CS 516—Software Foundations via Formal Languages—Spring 2025

Problem Set 6

Model Answers

Problem 1

Easy mathematical inductions show that for all $n \in \mathbb{N}$, $\textbf{diff}(1^n) = n$ and $\textbf{diff}(0^n) = -2n$.

Lemma PS6.1.1
For all $n \in \mathbb{N}$, $1^{2n}0^n \in Y$.

Proof. Let X be the least subset of $\{0, 1\}^*$ such that:

(1) $\% \in X$;

(2) $1 \in X$;

(3) for all $x, y \in X$, $1x1y0 \in X$;

(4) for all $x, y \in X$, $xy \in X$.

In Problem Set 2, we proved $X = Y$. Consequently, it will suffice to show that, for all $n \in \mathbb{N}$, $1^{2n}0^n \in X$. We proceed by mathematical induction.

- **(basis step)** We have that $1^{2\cdot 0}0^0 = 1^0 0^0 = \%\% = \% \in X$, by Rule (1) of X's definition.
- **(inductive step)** Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $1^{2n}0^n \in X$. Then $1^{2(n+1)}0^{n+1} = 1^{2n+2}0^n0 = 1^{1+1+2n}0^n0 = 111^{2n}0^n0 = 1(\%)1(1^{2n}0^n)0 \in X$, by Rule (3) of X's definition, since $\% \in X$ (by Rule (1) of X's definition) and $1^{2n}0^n \in X$ (by the inductive hypothesis).

□

Suppose, toward a contradiction, that Y is regular. Thus there is an $n \in \mathbb{N} - \{0\}$ with the property of the Pumping Lemma, where Y has been substituted for L. Suppose $z = 1^{2n}0^n$. By Lemma PS6.1.1, we have that $z \in Y$. Thus, since $|z| = 2n+n = 3n \geq n$, it follows there are $u, v, w \in$ **Str** such that $z = uvw$ and properties (1)–(3) of the lemma hold. Since $uvw = z = 1^{2n}0^n = 1^n1^n0^n$, (1) tells us that there are $i, j, k \in \mathbb{N}$ such that

$$u = 1^i, \quad v = 1^j, \quad w = 1^k1^n0^n, \quad i + j + k = n.$$

By (2), we have that $j \geq 1$, and thus that $i + k = n - j < n$. By (3), we have that $1^{i+k+n}0^n = 1^i1^k1^n0^n = uw = u\%w = uv^0w \in Y$. Because $1^{i+k+n}0^n$ is a prefix of itself, we have that $i + k - n = i + k + n + -n + -n = (i + k + n) + -2n = \textbf{diff}(1^{i+k+n}) + \textbf{diff}(0^n) = \textbf{diff}(1^{i+k+n}0^n) \geq 0$, and thus that $i + k \geq n$. But since $i + k < n$—contradiction. Thus Y is not regular.

Problem 2

$$\begin{aligned}
\mathsf{A} &\to \mathsf{B}\langle\mathsf{3}\rangle \mid \langle\mathsf{0}\rangle\mathsf{C}, \\
\mathsf{B} &\to \mathsf{0B2} \mid \langle\mathsf{1}\rangle\mathsf{2}\langle\mathsf{2}\rangle, \\
\mathsf{C} &\to \mathsf{1C3} \mid \mathsf{1}\langle\mathsf{1}\rangle\langle\mathsf{2}\rangle, \\
\langle\mathsf{0}\rangle &\to \% \mid \mathsf{0}\langle\mathsf{0}\rangle, \\
\langle\mathsf{1}\rangle &\to \% \mid \mathsf{1}\langle\mathsf{1}\rangle, \\
\langle\mathsf{2}\rangle &\to \% \mid \mathsf{2}\langle\mathsf{2}\rangle, \\
\langle\mathsf{3}\rangle &\to \% \mid \mathsf{3}\langle\mathsf{3}\rangle
\end{aligned}$$

Problem 3

(a) First we give some standard definitions:

$$\begin{aligned}
\textbf{minAndRen} &= \textbf{renameStatesCanonically} \circ \textbf{minimize}, \\
\textbf{efaToDFA} &= \textbf{nfaToDFA} \circ \textbf{efaToNFA}, \\
\textbf{strToEFA} &= \textbf{faToEFA} \circ \textbf{strToFA}, \\
\textbf{allStrEFA} &= \textbf{closure}(\textbf{union}(\textbf{symToNFA}\, \mathsf{0}, \textbf{symToNFA}\, \mathsf{1})), \text{ and} \\
\textbf{allStrDFA} &= \textbf{minAndRen}(\textbf{efaToDFA}\, \textbf{allStrEFA}).
\end{aligned}$$

Thus $\textbf{minAndRen} \in \textbf{DFA} \to \textbf{DFA}$, $\textbf{efaToDFA} \in \textbf{EFA} \to \textbf{DFA}$, $\textbf{strToEFA} \in \textbf{Str} \to \textbf{EFA}$, $\textbf{allStrEFA} \in \textbf{EFA}$ and $\textbf{allStrDFA} \in \textbf{DFA}$.

Next, we define $\textbf{hasSubEFA} \in \{\mathsf{0},\mathsf{1}\}^* \to \textbf{EFA}$ by: for all $x \in \{\mathsf{0},\mathsf{1}\}^*$,

$$\textbf{hasSubEFA}\, x = \textbf{concat}(\textbf{allStrDFA}, \textbf{concat}(\textbf{strToEFA}\, x, \textbf{allStrDFA})).$$

Define $\textbf{hasSubDFA} \in \{\mathsf{0},\mathsf{1}\}^* \to \textbf{DFA}$ by:

$$\textbf{hasSubDFA} = \textbf{minAndRen} \circ \textbf{efaToDFA} \circ \textbf{hasSubEFA}.$$

Define $\textbf{hasNotSubDFA} \in \{\mathsf{0},\mathsf{1}\}^* \to \textbf{DFA}$ by: for all $x \in \{\mathsf{0},\mathsf{1}\}^*$,

$$\textbf{hasNotSubDFA}\, x = \textbf{minAndRen}(\textbf{minus}(\textbf{allStrDFA}, \textbf{hasSubDFA}\, x)).$$

Define $\textbf{someUnmatchedEFA} \in \{\mathsf{0},\mathsf{1}\}^* \times \{\mathsf{0},\mathsf{1}\}^* \to \textbf{EFA}$ by: for all $x, y \in \{\mathsf{0},\mathsf{1}\}^*$,

$$\begin{aligned}
&\textbf{someUnmatchedEFA}(x, y) \\
&= \textbf{concat}(\textbf{hasNotSubDFA}\, y, \textbf{concat}(\textbf{strToEFA}\, x, \textbf{hasNotSubDFA}\, y)).
\end{aligned}$$

Define $\textbf{someUnmatchedDFA} \in \{\mathsf{0},\mathsf{1}\}^* \times \{\mathsf{0},\mathsf{1}\}^* \to \textbf{DFA}$ by:

$$\textbf{someUnmatchedDFA} = \textbf{minAndRen} \circ \textbf{efaToDFA} \circ \textbf{someUnmatchedEFA}.$$

Define $\textbf{allMatchedDFA} \in \{\mathsf{0},\mathsf{1}\}^* \times \{\mathsf{0},\mathsf{1}\}^* \to \textbf{DFA}$ by: for all $x, y \in \{\mathsf{0},\mathsf{1}\}^*$,

$$\textbf{allMatchedDFA}(x, y) = \textbf{minAndRen}(\textbf{minus}(\textbf{allStrDFA}, \textbf{someUnmatchedDFA}(x, y))).$$

Finally, define **dcsDFA** $\in \{0,1\}^* \times \{0,1\}^* \to$ **DFA** by: for all $x, y \in \{0,1\}^*$,

$$\mathbf{dcsDFA}(x,y) = \mathbf{minAndRen}(\mathbf{inter}(\mathbf{allMatchedDFA}(x,y), \mathbf{allMatchedDFA}(y,x))).$$

(b) Our definition of `dcsDFA` is in the file `ps6.sml`:

```
val zero      = Sym.fromString "0";
val one       = Sym.fromString "1";
val minAndRen =
      DFA.renameStatesCanonically o DFA.minimize;
val efaToDFA  = nfaToDFA o efaToNFA;
val strToEFA  = faToEFA o strToFA;
val allStrEFA =
      EFA.closure
      (EFA.union(injNFAToEFA(symToNFA zero), injNFAToEFA(symToNFA one)));
val allStrDFA = minAndRen(efaToDFA allStrEFA);

fun hasSubEFA x =
      EFA.concat
      (injDFAToEFA allStrDFA,
       EFA.concat(strToEFA x, injDFAToEFA allStrDFA));

val hasSubDFA = minAndRen o efaToDFA o hasSubEFA;

fun hasNotSubDFA x = minAndRen(DFA.minus(allStrDFA, hasSubDFA x));

fun someUnmatchedEFA(x, y) =
      EFA.concat
      (injDFAToEFA(hasNotSubDFA y),
       EFA.concat(strToEFA x, injDFAToEFA(hasNotSubDFA y)));

val someUnmatchedDFA = minAndRen o efaToDFA o someUnmatchedEFA;

fun allMatchedDFA(x, y) =
      minAndRen(DFA.minus(allStrDFA, someUnmatchedDFA(x, y)));

fun dcsDFA(x, y) =
      minAndRen(DFA.inter(allMatchedDFA(x, y), allMatchedDFA(y, x)));
```

We load it into Forlan:

```
- use "ps6.sml";
[opening ps6.sml]
val zero = - : sym
val one = - : sym
val minAndRen = fn : dfa -> dfa
val efaToDFA = fn : efa -> dfa
val strToEFA = fn : str -> efa
val allStrEFA = - : efa
```

```
val allStrDFA = - : dfa
val hasSubEFA = fn : str -> efa
val hasSubDFA = fn : str -> dfa
val hasNotSubDFA = fn : str -> dfa
val someUnmatchedEFA = fn : str * str -> efa
val someUnmatchedDFA = fn : str * str -> dfa
val allMatchedDFA = fn : str * str -> dfa
val dcsDFA = fn : str * str -> dfa
val it = () : unit
```

And then we execute:

```
- val dfa1 = dcsDFA(Str.fromString "11", Str.fromString "00");
val dfa1 = - : dfa
- DFA.numStates dfa1;
val it = 8 : int
- val dfa2 = dcsDFA(Str.fromString "011", Str.fromString "110");
val dfa2 = - : dfa
- DFA.numStates dfa2;
val it = 29 : int
```

(c) First, we note that, because **renameStatesCanonically** and **minimize** preserve the meaning of DFAs, for all DFAs M,

$$\begin{aligned} L(\mathbf{minAndRen}\,M) &= L(\mathbf{renameStatesCanonically}(\mathbf{minimize}\,M)) \\ &= L(\mathbf{minimize}\,M) = L(M), \end{aligned}$$

and thus $\mathbf{minAndRen}\,M \approx M$.

Lemma PS6.3.1
For all DFAs M, $\mathbf{minimize}(\mathbf{minAndRen}\,M)$ is isomorphic to $\mathbf{minAndRen}\,M$.

Proof. We have that $\mathbf{minAndRen}\,M = \mathbf{renameStatesCanonically}(\mathbf{minimize}\,M)$ is isomorphic to $\mathbf{minimize}\,M$. Thus it will suffice to show that $\mathbf{minimize}(\mathbf{minAndRen}\,M)$ is isomorphic to $\mathbf{minimize}\,M$. By Theorem 3.13.12, it will suffice to show that

(1) $\mathbf{minimize}(\mathbf{minAndRen}\,M) \approx M$;

(2) $\mathbf{alphabet}(\mathbf{minimize}(\mathbf{minAndRen}\,M)) = \mathbf{alphabet}(L(M))$; and

(3) $|Q_{\mathbf{minimize}(\mathbf{minAndRen}\,M)}| \leq |Q_{\mathbf{minimize}\,M}|$.

For (1), we have that $\mathbf{minimize}(\mathbf{minAndRen}\,M) \approx \mathbf{minAndRen}\,M \approx M$.
For (2), by Theorem 3.13.12, we have that

$$\mathbf{alphabet}(\mathbf{minimize}(\mathbf{minAndRen}\,M)) = \mathbf{alphabet}(L(\mathbf{minAndRen}\,M)) = \mathbf{alphabet}(L(M)).$$

For (3), by Theorem 3.13.12, we have that

$$\begin{aligned} |Q_{\mathbf{minimize}(\mathbf{minAndRen}\,M)}| &\leq |Q_{\mathbf{minAndRen}\,M}| \\ &= |Q_{\mathbf{renameStatesCanonically}(\mathbf{minimize}\,M)}| = |Q_{\mathbf{minimize}\,M}|. \end{aligned}$$

□

Define $\mathbf{HasSub} \in \{0,1\}^* \to \mathcal{P}(\{0,1\}^*)$ by: for all $x \in \{0,1\}^*$, $\mathbf{HasSub}\, x = \{\, w \in \{0,1\}^* \mid x$ is a substring of $w\,\}$.

Clearly:

Lemma PS6.3.2
For all $x \in \{0,1\}^*$, $\mathbf{HasSub}\, x = \{0,1\}^* \{x\} \{0,1\}^*$.

Lemma PS6.3.2 and easy calculations show:

Lemma PS6.3.3

(1) For all $x \in \{0,1\}^*$, $L(\mathbf{hasSubEFA}\, x) = \mathbf{HasSub}\, x$.

(2) For all $x \in \{0,1\}^*$, $L(\mathbf{hasSubDFA}\, x) = \mathbf{HasSub}\, x$.

Define $\mathbf{HasNotSub} \in \{0,1\}^* \to \mathcal{P}(\{0,1\}^*)$ by: for all $x \in \{0,1\}^*$, $\mathbf{HasNotSub}\, x = \{\, w \in \{0,1\}^* \mid x$ is not a substring of $w\,\}$.

Because complementation corresponds to negation, we have:

Lemma PS6.3.4
For all $x \in \{0,1\}^*$, $\mathbf{HasNotSub}\, x = \{0,1\}^* - \mathbf{HasSub}\, x$.

Lemmas PS6.3.3 and PS6.3.4, and an easy calculation show:

Lemma PS6.3.5
For all $x \in \{0,1\}^*$, $L(\mathbf{hasNotSubDFA}\, x) = \mathbf{HasNotSub}\, x$.

Define $\mathbf{SomeUnmatched} \in \{0,1\}^* \times \{0,1\}^* \to \mathcal{P}(\{0,1\}^*)$ by: for all $x, y \in \{0,1\}^*$, $\mathbf{SomeUnmatched}(x, y)$ is the set of all $w \in \{0,1\}^*$ such that there are $u, v \in \{0,1\}^*$ such that $w = uxv$, y is not a substring of u, and y is not a substring of v.

It is easy to show:

Lemma PS6.3.6
For all $x, y \in \{0,1\}^*$, $\mathbf{SomeUnmatched}(x, y) = \mathbf{HasNotSub}\, y\, \{x\}\, \mathbf{HasNotSub}\, y$.

Lemmas PS6.3.5 and PS6.3.6, and easy calculations show:

Lemma PS6.3.7

(1) For all $x, y \in \{0,1\}^*$, $L(\mathbf{someUnmatchedEFA}(x, y)) = \mathbf{SomeUnmatched}(x, y)$.

(2) For all $x, y \in \{0,1\}^*$, $L(\mathbf{someUnmatchedDFA}(x, y)) = \mathbf{SomeUnmatched}(x, y)$.

Define $\mathbf{AllMatched} \in \{0,1\}^* \times \{0,1\}^* \to \mathcal{P}(\{0,1\}^*)$ by: for all $x, y \in \{0,1\}^*$, $\mathbf{AllMatched}(x, y)$ is the set of all $w \in \{0,1\}^*$ such that, for all $u, v \in \{0,1\}^*$, if $w = uxv$, then y is a substring of u, or y is a substring of v.

Lemma PS6.3.8
For all $x, y \in \{0,1\}^*$, $\mathbf{AllMatched}(x, y) = \{0,1\}^* - \mathbf{SomeUnmatched}(x, y)$.

Proof. Follows from the relationship between complementation and negation, since, if $w \in \{0,1\}^*$, then:

there do not exist $u, v \in \{0, 1\}^*$ such that $w = uxv$, and y is not a substring of u, and y is not a substring of v

iff

for all $u, v \in \{0, 1\}^*$ it is not the case that: $w = uxv$, and y is not a substring of u, and y is not a substring of v

iff

for all $u, v \in \{0, 1\}^*$, $w \neq uxv$, or y is a substring of u, or y is a substring of v

iff

for all $u, v \in \{0, 1\}^*$, $w \neq uxv$, or: y is a substring of u, or y is a substring of v

iff

for all $u, v \in \{0, 1\}^*$, if $w = uxv$, then y is a substring of u, or y is a substring of v.

□

Lemmas PS6.3.7 and PS6.3.8, and an easy calculation show:

Lemma PS6.3.9
For all $x, y \in \{0, 1\}^*$, $L(\mathbf{allMatchedDFA}(x, y)) = \mathbf{AllMatched}(x, y)$.

Because intersection corresponds to conjunction, we have:

Lemma PS6.3.10
For all $x, y \in \{0, 1\}^*$, $\mathbf{DCS}(x, y) = \mathbf{AllMatched}(x, y) \cap \mathbf{AllMatched}(y, x)$.

Lemmas PS6.3.9 and PS6.3.10, and an easy calculation, show:

Lemma PS6.3.11
For all $x, y \in \{0, 1\}^*$, $L(\mathbf{dcsDFA}(x, y)) = \mathbf{DCS}(x, y)$.

Finally, Lemma PS6.3.1 tells us that:

Lemma PS6.3.12
For all $x, y \in \{0, 1\}^*$, $\mathbf{minimize}(\mathbf{dcsDFA}(x, y))$ *is isomorphic to* $\mathbf{dcsDFA}(x, y)$.