Homework 2
Mike Gao
260915701
Prof. Prakash Panangaden

Question 1

Prove F is not regular:

$$F' = \overline{F} \cap ab^*c^* = \{ab^jc^k \mid j,k \geq 0 \text{ and } j \neq k\}$$
$$F'' = \overline{F'} \cap ab^*c^* = \{ab^rc^r \mid r \geq 0\}$$

Since $\{a^nb^n|n \geq 0\}$ is not regular by pumping lemma, F'' is not regular, therefore, F is not regular.

Show that it satisfy the pumping lemma: We choose $p = 2$, So $\forall w \in L$ such that $|w| \geq p$, $w = xyz$ where $|y| \geq 1$ and $|xy| \leq p$, four cases:

1. $i = 0$, so $w = b^jc^k$. Then $y = b$ or c is the first letter of the word, and $xy^nz = b^nb^{j-1}c^k \in L$ or $xy^nz = c^nc^{k-1} \in L$.
2. $i = 1$, so $j = k$, $w = ab^jc^j$. Set $y = a$ and $x = \epsilon$. $w = a^nb^jc^j \in L$.
3. $i = 2$, so $w = a^2b^jc^k$. Set $y = aa$ and $x = \epsilon$. $w = (aa)^nb^jc^k \in L$.
4. $i > 2$, so $w = a^ib^jc^k$. Set $y = a$ and $x = \epsilon$. $w = a^na^{i-1}b^jc^k \in L$.

It satisfy the pumping lemma, but is in no way contradicting, since the pumping lemma states that all regular language can be pumped, not the other way around.

Question 2

2.1

False. Suppose $A = nil$ and B is any non regular language, clearly, $A \subseteq B$ but B not regular.

2.2

False. Let $A = \{a^*\} \subseteq \Sigma^*$ be a regular language, $B = \{a^{2n}|n \geq 0\} \subseteq \Sigma^*$ be a non regular language. AB and A are regular, but B clearly not.

2.3

False. Counterexample: Assume $\Sigma = \{x, y\}$ The sets $S_1 = \{xy\}, S_2 = \{xxyy\}, S_3 = \{xxxyyy\}$ etc are regular. While $\bigcup_{i=1}^{\inf} S_i = \{x^ny^n|n \geq 0\}$ But it is clearly not regular. So this is false.

2.4

False. Let A be some non regular language on a finite alphabet Σ , thus $A \subseteq \Sigma^*$ and Σ^* is regular.

Question 3

3.1

Let DFA $D = (S, s_0, F, \delta)$ be a DFA that recognizes L, now we construct a NFA (Q, q_0, F', δ) that recognizes $CYC(L)$. The state space of the NFA to be $Q = S \times S \times S \times \{0, 1\}$ The first S tracks v, the second and third S guess where u start at, {0,1} tracks whether we're reading u or v.

Start state: $Q_0 = \{(s_0, s, s, 0) | s \in S\}$

Accepting state: $F^{'} = \{(s, t, s, 1) \mid t \in F\}$

If $b = 0$ and $\delta(t, a) \notin F$ then $\Delta((s, t, t_c, b), a) = \{(s, t', t_c, b) \mid t' = \delta(t, a)\}$.

If $b = 0$ and $\delta(t, a) \in F$ then

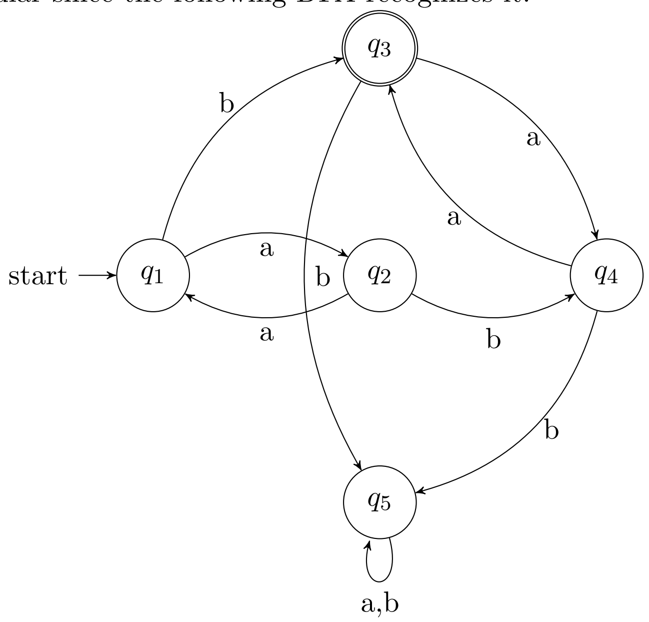
$\Delta((s, t, t_c, b), a) = \{(s, t', t_c, b) \mid t' = \delta(t, a)\} \cup \{(s', t', t_c, b') \mid b' = 1, t' = \delta(t, a)\}$

If $b = 1$ then $\Delta((s, t, t_c, b), a) = \{(s', t, t_c, b) \mid s' = \delta(s, a)\}$.

This works because the end of v will reach the start of u. u will end in an accepting state. So that $vu \in L$.

3.2

Let $L = a^n ba^n$, the language is not regular. $CYC(L) = a^m ba^n, m + n \equiv 0 \bmod 2$ which is regular since the following DFA recognizes it.



Question 4

Counter example: Consider the following finite language $L = \{aa, ac, ba, bc, cb, db\}$, it is clear that $a \sim b$ $c \sim d$ However, it is false that $ac \sim bd$ since $ac \in L$ while $bd \notin L$. Hence $\equiv_L$ is not a congruence relation.

Show $x \approx_L y \leftrightarrow \forall u, v \in \Sigma^*, uxv \in L \leftrightarrow uyv \in L$

Let us assume that $x \approx_L y$ and $p \approx_L q$ Let us have arbitrary $u, v \in \Sigma^*$. Since we have $x \approx_L y$, $uxpv \in L \leftrightarrow uypv \in L$ Since we have $p \approx_L q$ we can get $uypv \in L \leftrightarrow uyqv \in L$ Hence we have $xp \approx_L yq$

Question 5

5.1

Not Equivalent.

$$T = (\phi, \neg, \psi) \rightarrow (\neg\phi, \neg\psi) \rightarrow (\neg\phi, \neg\psi) \rightarrow (\neg\phi, \neg\psi) \rightarrow \cdots$$

$T \models (\Box\Diamond\phi \implies \Box\Diamond\psi)$ since the left side of the arrow is false, and false can imply anything. However, $T \not\models \Box(\phi \implies \Diamond\psi)$.

5.2

Not Equivalent

Let T be a transition system where $\phi \wedge \neg\psi$ for all odd step and $\neg\phi \wedge \psi$ for all even step, $T \models \Diamond\phi \wedge \Diamond\psi$ however $T \not\models \Diamond(\phi \wedge \psi)$ since no step in ϕ and ψ are both true.

5.3

Equivalent

$$\begin{aligned} \bigcirc\Diamond\phi &\equiv \bigcirc(\text{ true } U\phi) \\ &\equiv (\bigcirc \text{ true })U(\bigcirc\phi) \\ &\equiv \text{ true } U(\bigcirc\phi) \\ &\equiv \Diamond \bigcirc \phi \end{aligned}$$

Question 6

6.1

Formula : $\phi p = vX.p \wedge \bigcirc \bigcirc X$

$$\begin{aligned} (\phi p)_0 &= \text{true} \\ (\phi p)_1 &= p \\ (\phi p)_2 &= p \wedge \bigcirc \bigcirc p \\ (\phi p)_3 &= \text{true} \\ &\vdots \\ (\phi p)_n &= p \wedge \bigcirc \bigcirc p \wedge \bigcirc \bigcirc \bigcirc \bigcirc p \wedge \cdots \wedge \bigcirc^{n-1} p \end{aligned}$$

6.2

Formula: $\star p = \mu X.p \wedge \bigcirc \bigcirc X$

$$\begin{aligned} (\star p)_0 &= \text{false} \\ (\star p)_1 &= p \wedge \bigcirc \bigcirc \text{false} \\ &= \text{false} \end{aligned}$$

6.3

Because it says immediately after a p there is $\neg p$ and immediately after a $\neg p$ there is a p. Odd(P) doesn't necessarily have alternating p and $\neg p$ states.

Question 7

Lets first analyze the effect of the binary operator U. Let $\phi = \psi_1 U \psi_2$ be a LTL formula. ϕ is true on σ if there exists $j \geq 0$ such that $\delta[j...] \models \psi_2$ and for all $i < j$ $\delta[i...] \models \psi_1$ We can see that the truth value of ϕ depends on the existence of a state that satisfy ψ_2 after step $j \geq 0$ where all state before that satisfy ψ_1 no matter how many U operators were added to ϕ. The truth value stays the same for some $j \geq 0$ An example: let a and b be some states:

$$\begin{aligned}(((a) \to (b) \to (a) \to (a) \to \cdots) \models \phi) &\equiv (((a) \to (a) \to (b) \to (a) \to (a) \to \cdots) \models \phi) \\ &\equiv (((a) \to \cdots \to (a) \to (b) \to (a) \to (a) \to \cdots) \models \phi)\end{aligned}$$

Since U has this property, other LTL operators such as $\Box$ also has this property since they can be written as U, so, in a LTL formula ϕ with $U, \Diamond$ and $\Box$ only, ϕ has the same truth value on every σ_i for $i \geq 0$.

Now lets discuss $\bigcirc$: Let $\phi = \bigcirc^i \psi$. We can see that the truth value of ϕ depends soly on $\sigma[i]$ since ϕ is true on σ if $\sigma[i..] \models \psi$.

Consider a LTL formula ϕ with only $\bigcirc$ operators, since ϕ depends solely on $\sigma[i]$ and $\sigma[i]$ remains unchanged with variation of j, ϕ has the same truth value on every σ_j for $j > i$.

Thus, given a proposition p and any LTL formula ϕ containing n next operators, the fomula ϕ has the same truth value on every σ_i with $i > n$.

Now, it is pretty clear that $odd(p)$ cannot be expressed in LTL. Let's prove this by contradiction. Assume it can be expressed in LTL ϕ. Then ϕ must have a finite number of next operators. Assume ϕ has n next operators. Then, by the statement we proved earlier, σ_i is true for all $i > n$ which is a contradiction since one of σ_{n+1} and σ_{n+2} must be true, and the other one must be false.

Question 8

8.1

True. We want to show that both expression recognize the same language, such that: $L_\omega((E_1 + E_2) \cdot F^\omega) \equiv L_\omega(E_1 \cdot F^\omega + E_2 \cdot F^\omega)$

$$\begin{aligned}L_\omega((E_1 + E_2) \cdot F^\omega) &\equiv L_\omega(E_1 + E_2) \cdot L_\omega(F^\omega) \\ &\equiv (L_\omega(E_1) \cup L_\omega(E_2)) \cdot L_\omega(F^\omega) \\ &\equiv \{xy | x \in (L_\omega(E_1) \cup L_\omega(E_2)) \wedge y \in L_\omega(F^\omega)\} \\ &\equiv \{xy | x \in (L_\omega(E_1) \cup L_\omega(E_2)) \wedge y \in L_\omega(F^\omega)\} \\ &\equiv \{xy | (x \in L_\omega(E_1) \wedge y \in L_\omega(F^\omega)) \vee (x \in L_\omega(E_2) \wedge y \in L_\omega(F^\omega))\} \\ &\equiv \{xy | x \in L_\omega(E_1) \wedge y \in L_\omega(F^\omega)\} \cup \{xy | x \in L_\omega(E_2) \wedge y \in L_\omega(F^\omega)\} \\ &\equiv L_\omega(E_1 \cdot F^\omega) \cup L_\omega(E_2 \cdot F^\omega) \\ &\equiv L_\omega(E_1 \cdot F^\omega + E_2 \cdot F^\omega)\end{aligned}$$

8.2

False. Let $E = \epsilon$, $F_1 = x$, $F_2 = y$. $xyxyxyxyxy... \in E \cdot (F_1 + F_2)^\omega$ but $\notin E \cdot F_1^\omega + E \cdot F_2^\omega$

8.3

False. Consider $E = x, F = y$ Then $(E^*F)^\omega \equiv (x^*y)^\omega$ while $E^*F^\omega \equiv x^*y^\omega$ $(x^*y)^\omega$ recognize word $xyxyxyxyxy....$ but x^*y^ω is incapable of recognizing the same word.

Question 9

9.1

$t_0 \models [a]\langle b \rangle true$
$s_0 \not\models [a]\langle b \rangle true$

The formula state that there exists a connected state over path a from which does not exist path b. s_0 satisfy this formula because the path from s0 to s3 doesn't have a b path, whereas t_0 does not satisfy it because a path leads to a state with b path.

9.2

t_0 and s_0 agree on the following base case formulas: true, $\langle a \rangle T$, $\langle b \rangle T$, $\langle a \rangle \langle \langle b \rangle T \rangle$ $\langle b \rangle \langle \langle a \rangle T \rangle$

By induction, they all should agree on any boolean combination of these formulas, for example, $t_0, s_0 \models (\langle a \rangle \text{true}) \wedge (\langle a \rangle \langle b \rangle \text{true})$

Question 10