CIS 570 — Introduction to Formal Language Theory — Fall 2008

Exercise Set 6

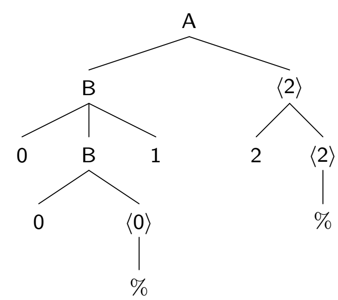
Model Answers

Exercise 1

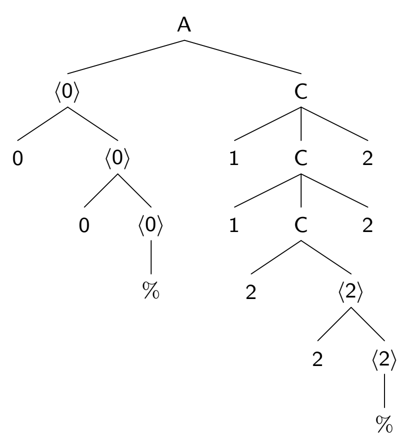
(a)

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

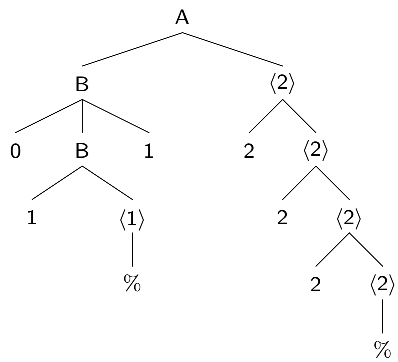
(b) Let pt_1 be the parse tree with yield 0012:



Let pt_2 be the parse tree with yield 00112222:



And, let pt_3 be the parse tree with yield 011222:



To check that our answers are correct, we place the description

```
{variables} A, B, C, <0>, <1>, <2>
{start variable} A
{productions}
A -> B<2> | <0>C;
B -> 0<0> | 1<1> | 0B1;
C -> 1<1> | 2<2> | 1C2;
<0> -> % | 0<0>;
<1> -> % | 1<1>;
<2> -> % | 2<2>
```

of G in the file `es6-ex1-gram`, and then load G into Forlan:

```
- val gram = Gram.input "es6-ex1-gram";
val gram = - : gram
```

Next, we put the description

```
A(B(0,
    B(0,
      <0>(%)),
    1),
  <2>(2,
      <2>(%)))
```

of pt_1 in the file `es6-ex1-pt1`, load pt_1 into Forlan, and check that it has the required properties:

```
- val pt1 = PT.input "es6-ex1-pt1";
val pt1 = - : pt
- Gram.validPT gram pt1;
val it = true : bool
- Sym.output("", PT.rootLabel pt1);
A
val it = () : unit
- Str.output("", PT.yield pt1);
0012
```

```
val it = () : unit
```

Next, we put the description

```
A(<0>(0,
      <0>(0,
          <0>(%))),
  C(1,
    C(1,
      C(2,
        <2>(2,
            <2>(%))),
      2),
    2))
```

of pt_2 in the file es6-ex1-pt2, load pt_2 into Forlan, and check that it has the required properties:

```
- val pt2 = PT.input "es6-ex1-pt2";
val pt2 = - : pt
- Gram.validPT gram pt2;
val it = true : bool
- Sym.output("", PT.rootLabel pt2);
A
val it = () : unit
- Str.output("", PT.yield pt2);
00112222
val it = () : unit
```

Finally, we put the description

```
A(B(0,
    B(1,
      <1>(%)),
    1),
  <2>(2,
      <2>(2,
          <2>(2,
              <2>(%)))))
```

of pt_3 in the file es6-ex1-pt3, load pt_3 into Forlan, and check that it has the required properties:

```
- val pt3 = PT.input "es6-ex1-pt3";
val pt3 = - : pt
- Gram.validPT gram pt3;
val it = true : bool
- Sym.output("", PT.rootLabel pt3);
A
val it = () : unit
- Str.output("", PT.yield pt3);
```

```
011222
val it = () : unit
```

(c) First, we define a function for testing whether a string is generated by G:

```
- val generated = Gram.generated gram;
val generated = fn : str -> bool
```

Next, we check that some strings in X are generated by G. We put the strings

```
0, 1, 2, 00, 01, 02, 11, 12, 22, 000, 001, 002, 011, 022, 111, 112,
122, 222, 0000, 0001, 0002, 0011, 0012, 0022, 0111, 0112, 0122, 0222,
1111, 1112, 1122, 1222, 2222, 00000, 00001, 00002, 00011, 00012,
00022, 00111, 00112, 00122, 00222, 01111, 01112, 01122, 01222, 02222,
11111, 11112, 11122, 11222, 12222, 22222, 000000, 000001, 000002,
000011, 000012, 000022, 000111, 000112, 000122, 000222, 001111,
001112, 001222, 002222, 011111, 011112, 011122, 011222, 012222,
022222, 111111, 111112, 111122, 111222, 112222, 122222, 222222,
0000000, 0000001, 0000002, 0000011, 0000012, 0000022, 0000111,
0000112, 0000122, 0000222, 0001111, 0001112, 0001122, 0001222,
0002222, 0011111, 0011112, 0011122, 0011222, 0012222, 0022222,
0111111, 0111112, 0111122, 0111222, 0112222, 0122222, 0222222,
1111111, 1111112, 1111122, 1111222, 1112222, 1122222, 1222222, 2222222
```

in the file `es6-ex1-positive-str-set`, and then proceed as follows:

```
- val positive = StrSet.input "es6-ex1-positive-str-set";
val positive = - : str set
- map generated (Set.toList positive);
val it =
  [true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true,true,true,true,
   true,true,true,true,true,true,true,true,true,true,true,true] : bool list
```

Next, we check that some strings that are not in X are generated by G. We put the strings

```
%, 012, 001122, 000111222,
02102, 02110, 02111, 02112, 02120, 02121, 02122, 02200, 02201, 02202,
02210, 02211, 02212, 02220, 02221, 10000, 10001, 10002, 10010, 10011,
10012, 10020, 10021, 10022, 10100, 10101, 10102, 10110, 10111, 10112,
10120, 10121, 10122, 10200, 10201, 10202, 10210, 10211, 10212, 10220,
10221, 10222, 11000, 11001, 11002, 11010, 11011, 11012, 11020, 11021,
11022, 11100, 11101, 11102, 11110, 11120, 11121, 11200, 11201, 11202,
11210, 11211, 11212, 11220, 11221, 12000, 12001, 12002, 12010, 12011,
12012, 12020, 12021, 12022, 12100, 12101, 12102, 12110, 12111, 12112,
12120, 12121, 12122, 12200, 12201, 12202, 12210, 12211, 12212, 12220,
12221, 20000, 20001, 20002, 20010, 20011, 20012, 20020, 20021
```

in the file `es6-ex1-negative-str-set`, and then proceed as follows:

```
- val negative = StrSet.input "es6-ex1-negative-str-set";
val negative = - : str set
- map generated (Set.toList negative);
val it =
  [false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false,false,false,false,false,false,
   false,false,false,false,false,false,false] : bool list
```

(d) Let

$$Y = \{\, \mathsf{0}^i\mathsf{1}^j \mid i, j \in \mathbb{N} \text{ and } i \neq j \,\},$$
$$Z = \{\, \mathsf{1}^j\mathsf{2}^k \mid j, k \in \mathbb{N} \text{ and } j \neq k \,\}.$$

We will show that $\Pi_{\mathsf{A}} = X$, $\Pi_{\mathsf{B}} = Y$, $\Pi_{\mathsf{C}} = Z$, $\Pi_{\langle\mathsf{0}\rangle} = \{\mathsf{0}\}^*$, $\Pi_{\langle\mathsf{1}\rangle} = \{\mathsf{1}\}^*$ and $\Pi_{\langle\mathsf{2}\rangle} = \{\mathsf{2}\}^*$.

Lemma ES6.1.1
$\{\mathsf{0}\}^* \subseteq \Pi_{\langle\mathsf{0}\rangle}$.

Proof. It will suffice to show that, for all $n \in \mathbb{N}$, $\mathsf{0}^n \in \Pi_{\langle\mathsf{0}\rangle}$. We proceed by mathematical induction.

- (Basis Step) Because $\langle\mathsf{0}\rangle \to \% \in P$, we have that $\mathsf{0}^0 = \% \in \Pi_{\langle\mathsf{0}\rangle}$.
- (Inductive Step) Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $\mathsf{0}^n \in \Pi_{\langle\mathsf{0}\rangle}$. Because $\langle\mathsf{0}\rangle \to \mathsf{0}\langle\mathsf{0}\rangle \in P$, it follows that $\mathsf{0}^{n+1} = \mathsf{0}\mathsf{0}^n \in \Pi_{\langle\mathsf{0}\rangle}$.

□

Lemma ES6.1.2
$\{\mathsf{1}\}^* \subseteq \Pi_{\langle\mathsf{1}\rangle}$.

Proof. It will suffice to show that, for all $n \in \mathbb{N}$, $\mathsf{1}^n \in \Pi_{\langle\mathsf{1}\rangle}$. We proceed by mathematical induction.

- (Basis Step) Because $\langle\mathsf{1}\rangle \to \% \in P$, we have that $\mathsf{1}^0 = \% \in \Pi_{\langle\mathsf{1}\rangle}$.
- (Inductive Step) Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $\mathsf{1}^n \in \Pi_{\langle\mathsf{1}\rangle}$. Because $\langle\mathsf{1}\rangle \to \mathsf{1}\langle\mathsf{1}\rangle \in P$, it follows that $\mathsf{1}^{n+1} = \mathsf{1}\mathsf{1}^n \in \Pi_{\langle\mathsf{1}\rangle}$.

□

Lemma ES6.1.3
$\{2\}^* \subseteq \Pi_{\langle 2\rangle}$.

Proof. It will suffice to show that, for all $n \in \mathbb{N}$, $2^n \in \Pi_{\langle 2\rangle}$. We proceed by mathematical induction.

- (Basis Step) Because $\langle 2\rangle \to \% \in P$, we have that $2^0 = \% \in \Pi_{\langle 2\rangle}$.
- (Inductive Step) Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $2^n \in \Pi_{\langle 2\rangle}$. Because $\langle 2\rangle \to 2\langle 2\rangle \in P$, it follows that $2^{n+1} = 22^n \in \Pi_{\langle 2\rangle}$.

□

Lemma ES6.1.4
$Y \subseteq \Pi_{\mathsf{B}}$.

Proof. First, we show a fact that we call (†) below: for all $x \in \{0\}\{0\}^* \cup \{1\}\{1\}^*$ and $n \in \mathbb{N}$, $0^n x 1^n \in \Pi_{\mathsf{B}}$. Let $x \in \{0\}\{0\}^* \cup \{1\}\{1\}^*$. We use mathematical induction to show that, for all $n \in \mathbb{N}$, $0^n x 1^n \in \Pi_{\mathsf{B}}$.

- (Basis Step) Because $x \in \{0\}\{0\}^* \cup \{1\}\{1\}^*$, there are two cases to consider.
  - Suppose $x \in \{0\}\{0\}^*$. Thus $x = 0y$ for some $y \in \{0\}^*$. By Lemma ES6.1.1, we have that $y \in \{0\}^* \subseteq \Pi_{\langle 0\rangle}$. Thus, since $\mathsf{B} \to 0\langle 0\rangle \in P$, we have that $0^0 x 1^0 = \% x \% = x = 0y \in \Pi_{\mathsf{B}}$.
  - Suppose $x \in \{1\}\{1\}^*$. Thus $x = 1y$ for some $y \in \{1\}^*$. By Lemma ES6.1.2, we have that $y \in \{1\}^* \subseteq \Pi_{\langle 1\rangle}$. Thus, since $\mathsf{B} \to 1\langle 1\rangle \in P$, we have that $0^0 x 1^0 = \% x \% = x = 1y \in \Pi_{\mathsf{B}}$.
- (Inductive Step) Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $0^n x 1^n \in \Pi_{\mathsf{B}}$. Thus, since $\mathsf{B} \to 0\mathsf{B}1 \in P$, it follows that $0^{n+1} x 1^{n+1} = 0(0^n x 1^n)1 \in \Pi_B$.

Now, we use (†) to show that $Y \subseteq \Pi_{\mathsf{B}}$. Suppose $w \in Y$, so that $w = 0^i 1^j$, for some $i, j \in \mathbb{N}$ such that $i \neq j$. Thus either $i < j$ or $j < i$. There are two cases to consider.

- Suppose $i < j$. Thus $j = n+i$ for some $n \in \mathbb{N} - \{0\}$. Because $n \geq 1$, we have that $1^n = 11^{n-1} \in \{1\}\{1\}^* \subseteq \{0\}\{0\}^* \cup \{1\}\{1\}^*$. Hence, (†) tells us that $w = 0^i 1^j = 0^i 1^{n+i} = 0^i 1^n 1^i \in \Pi_{\mathsf{B}}$.
- Suppose $j < i$. Thus $i = j+n$ for some $n \in \mathbb{N} - \{0\}$. Because $n \geq 1$, we have that $0^n = 00^{n-1} \in \{0\}\{0\}^* \subseteq \{0\}\{0\}^* \cup \{1\}\{1\}^*$. Hence, (†) tells us that $w = 0^i 1^j = 0^{j+n} 1^j = 0^j 0^n 1^j \in \Pi_{\mathsf{B}}$.

□

Lemma ES6.1.5
$Z \subseteq \Pi_{\mathsf{C}}$.

Proof. First, we show a fact that we call (†) below: for all $x \in \{1\}\{1\}^* \cup \{2\}\{2\}^*$ and $n \in \mathbb{N}$, $1^n x 2^n \in \Pi_{\mathsf{C}}$. Let $x \in \{1\}\{1\}^* \cup \{2\}\{2\}^*$. We use mathematical induction to show that, for all $n \in \mathbb{N}$, $1^n x 2^n \in \Pi_{\mathsf{C}}$.

- (Basis Step) Because $x \in \{1\}\{1\}^* \cup \{2\}\{2\}^*$, there are two cases to consider.
  - Suppose $x \in \{1\}\{1\}^*$. Thus $x = 1y$ for some $y \in \{1\}^*$. By Lemma ES6.1.2, we have that $y \in \{1\}^* \subseteq \Pi_{\langle 1\rangle}$. Thus, since $\mathsf{C} \to 1\langle 1\rangle \in P$, we have that $1^0 x 2^0 = \% x \% = x = 1y \in \Pi_{\mathsf{C}}$.

  - Suppose $x \in \{2\}\{2\}^*$. Thus $x = 2y$ for some $y \in \{2\}^*$. By Lemma ES6.1.3, we have that $y \in \{2\}^* \subseteq \Pi_{\langle 2\rangle}$. Thus, since $\mathsf{C} \rightarrow 2\langle 2\rangle \in P$, we have that $1^0x2^0 = \%x\% = x = 2y \in \Pi_{\mathsf{C}}$.

- (Inductive Step) Suppose $n \in \mathbb{N}$, and assume the inductive hypothesis: $1^nx2^n \in \Pi_{\mathsf{C}}$. Thus, since $\mathsf{C} \rightarrow 1\mathsf{C}2 \in P$, it follows that $1^{n+1}x2^{n+1} = 1(1^nx2^n)2 \in \Pi_C$.

Now, we use (†) to show that $Z \subseteq \Pi_{\mathsf{C}}$. Suppose $w \in Z$, so that $w = 1^j2^k$, for some $j, k \in \mathbb{N}$ such that $j \neq k$. Thus either $j < k$ or $k < j$. There are two cases to consider.

- Suppose $j < k$. Thus $k = n + j$ for some $n \in \mathbb{N} - \{0\}$. Because $n \geq 1$, we have that $2^n = 22^{n-1} \in \{2\}\{2\}^* \subseteq \{1\}\{1\}^* \cup \{2\}\{2\}^*$. Hence, (†) tells us that $w = 1^j2^k = 1^j2^{n+j} = 1^j2^n2^j \in \Pi_{\mathsf{C}}$.

- Suppose $k < j$. Thus $j = k + n$ for some $n \in \mathbb{N} - \{0\}$. Because $n \geq 1$, we have that $1^n = 11^{n-1} \in \{1\}\{1\}^* \subseteq \{1\}\{1\}^* \cup \{2\}\{2\}^*$. Hence, (†) tells us that $w = 1^j2^k = 1^{k+n}2^k = 1^k1^n2^k \in \Pi_{\mathsf{C}}$.

□

Lemma ES6.1.6
$X \subseteq \Pi_{\mathsf{A}}$.

Proof. Suppose $w \in X$. Thus $w = 0^i1^j2^k$ for some $i, j, k \in \mathbb{N}$ such that $i \neq j$ or $j \neq k$. There are two cases to consider.

- Suppose $i \neq j$. Thus $0^i1^j \in Y \subseteq \Pi_{\mathsf{B}}$, by Lemma ES6.1.4, and $2^k \in \{2\}^* \subseteq \Pi_{\langle 2\rangle}$, by Lemma ES6.1.3. Hence, because $\mathsf{A} \rightarrow \mathsf{B}\langle 2\rangle \in P$, we have that $w = (0^i1^j)2^k \in \Pi_{\mathsf{A}}$.

- Suppose $j \neq k$. Thus $0^i \in \{0\}^* \subseteq \Pi_{\langle 0\rangle}$, by Lemma ES6.1.1, and $1^j2^k \in Z \subseteq \Pi_{\mathsf{C}}$, by Lemma ES6.1.5. Hence, because $\mathsf{A} \rightarrow \langle 0\rangle\mathsf{C} \in P$, we have that $w = 0^i(1^j2^k) \in \Pi_{\mathsf{A}}$.

□

Lemma ES6.1.7
(A) $\Pi_{\mathsf{A}} \subseteq X$.

(B) $\Pi_{\mathsf{B}} \subseteq Y$.

(C) $\Pi_{\mathsf{C}} \subseteq Z$.

(D) $\Pi_{\langle 0\rangle} \subseteq \{0\}^*$.

(E) $\Pi_{\langle 1\rangle} \subseteq \{1\}^*$.

(F) $\Pi_{\langle 2\rangle} \subseteq \{2\}^*$.

Proof. It will suffice to show that:

(A) For all $w \in \Pi_{\mathsf{A}}$, $w \in X$.

(B) For all $w \in \Pi_{\mathsf{B}}$, $w \in Y$.

(C) For all $w \in \Pi_{\mathsf{C}}$, $w \in Z$.

(D) For all $w \in \Pi_{\langle 0 \rangle}$, $w \in \{0\}^*$.

(E) For all $w \in \Pi_{\langle 1 \rangle}$, $w \in \{1\}^*$.

(F) For all $w \in \Pi_{\langle 2 \rangle}$, $w \in \{2\}^*$.

We proceed by induction on Π. There are 14 productions to consider.

- $(\mathsf{A} \to \mathsf{B}\langle 2 \rangle)$ Suppose $x \in \Pi_{\mathsf{B}}$ and $y \in \Pi_{\langle 2 \rangle}$, and assume the inductive hypothesis: $x \in Y$ and $y \in \{2\}^*$. Hence $x = 0^i1^j$ for some $i, j \in \mathbb{N}$ such that $i \neq j$, and $y = 2^k$ for some $k \in \mathbb{N}$. Thus $xy = 0^i1^j2^k \in X$.
- $(\mathsf{A} \to \langle 0 \rangle\mathsf{C})$ Suppose $x \in \Pi_{\langle 0 \rangle}$ and $y \in \Pi_{\mathsf{C}}$, and assume the inductive hypothesis: $x \in \{0\}^*$ and $y \in Z$. Hence $x = 0^i$ for some $i \in \mathbb{N}$, and $y = 1^j2^k$ for some $j, k \in \mathbb{N}$ such that $j \neq k$. Thus $xy = 0^i1^j2^k \in X$.
- $(\mathsf{B} \to 0\langle 0 \rangle)$ Suppose $w \in \Pi_{\langle 0 \rangle}$, and assume the inductive hypothesis: $w \in \{0\}^*$. Thus $w = 0^i$ for some $i \in \mathbb{N}$. Hence $i + 1 \neq 0$, so that $0w = 00^i = 0^{i+1} = 0^{i+1}\% = 0^{i+1}1^0 \in Y$.
- $(\mathsf{B} \to 1\langle 1 \rangle)$ Suppose $w \in \Pi_{\langle 1 \rangle}$, and assume the inductive hypothesis: $w \in \{1\}^*$. Thus $w = 1^j$ for some $j \in \mathbb{N}$. Hence $0 \neq j + 1$, so that $1w = 11^j = 1^{j+1} = \%1^{j+1} = 0^01^{j+1} \in Y$.
- $(\mathsf{B} \to 0\mathsf{B}1)$ Suppose $w \in \Pi_{\mathsf{B}}$, and assume the inductive hypothesis: $w \in Y$. Thus $w = 0^i1^j$ for some $i, j \in \mathbb{N}$ such that $i \neq j$. Hence $i + 1 \neq j + 1$, so that $0w1 = 00^i1^j1 = 0^{i+1}1^{j+1} \in Y$.
- $(\mathsf{C} \to 1\langle 1 \rangle)$ Suppose $w \in \Pi_{\langle 1 \rangle}$, and assume the inductive hypothesis: $w \in \{1\}^*$. Thus $w = 1^j$ for some $j \in \mathbb{N}$. Hence $j + 1 \neq 0$, so that $1w = 11^j = 1^{j+1} = 1^{j+1}\% = 1^{j+1}2^0 \in Z$.
- $(\mathsf{C} \to 2\langle 2 \rangle)$ Suppose $w \in \Pi_{\langle 2 \rangle}$, and assume the inductive hypothesis: $w \in \{2\}^*$. Thus $w = 2^k$ for some $k \in \mathbb{N}$. Hence $0 \neq k + 1$, so that $2w = 22^k = 2^{k+1} = \%2^{k+1} = 1^02^{k+1} \in Z$.
- $(\mathsf{C} \to 1\mathsf{C}2)$ Suppose $w \in \Pi_{\mathsf{C}}$, and assume the inductive hypothesis: $w \in Z$. Thus $w = 1^j2^k$ for some $j, k \in \mathbb{N}$ such that $j \neq k$. Hence $j+1 \neq k+1$, so that $1w2 = 11^j2^k2 = 1^{j+1}2^{k+1} \in Z$.
- $(\langle 0 \rangle \to \%)$ We have that $\% \in \{0\}^*$.
- $(\langle 0 \rangle \to 0\langle 0 \rangle)$ Suppose $w \in \Pi_{\langle 0 \rangle}$, and assume the inductive hypothesis: $w \in \{0\}^*$. Thus $w = 0^i$ for some $i \in \mathbb{N}$. Hence $0w = 00^i = 0^{i+1} \in \{0\}^*$.
- $(\langle 1 \rangle \to \%)$ We have that $\% \in \{1\}^*$.
- $(\langle 1 \rangle \to 1\langle 1 \rangle)$ Suppose $w \in \Pi_{\langle 1 \rangle}$, and assume the inductive hypothesis: $w \in \{1\}^*$. Thus $w = 1^j$ for some $j \in \mathbb{N}$. Hence $1w = 11^j = 1^{j+1} \in \{1\}^*$.
- $(\langle 2 \rangle \to \%)$ We have that $\% \in \{2\}^*$.
- $(\langle 2 \rangle \to 2\langle 2 \rangle)$ Suppose $w \in \Pi_{\langle 2 \rangle}$, and assume the inductive hypothesis: $w \in \{2\}^*$. Thus $w = 2^k$ for some $k \in \mathbb{N}$. Hence $2w = 22^k = 2^{k+1} \in \{2\}^*$.

□

By Lemmas ES6.1.1, ES6.1.2, ES6.1.3, ES6.1.4, ES6.1.5, ES6.1.6 and ES6.1.7, we have that $L(G) = \Pi_{\mathsf{A}} = X$, $\Pi_{\mathsf{B}} = Y$, $\Pi_{\mathsf{C}} = Z$, $\Pi_{\langle 0 \rangle} = \{0\}^*$, $\Pi_{\langle 1 \rangle} = \{1\}^*$ and $\Pi_{\langle 2 \rangle} = \{2\}^*$.

(e) We will show that X is not regular. Suppose, toward a contradiction, that X is regular. Let $Y = \{\, 0^n1^n2^n \mid n \in \mathbb{N}\,\}$. In Section 4.10, we proved that Y is not context-free. Because the regular languages are a subset of the context-free languages, it follows that Y is not regular.

Lemma ES6.1.8
$Y = (\{0\}^*\{1\}^*\{2\}^*) - X$.

Proof. We will show that $Y \subseteq (\{0\}^*\{1\}^*\{2\}^*) - X \subseteq Y$.

- ($Y \subseteq (\{0\}^*\{1\}^*\{2\}^*) - X$) Suppose $w \in Y$, so that $w = 0^n1^n2^n$ for some $n \in \mathbb{N}$. Thus $w \in \{0\}^*\{1\}^*\{2\}^*$.

  Suppose, toward a contradiction, that $w \in X$. Hence $w = 0^i1^j2^k$ for some $i, j, k \in \mathbb{N}$ such that $i \neq j$ or $j \neq k$. Because $0^n1^n2^n = w = 0^i1^j2^k$, it follows that $i = j = k = n$. Thus $n \neq n$—contradiction. Hence $w \notin X$.

  Because $w \in \{0\}^*\{1\}^*\{2\}^*$ and $w \notin X$, we have that $w \in (\{0\}^*\{1\}^*\{2\}^*) - X$.

- ($(\{0\}^*\{1\}^*\{2\}^*) - X \subseteq Y$) Suppose $w \in (\{0\}^*\{1\}^*\{2\}^*) - X$. Hence $w \in \{0\}^*\{1\}^*\{2\}^*$ and $w \notin X$. Because $w \in \{0\}^*\{1\}^*\{2\}^*$, we have that $w = 0^i1^j2^k$ for some $i, j, k \in \mathbb{N}$.

  Suppose, toward a contradiction, that $i \neq j$. Then $w = 0^i1^j2^k \in X$—contradiction. Thus $i = j$.

  Suppose, toward a contradiction, that $j \neq k$. Then $w = 0^i1^j2^k \in X$—contradiction. Thus $j = k$.

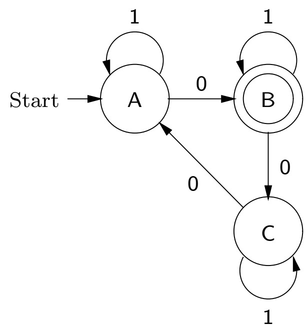
  Because $w = 0^i1^j2^k$ and $i = j = k$, we have that $w = 0^i1^i2^i$, so that $w \in Y$.

□

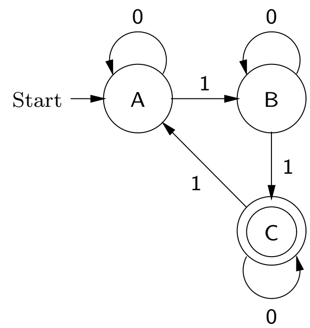
Because $L(0^*1^*2^*) = \{0\}^*\{1\}^*\{2\}^*$, we have that $\{0\}^*\{1\}^*\{2\}^*$ is regular. Thus, since X is regular, and the regular languages are closed under set difference, we have that $(\{0\}^*\{1\}^*\{2\}^*) - X$ is regular. Hence, by Lemma ES6.1.8, it follows that $Y = (\{0\}^*\{1\}^*\{2\}^*) - X$ is regular—contradiction. Thus X is not regular.

Exercise 2

(a) Let the DFAs M_1 and M_2 be:



(M_1)



(M_2)

We have that

$$L(M_1) = \{\, w \in \{0,1\}^* \mid \mathbf{zeros}\, w \,\mathbf{mod}\, 3 = 1 \,\},$$
$$L(M_2) = \{\, w \in \{0,1\}^* \mid \mathbf{ones}\, w \,\mathbf{mod}\, 3 = 2 \,\}.$$

We put the Forlan description

```
{states} A, B, C {start state} A {accepting states} B
{transitions}
A, 0 -> B; B, 0 -> C; C, 0 -> A;
A, 1 -> A; B, 1 -> B; C, 1 -> C
```

of M_1 in the file `es6-ex2-dfa1`, and the Forlan description

```
{states} A, B, C {start state} A {accepting states} C
{transitions}
A, 1 -> B; B, 1 -> C; C, 1 -> A;
A, 0 -> A; B, 0 -> B; C, 0 -> C
```

of M_2 in the file `es6-ex2-dfa2`.

We put the Forlan code

```
val efaToDFA  = nfaToDFA o efaToNFA;
val regToEFA  = faToEFA o regToFA;
val regToDFA  = efaToDFA o regToEFA;
val faToDFA   = efaToDFA o faToEFA;
val minAndRen = DFA.renameStatesCanonically o DFA.minimize;

(* accepts {0, 1}* *)
val allStrDFA = minAndRen(regToDFA(Reg.fromString "(0 + 1)*"));
val allStrFA  = injDFAToFA allStrDFA;

(* accepts all nonempty elements of {0, 1}* *)
```

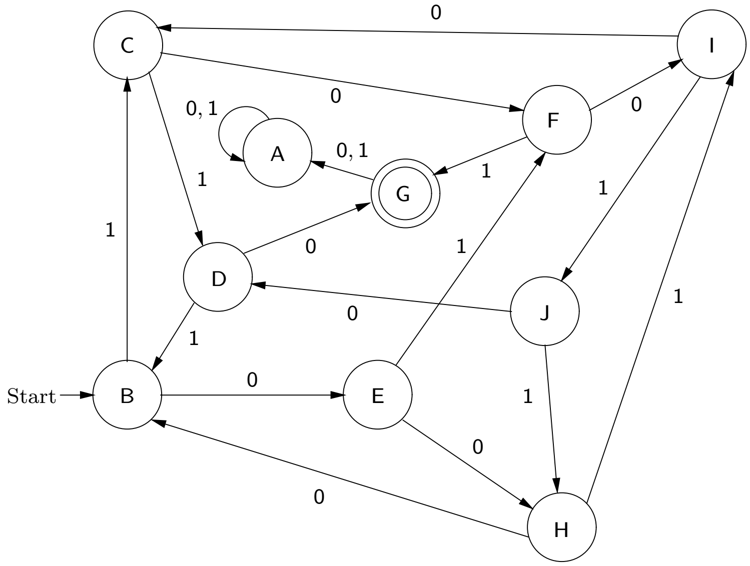
```
val allNEStrDFA = minAndRen(regToDFA(Reg.fromString "(0 + 1)(0 + 1)*"));
val allNEStrFA  = injDFAToFA allNEStrDFA;

(* M1 -- accepts elements w of {0, 1}* such that zeros w mod 3 = 1 *)
val dfa1 = DFA.input "es6-ex2-dfa1";

(* M2 -- accepts elements w of {0, 1}* such that ones w mod 3 = 2 *)
val dfa2 = DFA.input "es6-ex2-dfa2";

(* accepts X *)
val bothDFA = DFA.inter(dfa1, dfa2);

(* accepts elements of {0, 1}* with a proper prefix in X *)
val somePropPrefBothFA  = FA.concat(injDFAToFA bothDFA, allNEStrFA);
val somePropPrefBothDFA = minAndRen(faToDFA somePropPrefBothFA);

(* accepts elements of {0, 1}* with no proper prefix in X *)
val noPropPrefBothDFA = DFA.minus(allStrDFA, somePropPrefBothDFA);

(* accepts Y *)
val firstBothDFA = minAndRen(DFA.inter(bothDFA, noPropPrefBothDFA));

val dfa = firstBothDFA;
```

in the file `es6-ex2a.sml`. We then proceed as follows:

```
- use "es6-ex2a.sml";
[opening es6-ex2a.sml]
val efaToDFA = fn : efa -> dfa
val regToEFA = fn : reg -> efa
val regToDFA = fn : reg -> dfa
val faToDFA = fn : fa -> dfa
val minAndRen = fn : dfa -> dfa
val allStrDFA = - : dfa
val allStrFA = - : fa
val allNEStrDFA = - : dfa
val allNEStrFA = - : fa
val dfa1 = - : dfa
val dfa2 = - : dfa
val bothDFA = - : dfa
val somePropPrefBothFA = - : fa
val somePropPrefBothDFA = - : dfa
val noPropPrefBothDFA = - : dfa
val firstBothDFA = - : dfa
val dfa = - : dfa
val it = () : unit
- DFA.output("", dfa);
{states} A, B, C, D, E, F, G, H, I, J {start state} B {accepting states} G
{transitions}
```

```
A, 0 -> A; A, 1 -> A; B, 0 -> E; B, 1 -> C; C, 0 -> F; C, 1 -> D; D, 0 -> G;
D, 1 -> B; E, 0 -> H; E, 1 -> F; F, 0 -> I; F, 1 -> G; G, 0 -> A; G, 1 -> A;
H, 0 -> B; H, 1 -> I; I, 0 -> C; I, 1 -> J; J, 0 -> D; J, 1 -> H
val it = () : unit
```

Here is a drawing of `dfa`:



(b) We put the Forlan code

```
(* returns (n, |Y_n|) *)
fun sizeFirstBothLen n =
      let (* accepts all elements of {0, 1}* with length n *)
          val allStrLenDFA =
                minAndRen(regToDFA(Reg.power(Reg.fromString "0 + 1", n)))

          (* accepts Y_n *)
          val firstBothLenDFA =
                minAndRen(DFA.inter(firstBothDFA, allStrLenDFA))

          (* generates Y_n *)
          val firstBothLenReg =
                faToReg (SOME 1, Reg.weakSimplify)
                        (injDFAToFA firstBothLenDFA)
      in (n, Set.size(Reg.toStrSet firstBothLenReg)) end;

(* returns [n : m] *)
fun upto(n, m) = if n > m then nil else n :: upto(n + 1, m);
```

in the file `es6-ex2b.sml`. We then proceed as follows:

```
- use "es6-ex2b.sml";
```

```
[opening es6-ex2b.sml]
val sizeFirstBothLen = fn : int -> int * int
val upto = fn : int * int -> int list
val it = () : unit
- map sizeFirstBothLen (upto(0, 25));
val it =
  [(0,0),(1,0),(2,0),(3,3),(4,0),(5,0),(6,15),(7,0),(8,0),(9,75),(10,0),
   (11,0),(12,375),(13,0),(14,0),(15,1875),(16,0),(17,0),(18,9375),(19,0),
   (20,0),(21,46875),(22,0),(23,0),(24,234375),(25,0)] : (int * int) list
```

(On my moderately fast laptop, it took about one minute to compute these sizes.)