Compilers assignment #4

Rune Kok Nielsen (qkd362)

January 3, 2016

1 Task 1

First we define $s'_0 = \hat{\epsilon}(s_0) = \{1, 2, 4\}$. We can now construct the move function for s'_0.

$$move(s'_0, a) = \hat{\epsilon}(\{3, 4\} = \{3, 4, 6\} = s'_1$$
$$move(s'_0, b) = \hat{\epsilon}(\{5\}) = \{4, 5, 6\} = s'_2$$

We now construct the move function for s'_1 and s'_2

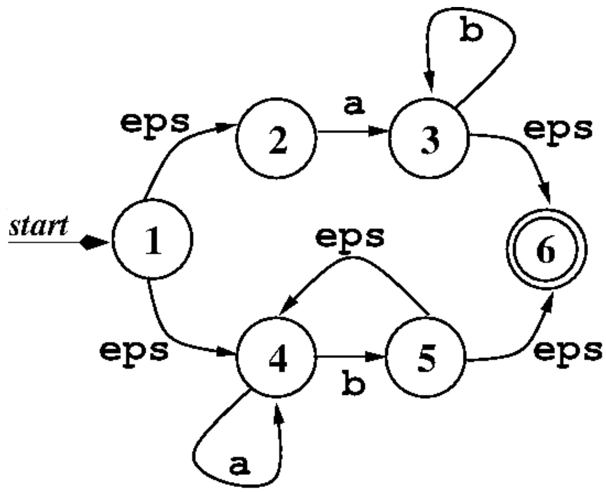
$$move(s'_1, a) = \hat{\epsilon}(\{4\}) = \{4\} = s'_3$$
$$move(s'_1, b) = \hat{\epsilon}(\{3, 5\}) = \{3, 4, 5, 6\} = s'_4$$
$$move(s'_2, a) = \hat{\epsilon}(\{4\}) = \{4\} = s'_3$$
$$move(s'_2, b) = \hat{\epsilon}(\{5\}) = \{4, 5, 6\} = s'_2$$

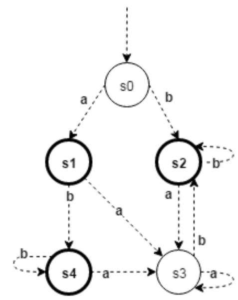
We now look at s'_3 and s'_4:

$$move(s'_3, a) = \hat{\epsilon}(\{4\}) = \{4\} = s'_3$$
$$move(s'_3, b) = \hat{\epsilon}(\{5\}) = \{4, 5, 6\} = s'_2$$
$$move(s'_4, a) = \epsilon(\hat{\{4\}}) = \{4\} = s'_3$$
$$move(s'_4, b) = \hat{\epsilon}(\{3, 5\}) = \{3, 4, 5, 6\} = s'_4$$

We have now derived all reachable FDA states in $S' = \{s'_0, s'_1, s'_2, s'_3, s'_4\}$. Of these states, s'_1, s'_2 and s'_4 contain the final state s_6, so we have $F' = \{s'_1, s'_2, s'_4\}$.

The FDA is drawn below





2 Task 2

We start by dividing into two groups: G_1 containing final states and G_2 containing non-final states.

$$G_1 = \{3\}$$
$$G_2 = \{1, 2, 4, 5, 6, 7, 8\}$$

We now select G_1 and check for consistency.

G_1	0	1
3	G_2	G_1

Naturally, G_1 is consistent as it contains only one state and so we mark it. We check G_2.

G_2	0	1
1	G_2	G_2
2	G_2	G_1
4	G_1	G_2
5	G_2	G_2
6	G_1	G_2
7	G_2	G_2
8	G_2	G_1

G_2 is not consistent so we divide it and remove the mark from G_1. We now start over with the following groups:

$$
\begin{aligned}
G_1 &= \{3\} \\
G_2 &= \{1, 5, 7\} \\
G_3 &= \{2, 8\} \\
G_4 &= \{4, 6\}
\end{aligned}
$$

G_1 is still consistent because it contains one state. Check G_2:

G_2	0	1
1	G_3	G_4
5	G_3	G_4
7	G_2	G_2

This is still not consistent so we must divide again. We now have the following groups:

$$
\begin{aligned}
G_1 &= \{3\} \\
G_2 &= \{1, 5\} \\
G_3 &= \{2, 8\} \\
G_4 &= \{4, 6\} \\
G_5 &= \{7\}
\end{aligned}
$$

We mark G_1 and G_7 as they both contain only one element. We check G_2:

G_2	0	1
1	G_3	G_4
5	G_3	G_4

G_2 is now consistent so we mark it. We now need to check G_3:

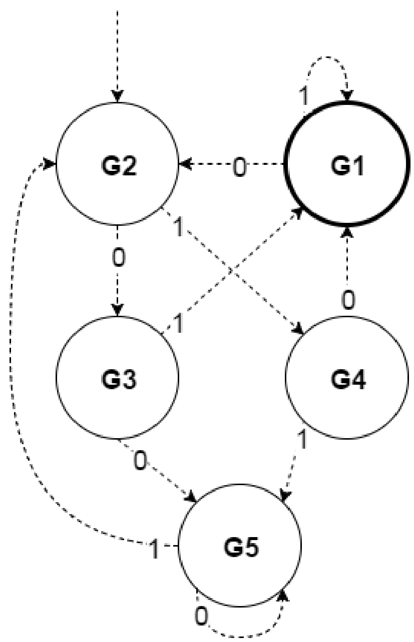
G_3	0	1
2	G_5	G_1
8	G_5	G_1

G_3 is consistent so we mark it. Lastly, we check G_4:

G_4	0	1
4	G_1	G_5
6	G_1	G_5

G_4 is consistent so we are done.

I have drawn the resulting minimized DFA below:



3 Task 3

3.1 a

3.1.1 i

This regular expression matches any number divisible by 5. I.e. any number ending with a 5 or 0. Allows leading zeroes.

$[0-9]*(5|0)$

3.1.2 ii

This regular expression matches any number with the digit 5 occuring exactly three times. This number consists of three fives with some amount (possibly 0) of none-5 digits between them, leading the first five or trailing the last five.

$([0-4]|[6-9])*5([0-4]|[6-9])*5([0-4]|[6-9])*5([0-4]|[6-9])$

3.2 b

3.2.1 i

It is not possible to have a regular expression matching exactly any number with equal amount of 1's and 2's since there are infinite such numbers and we have no variables to keep count of the number of occurences of a character.

3.2.2 ii

Assuming that we only wish to match non-negative numbers, there are only a finite amount of numbers between 0 and 1.000.000 in which the number of occurences of 1's and 2's are the same. Therefore, you could identify all these numbers and create a matching exactly these numbers. Better strategies (resulting in shorter expressions) exist but I will not delve into this as we are only concerned about whether these numbers *can* be described by regular expressions.