Table 6.5 Comparison of the differences between FA, PDA and TM $\,$

Issues	FA	PDA	TM
Label of the transitions in the state diagram	One symbol - input symbol Input symbol q ₀ q Q	Three symbols - input, pop and push symbols Input Pop Push symbol symbol symbol	Three symbols - read, write and move direction symbols States & Transitions Read Write Move Left or Right q ₁ a/b L q ₂
Transition functions	 δ(q₁, a) = q₂ It means that with input a, move from state q₀ to state q₂ The first two (q₁, a) are input. The last one (q₂) is output. (q₁, a) = q₂ current state new state current input symbol 	$\delta\left(q_{1},a,b\right)=\left\{\left(q_{2},w\right)\right\} \text{ or }\\ \delta\left(q_{1},a,b\right)=\left\{\left(q_{2},w\right),\left(q_{3},w\right)\right\}$ • It means that - From state q_{1} . - Read a from the tape, - Pop the string b from the stack, - To state q_{2} , - Push string w onto the stack. • The first three $\left(q_{1},a,b\right)$ are input. • The last two $\left(q_{2},w\right)$ are output. $\left(q_{1},a,b\right)=\left\{\left(q_{2},w\right)\right\}$ $\left(q_{2},a,b\right)=\left\{\left(q_{2},w\right)\right\}$ $\left(q_{2},a,b\right)=\left(q_{2},w\right)$ $\left(q_{2},a,b\right)=$	$\delta\left(q_{1},a\right)=\left\{\left(q_{2},b,R\right)\right\}$ • It means that $-\text{ From state }q_{1}.$ $-\text{ Read }a\text{ from the tape,}$ $-\text{ Write }b\text{ to the tape,}$ $-\text{ To state }q_{2},$ $-\text{ Move to right of the tape.}$ • The first two (q_{1},a) are input. • The last three (q_{2},b,R) are output. $(q_{1},a)=\left\{\left(q_{2},b,R\right)\right\}$ $\left(q_{1},a\right)=\left\{\left(q_{2},b,R\right)\right\}$ $\left(q_{2},b,R\right)$ $\left(q_{3},a\right)=\left\{\left(q_{2},b,R\right)\right\}$ $\left(q_{3},a\right)=\left\{\left(q_{2},b,R\right)\right\}$ write symbol new state
Configurations / Instantaneous Description	 Represented with the ordered pair [q_i, s] where q_i ∈ Q is the machine's current state; s ⊆ w and w ∈ Σ* is the remaining unprocessed input. 	 Represented with a triple (p, w, α) ∈ (K, Σ*, Γ*) where p is the current state w is the remaining input α is the current stack contents 	Denoted uq_ivB where B is blank symbol, all tape positions, to the right of the B are blanks and uv is the string spelled by the symbols on the tape from the left-hand boundary to the B . $a_1a_2 \qquad q_1 \qquad a_3 \qquad a_2a_1$ $beguence \qquad a_1a_2 \qquad q_1 \qquad a_1a_2 \qquad q_1 \qquad a_1a_2 \qquad q_1a_3 \qquad a_2a_1$ $a_1a_2 \qquad q_1 \qquad a_1a_2 \qquad q_1a_3 \qquad a_2a_1$ $a_1a_2 \qquad q_1 \qquad a_1a_2 \qquad q_1a_3 \qquad a_2a_1$ $a_1a_2 \qquad q_1 \qquad a_1a_2 \qquad q_1a_3 \qquad a_2a_1$ $a_1a_2 \qquad q_1a_3 \qquad a_2a_1$
Computations	The FA M ₁ : a a, b q ₂ b q ₂ The computations of M with input strings abba is $[q_0, abba] \models [q_0, bba]$ $\models [q_1, ba]$ $\models [q_2, a]$ $\models [q_2, \lambda]. accepts$	The PDA M_2 : $a, \lambda/A \\ b, \lambda/B$ $b, B/\lambda$ The computation of M_2 with input $abcba$ is $[q_0, abcba, \lambda] \vdash [q_0, bcba, A]$ $\vdash [q_0, cba, BA]$ $\vdash [q_1, ba, BA]$ $\vdash [q_1, a, A]$ $\vdash [q_1, \lambda, \lambda]$. accepts	The Turing machine M_3 : b/blk a/alk q_2 a/alk accepts the language $(a \cup b)*aa(a \cup b)*$. The computation of M_3 with input $aabb$ is $q_0BaabbB \mid Bq_1aabbB \mid Baq_2abbB \mid Baq_3bbB$. accepts
Determinism	Both deterministic (DFA) and non-deterministic (NFA) • Every state of DFA always has exactly one exiting transition arrow for each symbol in the alphabet while the NFA may has more. • In a DFA, labels on the transition arrows are from the	Non-deterministic only PDAs are non-deterministic. Allowed non-deterministic transitions - Multiple transitions on same pop/input, transitions may but do not have to push or pop.	Turing machine are deterministic. No lambda transitions allowed.

Formal definitions (how many member of the tuple? What they are?)	alphabet while NFA can have an arrow with the label ε . A FA is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$. Formal Definition Finite Automaton (FA) $M = (Q, \Sigma, \delta, q_0, F)$ States Final states Input alphabet Transition Initial states	A PDA is a 6-tuple (Q, Σ , Γ , δ , q_0 , F). Formal Definition Pushdown Automaton (PDA) $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ States Input alphabet Stack alphabet Transition function Initial states	A TM is a 7-tuple (Q, Σ , Γ , B , δ , q_0 , F). Turing Machine: Input alphabet M = (Q, Σ , Γ , δ , q_0 , δ , F) Transition Initial states Γ - gomma, δ - delta
Acceptance criteria	FA accepts w if the machine end up in a final state.	PDA accepts w if the machine end up in a final state with an empty stack.	A string is accepted by final state if the computation halts in a final state, but the TM need not read the entire input string to accept the string.
Example of state diagram for the language aa^*	$\rightarrow q_0$ a q_1	a, ε/ε q ₀ a, ε/ε q ₁	$ \xrightarrow{a/a, R} $