10+11)^*(0+1)11(00+01+10+11)^*+(00+01+10+11)^*(0+1)11(00+01+10+11)^*$$

2. ***a*** true

b true

c true

d false Let $R=0$. $RR=00\neq 0=R$

e false Let $R=0$ and $S=1$. $01\in(R+S)^*=(0+1)^*$ though $01\notin R^*+S^*=0^*+1^*$

3. ***a*** $00^*11^*22^*$

bThere does not exist a regular expression for $L=\left\{0^l1^m2^n \mid l>0\wedge n>m>0\right\}$ because it is not a regular language. If it were, the Pumping Lemma assures us of the existence of a *k*>0 such that 01^k2^{k+1} can be expressed as *xyz* such that $\left\{xy^iz \mid i\geq 0\right\}\subseteq L$. There are two cases to consider:

- $0\in y$ In this case xy^0z does not contain any 0's, and hence $xy^0z\notin L$
- $0\notin y$ In this case *y* contains a nonempty string of 1's and does not contain any 2's. So xy^2z contains at least as many 1's as 2's, so it must follow that $xy^2z\notin L$.

Both of these cases are impossible.

4. If L_0 and L_1 are regular, then there are DFAs $M_0=(Q_0,\Sigma,\delta_0,s_0,F_0)$ and $M_1=(Q_1,\Sigma,\delta_1,s_1,F_1)$ such that $L_0=L(M_0)$ and $L_1=L(M_1)$. The idea behind the following machine is to run M_0 and M_1 in parallel, using nondeterminism to guess which machine should consume the next input symbol and advance its state. A string is accepted by $N_{M_0\otimes M_1}$ if there exists a path which consumes the string and takes each of M_0 **and** M_1 to final states.

$$N_{M_0\otimes M_1}=\left(Q_0\times Q_1,\Sigma,\Delta,\left\{(s_0,s_1)\right\},F_0\times F_1\right)$$

where $\Delta\left(\left(q_i, q_j\right), a\right) = \left\{\left(\delta_0\left(q_i, a\right), q_j\right), \left(q_i, \delta_1\left(q_j, a\right)\right)\right\}$.