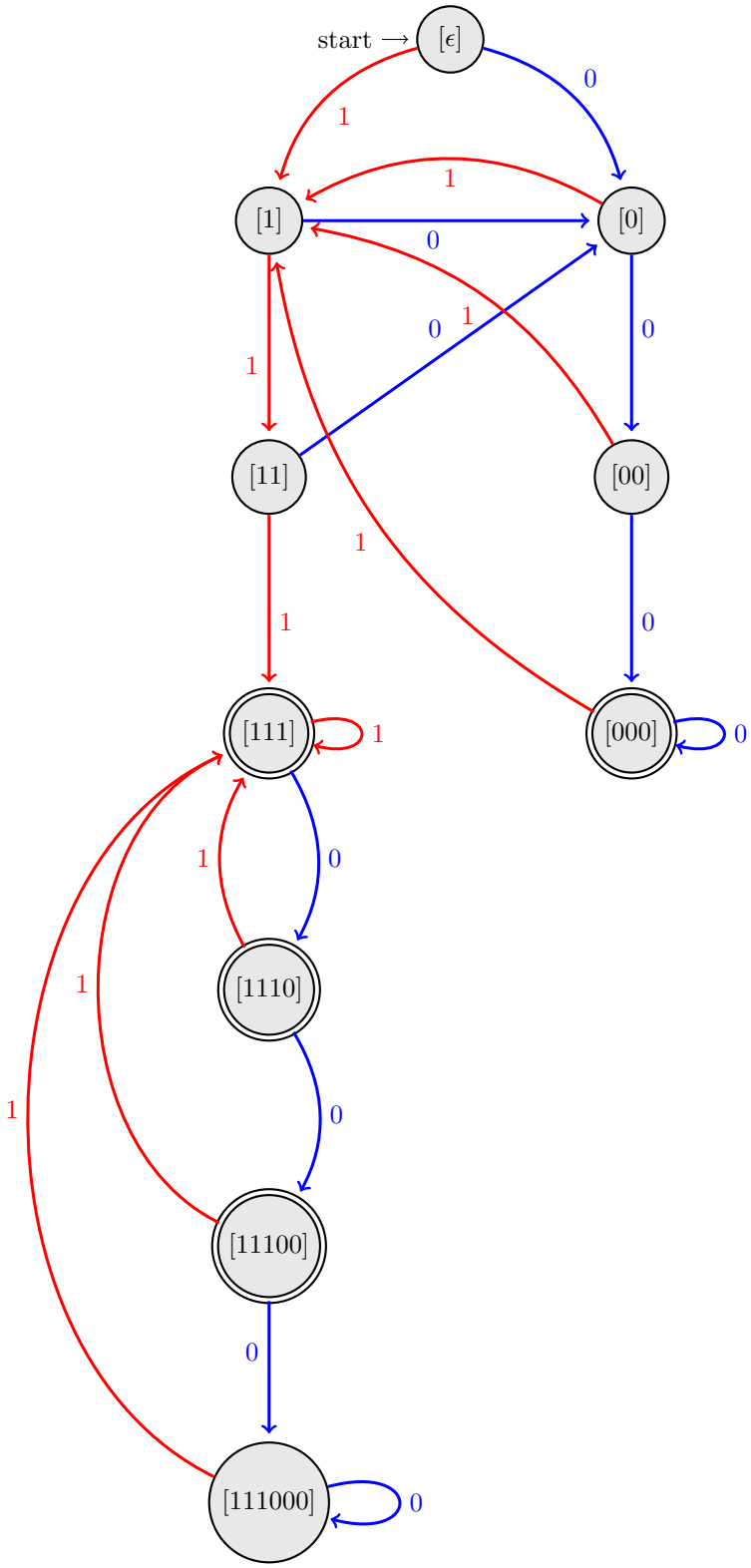
**ComS 331** **Spring 2025** **Name:** Mobina Amrollahi

**HW 3 Due: Feb 16$^{th}$ 2025**

1. Draw a DFA, simplified to the best of your abilities, that recognizes the language:

$$L = \{w \in \{0,1\}^* \mid w \text{ ends with } 000 \text{ or contains } 111, \text{ but not both}\}$$

**Answer**



2. Consider the $n$-bit binary representation of a natural number $x$:

$$(x_{n-1}x_{n-2}\dots x_1x_0)_2 \iff x = \sum_{i=0}^{n-1} x_i 2^i$$

where each bit $x_i$ is a binary digit, either 0 or 1. For example, $(00000101)_2$ is the 8-bit binary representation of the number 5 since: $0{\cdot}2^7{+}0{\cdot}2^6{+}0{\cdot}2^5{+}0{\cdot}2^4{+}0{\cdot}2^3{+}1{\cdot}2^2{+}0{\cdot}2^1{+}1{\cdot}2^0 = 4{+}1 = 5$. This is the format normally employed by digital computers to store nonnegative integers. Now, consider the language:

$$\begin{aligned} L = &\{a_0b_0c_0\dots a_{n-1}b_{n-1}c_{n-1} \mid n \in \mathbb{N} \wedge \forall i, 0 \le i < n, a_i \in \{0,1\}, b_i \in \{0,1\}, c_i \in \{0,1\} \wedge \\ &(a_{n-1}\dots a_0)_2 + (b_{n-1}\dots b_0)_2 = (c_{n-1}\dots c_0)_2\} \end{aligned}$$

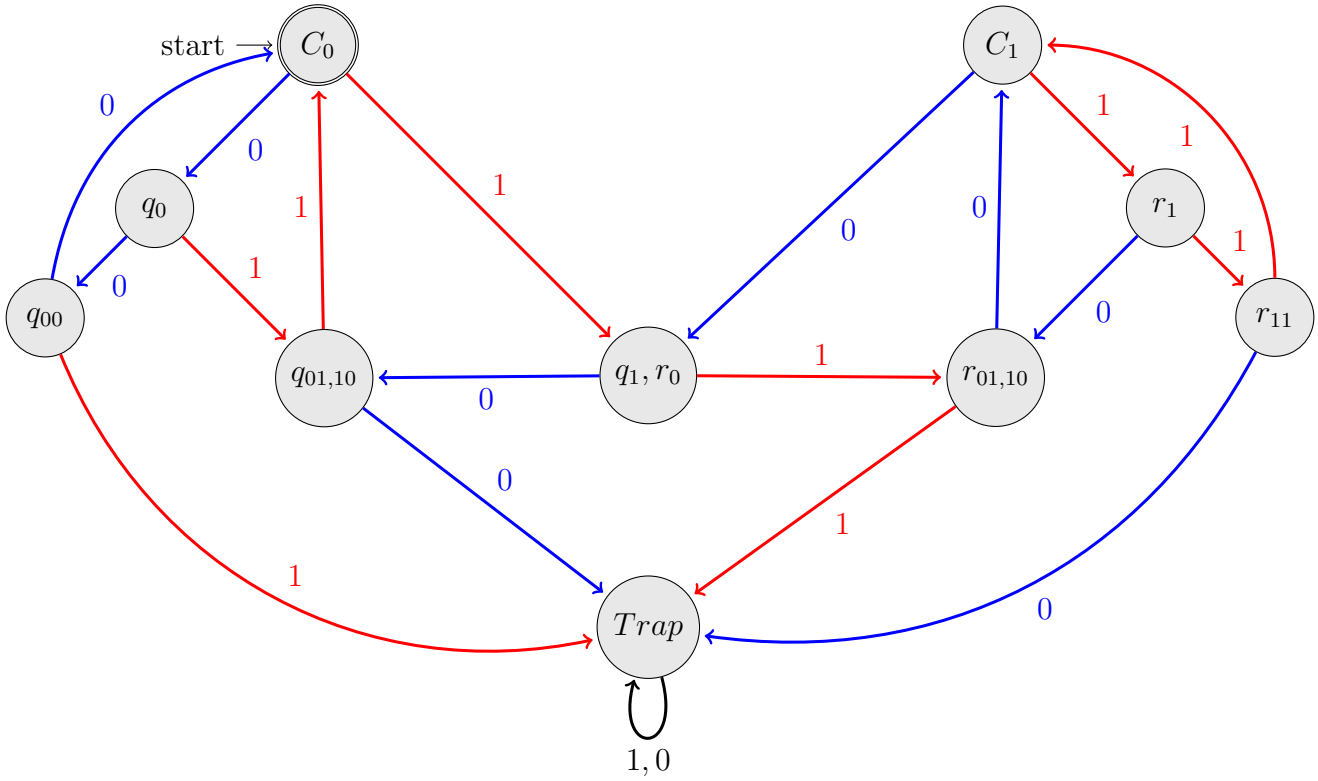
For example, since $5 + 3 = 8, 5 = (000101)_2, 3 = (000011)_2$, and $8 = (001000)_2$ then:

$$110 \quad 010 \quad 100 \quad 001 \quad 000 \quad 000 \in L$$

(the string is spaced every three digits for readability's sake only).

Draw a DFA that recognizes $L$.

**Answer**




3. Precisely describe in English the language accepted by the following NFA:

(Diagram Placeholder for NFA)

Then, give a regular expression for it.

**Answer**

The language accepted by this NFA consists of all binary strings with at least a length of four such

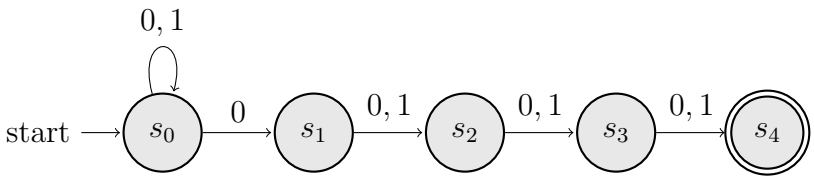
that if they start with 0, there must be a run of three consecutive 0s, and if they start with 1, there must be a run of three consecutive 1s somewhere in the string.
The corresponding regular expression is for this language is:

$$0(0+1)^*000(0+1)^* + 1(0+1)^*111(0+1)^*$$

4. Describe in a short English sentence the language accepted by the following DFA and give a regular expression for it (hint: the "names" of the states reflect their meaning). Then, draw a 5-state NFA that accepts the same language.
**Answer**
The DFA accepts the set of all binary strings that are at least four bits long, and the last four bits of the string must begin with 0.

The regular expression that this language is assessed by is:

$$(1+0)^*0(0+1)(0+1)(0+1)$$

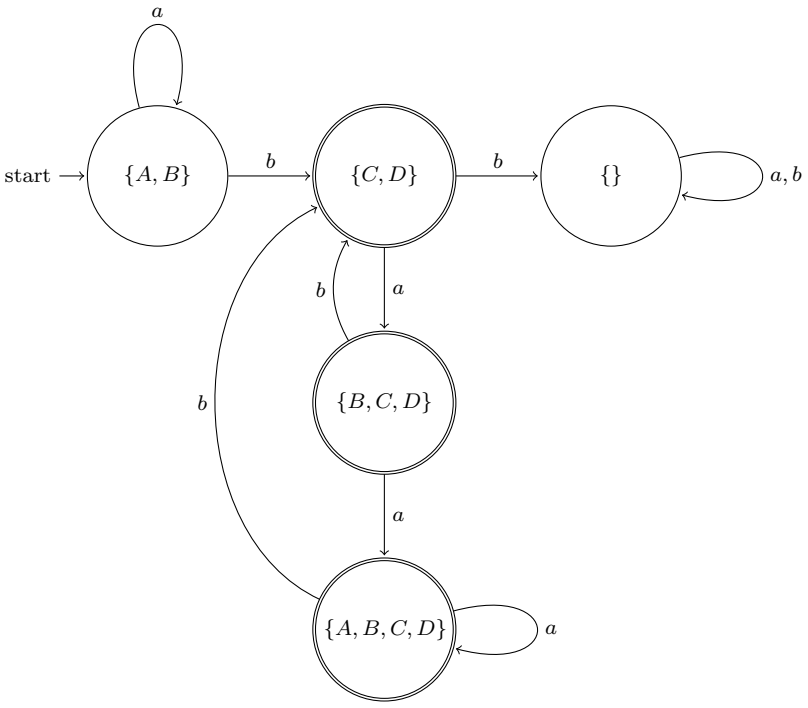



5. Consider the following NFA $M$:

(Diagram Placeholder for NFA)

- Using the algorithm described in class, derive an equivalent (non-minimized) DFA $M'$.
**Answer.** The initial stay in the DFA will be the set of the initial states in NFA and all the states we can go to from such state with $\epsilon$. Hence $s'_0 = \{A, B\}$.

| State | on a | on b |
|---|---|---|
| $\{A, B\}$ | $\{A, B\}$ | $\{C, D\}$ |
| $\{C, D\}$ | $\{B, C, D\}$ | $\{\}$ |
| $\{B, C, D\}$ | $\{A, B, C, D\}$ | $\{C, D\}$ |
| $\{A, B, C, D\}$ | $\{A, B, C, D\}$ | $\{C, D\}$ |




{} is the trap state. In case not readable, $\{C, D\}$, $\{B, C, D\}$, and $\{A, B, C, D\}$ are all final states since they have at least one of $C$ or $D$ in their set.

- Minimize $M'$ to obtain the minimized DFA $M''$.

  **Answer**

  Table 1 illustrates the process of distinguishing states. For each pair of states, we check whether reading strings $a$ and $b$ from them leads to distinguishable states. If they do, the entry is marked with F (False). Otherwise, it remains blank until the final step.

  We start with distinguishing final and non-final states. Hence, each element of $\{\{A, B\}, \{\}\}$ is distinguishable from each element of $\{\{C, D\}, \{B, C, D\}, \{A, B, C, D\}\}$.

  Then, by reading string $b$ from $\{A, B\}$ and $\{\}$ we'll end up in states $\{C, D\}$ and $\{\}$ respectively, which are distinguishable. Hence two states of $\{A, B\}$ and $\{\}$ are distinguishable.

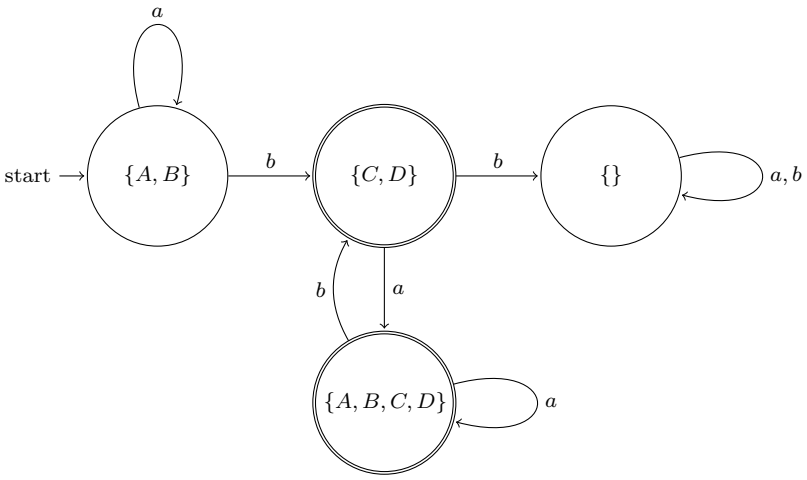
  Further, by reading string $b$ from $\{C, D\}$ and $\{A, B, C, D\}$ we'll end up in states $\{\}$ and $\{C, D\}$ respectively, which are distinguishable. Hence two states of $\{C, D\}$ and $\{A, B, C, D\}$ are distinguishable. Using the same argument, two states of $\{C, D\}$ and $\{B, C, D\}$ are distinguishable.

  Finally, the only pair left are two states of $\{B, C, D\}$ and $\{A, B, C, D\}$ which by reading string $a$ from both we'll end up in states $\{A, B, C, D\}$ and by reading string $b$ from both we'll end up in states $\{C, D\}$. Hence, these two states are not distinguishable, and can be merged.

Table 1: Non-Distinguishable Table: F if they can be distinguished by a string, T otherwise.

| | | | | | |
|---|---|---|---|---|---|
| {C,D} | $F$ | | | | |
| $\{\}$ | $F$ | $F$ | | | |
| $\{B, C, D\}$ | $F$ | $F$ | $F$ | | |
| $\{A, B, C, D\}$ | $F$ | $F$ | $F$ | $T$ | |
| | $\{A, B\}$ | $\{C, D\}$ | $\{\}$ | $\{B, C, D\}$ | |

Therefore, the minimized DFA $M''$ is:




- Describe in English, as succinctly as possible, the essential characteristics of the language accepted by these automata.

  **Answer**

  The regular expression for this language is:

  $$a^*b(a^+(b+\epsilon))^*$$

  Which is the language with having at least one b and no run of two or more b's.