Section 09: Solutions

1. CFGs

Write a context-free grammar to match each of these languages.

(a) All binary strings that start with 11.
Solution:

$$\mathbf{S} \to 11\mathbf{T}$$
$$\mathbf{T} \to 1\mathbf{T} \mid 0\mathbf{T} \mid \varepsilon$$

(b) All binary strings that contain at most one 1.
Solution:

$$\mathbf{S} \to \mathbf{ABA}$$
$$\mathbf{A} \to 0\mathbf{A} \mid \varepsilon$$
$$\mathbf{B} \to 1 \mid \varepsilon$$

(c) All strings over 0, 1, 2 with the same number of 1s and 0s and exactly one 2.
Hint: Try modifying the grammar from Section 8 2c for binary strings with the same number of 1s and 0s (You may need to introduce new variables in the process).
Solution:

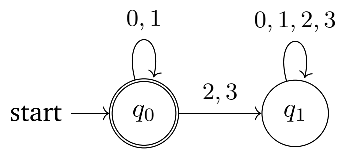
$$\mathbf{S} \to 2\mathbf{T} \mid \mathbf{T}2 \mid \mathbf{ST} \mid \mathbf{TS} \mid 0\mathbf{S}1 \mid 1\mathbf{S}0$$
$$\mathbf{T} \to \mathbf{TT} \mid 0\mathbf{T}1 \mid 1\mathbf{T}0 \mid \varepsilon$$

T is the grammar from Section 8 2c. It generates all binary strings with the same number of 1s and 0s. **S** matches a 2 at the beginning or end. The rest of the string must then match **T** since it cannot have another 2. If neither the first nor last character is a 2, then it falls into the usual cases of matching 0s and 1s, so we can mostly use the same rules as **T**. The main change is that **SS** becomes **ST** | **TS** to ensure that exactly one of the two parts contains a 2. The other change is that there is no ϵ since a 2 must appear somewhere.

2. DFAs, Stage 1

Construct DFAs to recognize each of the following languages. Let $\Sigma = \{0, 1, 2, 3\}$.

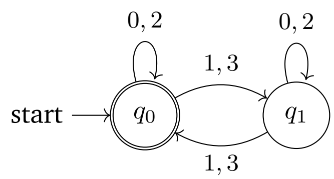
(a) All binary strings.
Solution:



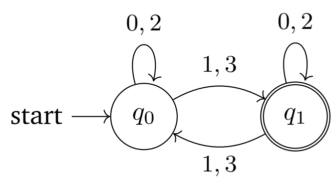
q_0: binary strings

q_1: strings that contain a character which is not 0 or 1.

(b) All strings whose digits sum to an even number.
Solution:



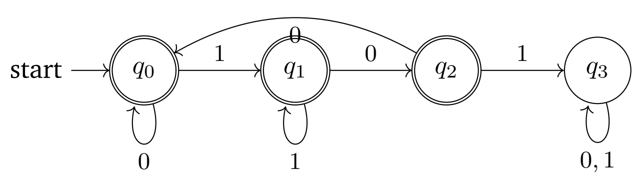
(c) All strings whose digits sum to an odd number.
Solution:



3. DFAs, Stage 2

Construct DFAs to recognize each of the following languages. Let $\Sigma = \{0, 1\}$.

(a) All strings which do not contain the substring 101.
Solution:



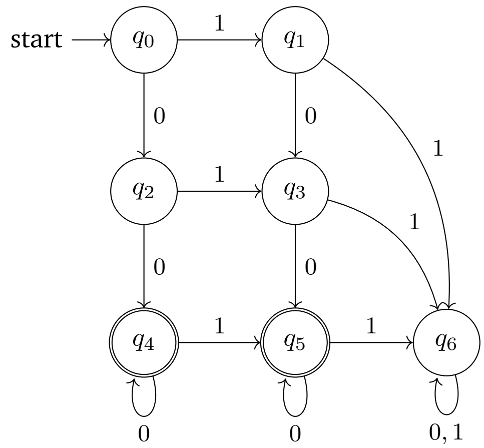
q_3: string that contain 101.

q_2: strings that don't contain 101 and end in 10.

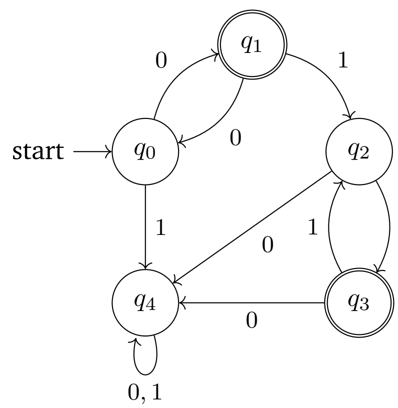
q_1: strings that don't contain 101 and end in 1.

q_0: ε, 0, strings that don't contain 101 and end in 00.

(b) All strings containing at least two 0's and at most one 1.
Solution:

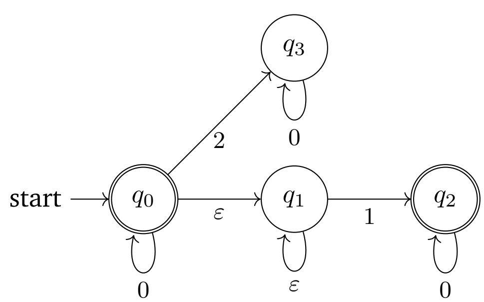


(c) All strings containing an even number of 1's and an odd number of 0's and not containing the substring 10.
Solution:



4. NFAs

(a) What language does the following NFA accept?

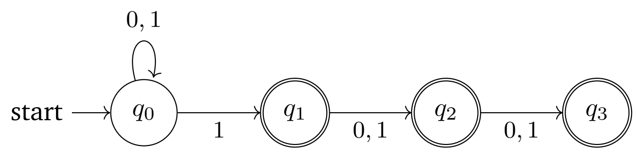


Solution:

All strings of only 0's and 1's not containing more than one 1.

(b) Create an NFA for the language "all binary strings that have a 1 as one of the last three digits".
Solution:

The following is one such NFA:

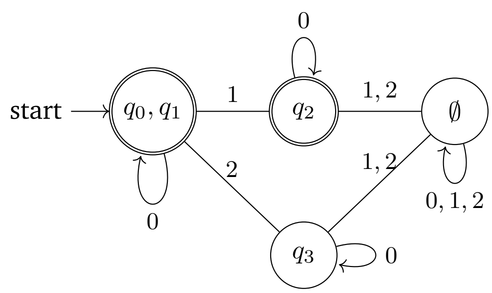


5. DFAs & Minimization

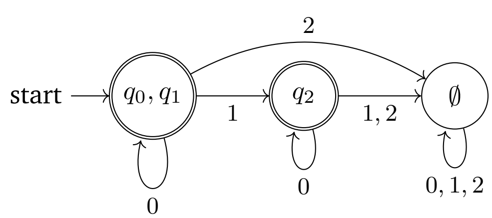
Note: We will not test you on minimization, although you may optionally read the extra slides and do this problem for fun

(a) Convert the NFA from 1a to a DFA, then minimize it.

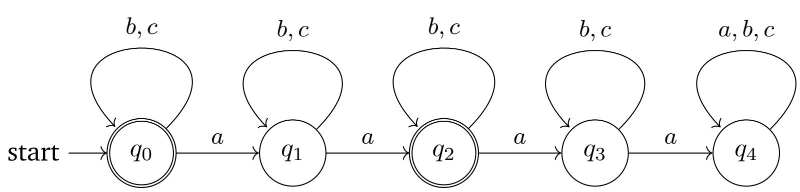
Solution:



Here is the minimized form:



(b) Minimize the following DFA:



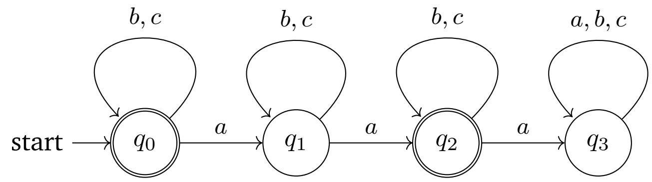
Solution:

Step 1: q_0, q_2 are final states and the rest are not final. So, we start with the initial partition with the following groups: group 1 is $\{q_0, q_2\}$ and group 2 is $\{q_1, q_3, q_4\}$.

Step 2: q_1 is sending a to group 1 while q_3, q_4 are sending a to group 2. So, we divide group 2. We get the following groups: group 1 is $\{q_0, q_2\}$, group 3 is $\{q_1\}$ and group 4 is $\{q_3, q_4\}$.

Step 3: q_0 is sending a to group 3 and q_2 is sending a to group 4. So, we divide group 1. We will have the following groups: group 3 is $\{q_1\}$, group 4 is $\{q_3, q_4\}$, group 5 is $\{q_0\}$ and group 6 is $\{q_2\}$.

The minimized DFA is the following:



6. Onto & One-to-One

Give an example of a function $f : \mathbb{N} \to \mathbb{N}$ which is onto but not one-to-one. Be specific.
Solution:

Let $f(n) = \lfloor \frac{n}{2} \rfloor$. Then f is onto. But f isn't one-to-one because (for example) both 0 and 1 are mapped onto 0.

7. Proving Onto

(a) Suppose that $f : \mathbb{Z}^2 \to \mathbb{Z}$ is defined by $f(x, y) = xy + yx^2 - x^2$. Prove that f is onto, where $\mathbb{Z}^2 = \mathbb{Z}$ x $\mathbb{Z}$ which represents all possible ordered pairs of integers.
Solution:

Notice that $f(x, y) = xy + (y - 1)x^2$.

Let p be an arbitrary integer. We need to find a pre-image (the pre-image of a value is the set of all input values (or elements) that map to that particular value under the function) for p.

Consider $m = (p, 1)$. m is an element of $\mathbb{Z}$ x $\mathbb{Z} = \mathbb{Z}^2$. We can compute

$$f(m) = p \cdot 1 + (1 - 1)p^2 = p + 0 \cdot p^2 = p$$

So m is a pre-image of p.

Since p was arbitrary and we found a pre-image for an arbitrarily chosen integer, f is onto.

(b) Suppose that A and B are sets. Suppose that $f : B \to A$ and $g : A \to B$ are functions such that $f(g(x)) = x$ for every $x \in A$. Prove that f is onto. **Solution:**

Let m be an arbitrary element of A. We need to find a pre-image for m.

Consider $n = g(m)$. n is an element of B. Furthermore, since $f(g(x)) = x$ for every $x \in A$, we have

$$f(n) = f(g(m)) = m.$$

So n is a pre-image of m.

Since m was arbitrary and we found a pre-image for an arbitrarily chosen element of A, f is onto.