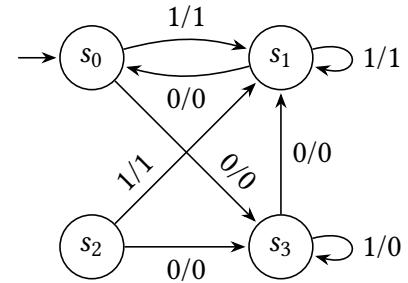
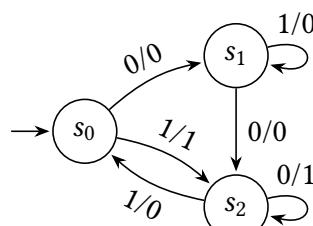
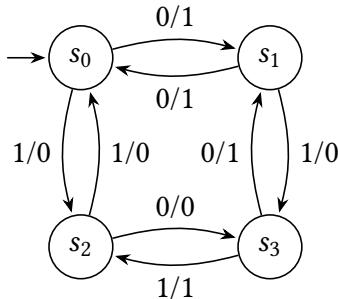


1. For each given regular expression P , construct a DFA (Deterministic Finite Automaton), and find the number of accepted word of length at most 5, i.e. the size of the set $\mathcal{L}' = \{w \in \mathcal{L}(P) \mid |w| \leq 5\}$. For “any” (.) and “negative” ($[\cdot]$) matches, assume that the alphabet is $\Sigma = \{a, b, c, d\}$.

(a) $P_1 = ab^*$	(c) $P_3 = [\cdot]cd + c\{3\}$	(e) $P_5 = d(a bc)^*$
(b) $P_2 = a+b?c$	(d) $P_4 = [\cdot]a (\cdot ddd)?$	(f) $P_6 = ((a ab)[cd])\{2\}$
2. Describe the set of strings defined by each of these sets of productions in EBNF² (extended Backus-Naur form).

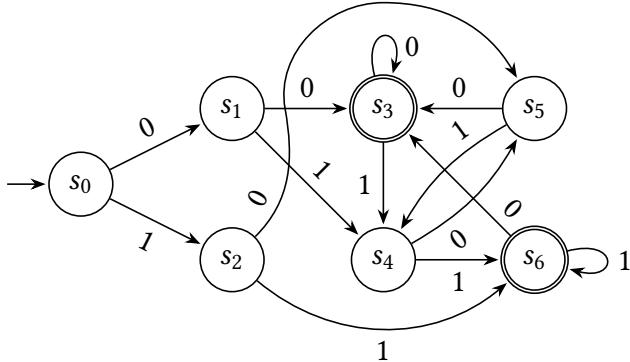
(a) $\langle string \rangle ::= \langle L \rangle + \langle D \rangle? \langle L \rangle +$ $\langle L \rangle ::= a \mid b \mid c$ $\langle D \rangle ::= 0 \mid 1$	(c) $\langle string \rangle ::= \langle L \rangle^* (\langle D \rangle^+)? \langle L \rangle^*$ $\langle L \rangle ::= x \mid y$ $\langle D \rangle ::= 0 \mid 1$
(b) $\langle string \rangle ::= \langle sign \rangle? \langle N \rangle$ $\langle sign \rangle ::= '+' \mid '-'$ $\langle N \rangle ::= \langle D \rangle (\langle D \rangle \mid 0)^*$ $\langle D \rangle ::= 1 \mid \dots \mid 9$	(d) $\langle string \rangle ::= \langle C \rangle \langle R \rangle^*$ $\langle C \rangle ::= a \mid \dots \mid z \mid A \mid \dots \mid Z$ $\langle D \rangle ::= 0 \mid \dots \mid 9$ $\langle R \rangle ::= \langle C \rangle \mid \langle D \rangle \mid '_'$
3. Let $\mathcal{G} = \langle V, T, S, P \rangle$ be the phrase-structure grammar with vocabulary $V = \{A, S\}$, terminal symbols $T = \{0, 1\}$, start symbol $S = S$, and set of productions P : $S \rightarrow 1S$, $S \rightarrow 00A$, $A \rightarrow 0A$, $A \rightarrow 0$.
 - (a) Show that 111000 belongs to the language generated by \mathcal{G} .
 - (b) Show that 11001 does not belong to the language generated by \mathcal{G} .
 - (c) What is the language generated by \mathcal{G} ?
4. Find the output generated from the input string 01110 for each of the following Mealy machines.



5. Construct a Moore machine for each of the following descriptions.
 - (a) Determine the residue modulo 3 of the input treated as a binary number. For example, for input ϵ (which corresponds to “value” 0) the residue is 0; 101 (5 in decimal) has residue 2; and 1010 (value 10) has residue 1.
 - (b) Output the residue modulo 5 of the input from $\{0, 1, 2\}^*$ treated as a ternary (base 3) number.
 - (c) Output A if the binary input ends with 101; output B if it ends with 110; otherwise output C .
6. Show that regular languages are *closed* under the following operations.
 - (a) Union, that is, if L_1 and L_2 are regular languages, then $L_1 \cup L_2$ is also regular.
 - (b) Concatenation, that is, if L_1 and L_2 are regular languages, then $L_1 \cdot L_2$ is also regular.
 - (c) Kleene star, that is, if L is a regular language, then L^* is also regular.
 - (d) Complement, that is, if L is a regular language, then $\bar{L} = \Sigma^* - L$ is also regular.
 - (e) Intersection, that is, if L_1 and L_2 are regular languages, then $L_1 \cap L_2$ is also regular.

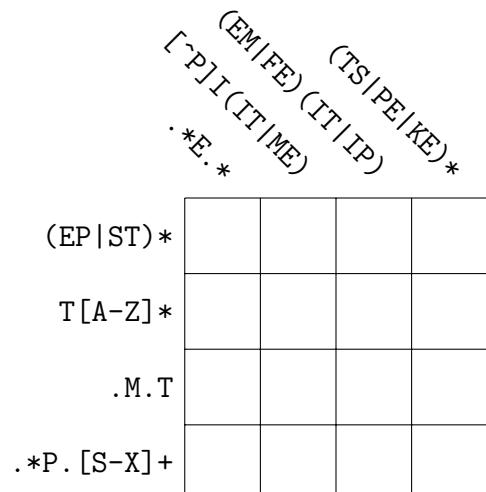
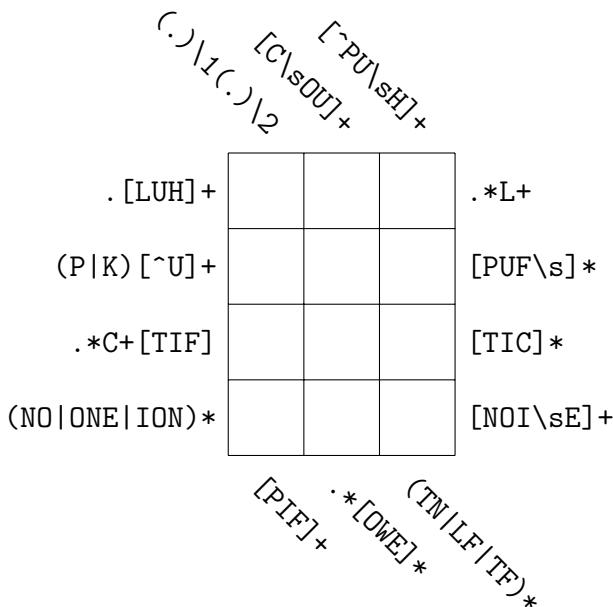
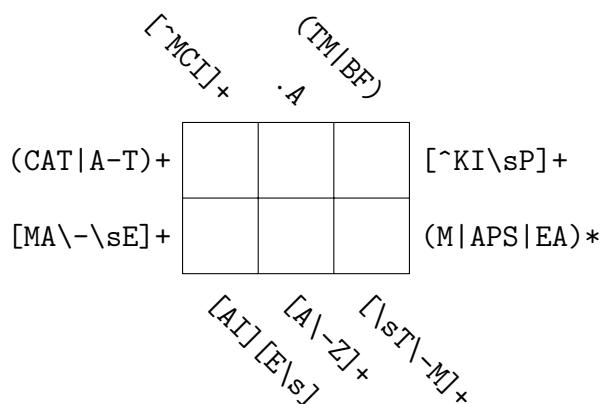
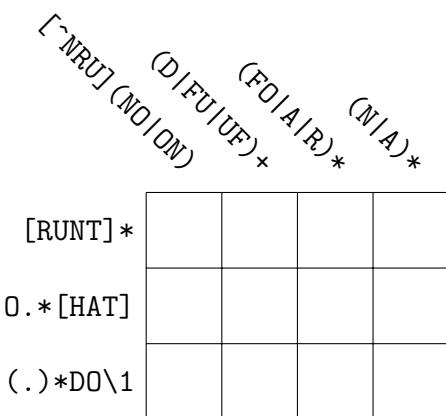
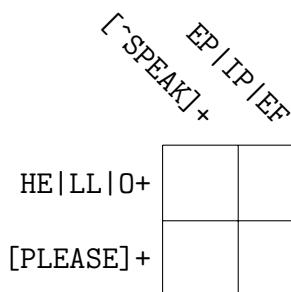
7. Determine whether the following languages are regular or not. For non-regular languages, use Pumping lemma to prove that they are not regular. For each regular language, provide a regular expression and construct an ε -NFA.
- $L_1 = \{w \in \{0, 1\}^* \mid \text{length of } w \text{ is odd}\}$
 - $L_2 = \{0^n 1^n \mid n \in \mathbb{N}\}$
 - $L_3 = \{w \in \{0, 1\}^* \mid w \text{ contains an even number of 1s}\}$
 - $L_4 = \{1^{n^2} \mid n \in \mathbb{N}\}$
8. Consider a finite-state automaton $M = (\Sigma, Q, q_0, F, \delta)$ and a non-negative integer k . Let R_k be the relation on the set of states of M such that $s R_k t$ if and only if for every input string $w \in \Sigma^*$ with $|w| \leq k$, $\delta(s, w)$ and $\delta(t, w)$ are both final states or both not final states. Furthermore, let R^* be the relation on the set of states of M such that $s R^* t$ if and only if for every input string $w \in \Sigma^*$, regardless of length, $\delta(s, w)$ and $\delta(t, w)$ are both final states or both not final states.
- Show that for every nonnegative integer k , R_k is an equivalence relation on S .
Two states s and t are called k -equivalent if $s R_k t$.
 - Show that R^* is an equivalence relation on S .
Two states s and t are called $*$ -equivalent if $s R^* t$.
 - Show that if two states s and t are k -equivalent ($k > 0$), then they are also $(k - 1)$ -equivalent.
 - Show that the equivalence classes of R_k are a *refinement* of the equivalence classes of R_{k-1} .
 - Show that if two states s and t are k -equivalent for every non-negative integer k , then they are $*$ -equivalent.
 - Show that all states in a given R^* -equivalence class are final or all are not final.
 - Show that if two states s and t are $*$ -equivalent, then $\delta(s, a)$ and $\delta(t, a)$ are also $*$ -equivalent for all $a \in \Sigma$.

9. Consider the finite-state automaton $M = (\Sigma, Q, q_0, F, \delta)$ depicted below.



- Find the k -equivalence classes of M for $k = 0, 1, 2, 3$.
- Find the $*$ -equivalence classes of M .
- Construct the quotient automaton \overline{M} of M .
 - The quotient automaton \overline{M} of the deterministic finite-state automaton $M = (\Sigma, S, s_0, F, \delta)$ is the finite state automaton $\overline{M} = (\Sigma, \overline{S}, [s_0]_{R^*}, \overline{F}, \overline{\delta})$, where the set of states \overline{S} is the set of R^* -equivalence classes of S ; the transition function $\overline{\delta}$ is defined by $\overline{\delta}([s]_{R^*}, a) = [\delta(s, a)]_{R^*}$ for all states $[s]_{R^*}$ of \overline{M} and input symbols $a \in \Sigma$; and \overline{F} is the set consisting of R^* -equivalence classes of final states of M .

10. Solve the following regex crosswords¹. Fill each cell with a single ASCII character (an uppercase letter, a digit, a punctuation mark, or a space). Each row/column, when read left to right or top to bottom must match the regular expression(s) given for that row/column.



¹ Credits: <https://regexcrossword.com>