1. If $\{a_n\}$ and $\{b_n\}$ are two sequences, we write $\{a_n\} = \{b_n\}$ if and only if $a_n = b_n, n \in \mathbb{N}_0$. Use mathematical induction to show that $\{a_n\} = \{b_n\}$ where
$$a_0 = 2, \ a_n = a_{n-1} + 3; \ b_n = 3n + 2$$

2. Are the following sets closed under the following operations? Justify your answer. If not, what are their respective closures?
   a. The odd length strings over the alphabet $\{\clubsuit, \blacklozenge, \heartsuit, \spadesuit\}$ under concatenation.
   b. $L = \{w \in \{\alpha,\beta\}^* : w \text{ ends in } \beta\}$ under Kleene star.

3. Let
$$\textbf{\textit{SameColour}} = \{\, (apricot, orange), (avocado, lime), (lime, cucumber),$$
$$(strawberry, tomato), (carrot, orange) \,\}$$
   be a relation defined over the set of fruits and vegetables
$$FV = \{\text{carrot, avocado, lime, tomato, pear, apricot, strawberry, orange, cucumber}\}$$
   [Assume the colour of the given fruits are green, orange/yellow or red]
   a. Which properties of an equivalence relation does **SameColour** lack?
   b. What is its equivalence closure?
   c. Represent the closure in graphical form.
   d. How many equivalence classes are there in the closure?

4. For each of the following statements, state whether it is *True* or *False*. Prove your answer.
   a. $\forall L_1, L_2$ where $L_1 \neq L_2 \ ((L_1 \ L_2) \neq (L_2 \ L_1))$.
   b. $\forall L_1, L_2 \ (L_1 = L_2 \text{ iff } L_1^+ = L_2^+)$.
   c. If $L_1 - L_2$ is finite, then at least one of L_1 or L_2 must also be finite.

5. Construct a deterministic finite state machine for each of the following languages. Give both the state-transition diagram and the quintuple representation for each.
   a. $\{w \in \{0, 1\}^* : w \text{ has } 101 \text{ as a substring}\}$.
   b. $\{w \in \{a, b\}^* : w \text{ has neither } bab \text{ nor } abb \text{ as a substring}\}$.

**6.** Consider the following deterministic finite state machine M
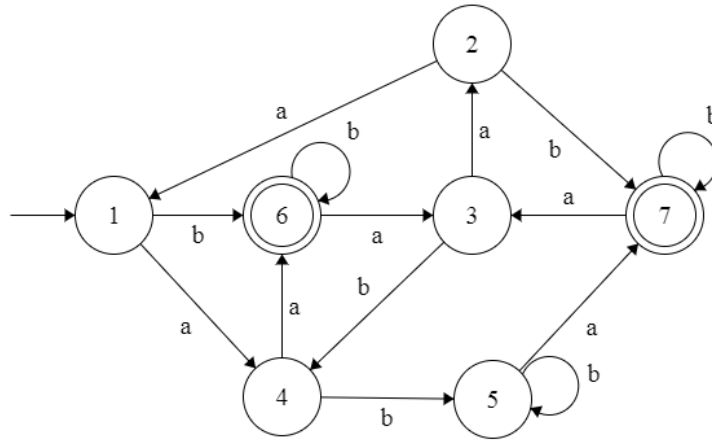
Does it contain the minimum number of required states for the target language? Prove your answer using *minDFSM* (Rich, 2008; page 92) or, as an extra challenge, by finding the number of equivalence classes in L(M).

7. Give nondeterministic finite automata that accept each of the following languages. Provide both state-transition diagrams and the corresponding quintuple representations (M = (K, Σ, Δ, s, F)).

    a. The set of strings in (a∪b)* such that there are two a's separated by a string whose length is 4*i*, for some *i*≥0.

    b. L={ w {a,b}*; w= (ba ∪ ((a ∪ bb) a b*)) }

8. Let *M* be the NDFSM below. Construct a DFSM that accepts ¬*L*(*M*). Show your work using the algorithm *ndfmtodfsm* (Rich 2008, page 75).
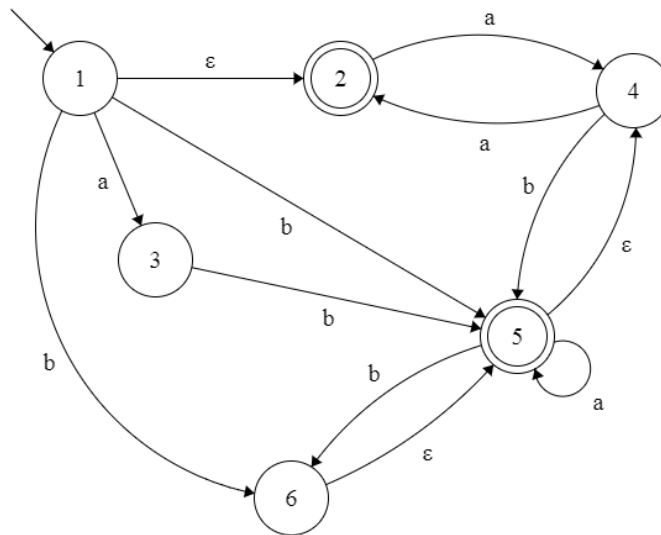
9. Convert the following FSMs to regular expressions. For part a) you MUST use the full algorithm *fsmtoregex* shown in class (Rich, 2008; page 142-143) and show your work including null transitions. For part b) you may use the heuristic. In both cases consider carefully which state to remove first; picking a certain state may drastically simplify the rest of the solution.

a.
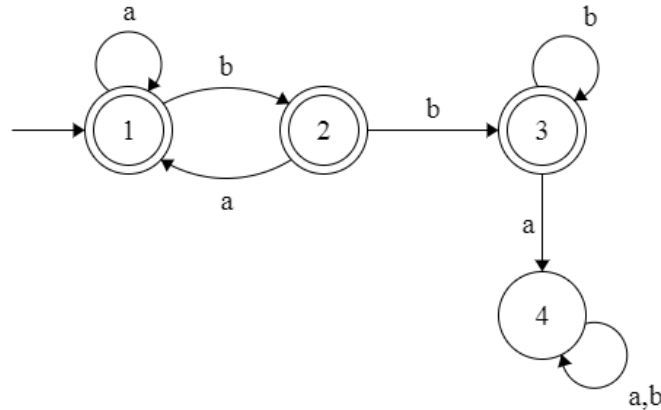
b.
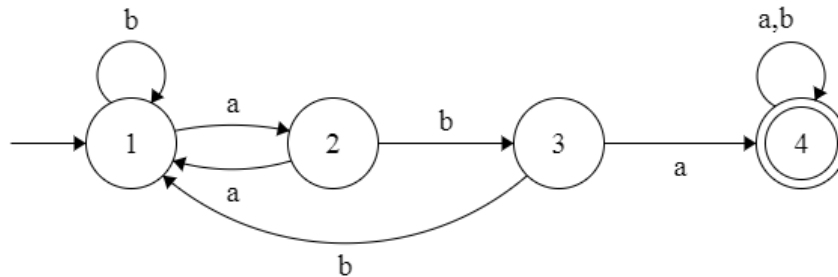
10. Consider a Dice Rolling game played between two players P and Q. Each player rolls a dice and the player with highest face value wins the roll. The player wins 3 rolls wins the game.

a. Define an alphabet Σ and describe a technique for encoding the Dice Rolling game as strings over Σ.

b. Let L_{DR} be the language of Dice Rolling game, encoded as strings as described in part (a), that corresponds to wins for player P. Show a FSM that accepts L_{DR}. [Note: you only need to consider how player P can win]

11. Consider a game Magic-Cheese-Hunt. The game characters are a mouse and a number of cats living in a house. The mouse lives in a mouse hole where the game starts from; it is hungry and wanders around the house searching for cheese. There are two types of cheeses: ordinary cheese and magic cheese. Ordinary cheese only satisfies his hunger and magic cheese satisfies his hunger and makes him invisible, however, he can eat only one kind of cheese at a time. If he finds some chees he eats it and get back to one of the mouse holes

in the house. The cats in the house hunt for the mouse. As soon as the visible mouse sees a cat it should flee until there is no longer any cat around and then continue what it was doing. After the mouse gets back to its hole it becomes hungry again. Represent the behaviour of the mouse using a finite state machine. Your FSM should contain all necessary states and all possible state transitions required to match the above rules of playing Magic-Cheese-Hunt. The transitions in your FSM should be based on what the mouse sees or gets: C (sees a cat), NC (does not see any cat), CH (gets some ordinary cheese), MC (gets some magic cheese).

Marking guide

Question	Part	Marks
1		5
2	a	4
	b	4
3	a	2
	b	2
	c	2
	d	2
4	a	5
	b	5
	c	5
5	a	10
	b	10
6		15
7	a	10
	b	10
8		15
9	a	10
	b	10
10	a	4
	b	10
11		10
Total		150

REFERENCES

[1] Elaine Rich, Automata Computability and Complexity: Theory and Applications, Pearson, Prentice Hall, 2008.