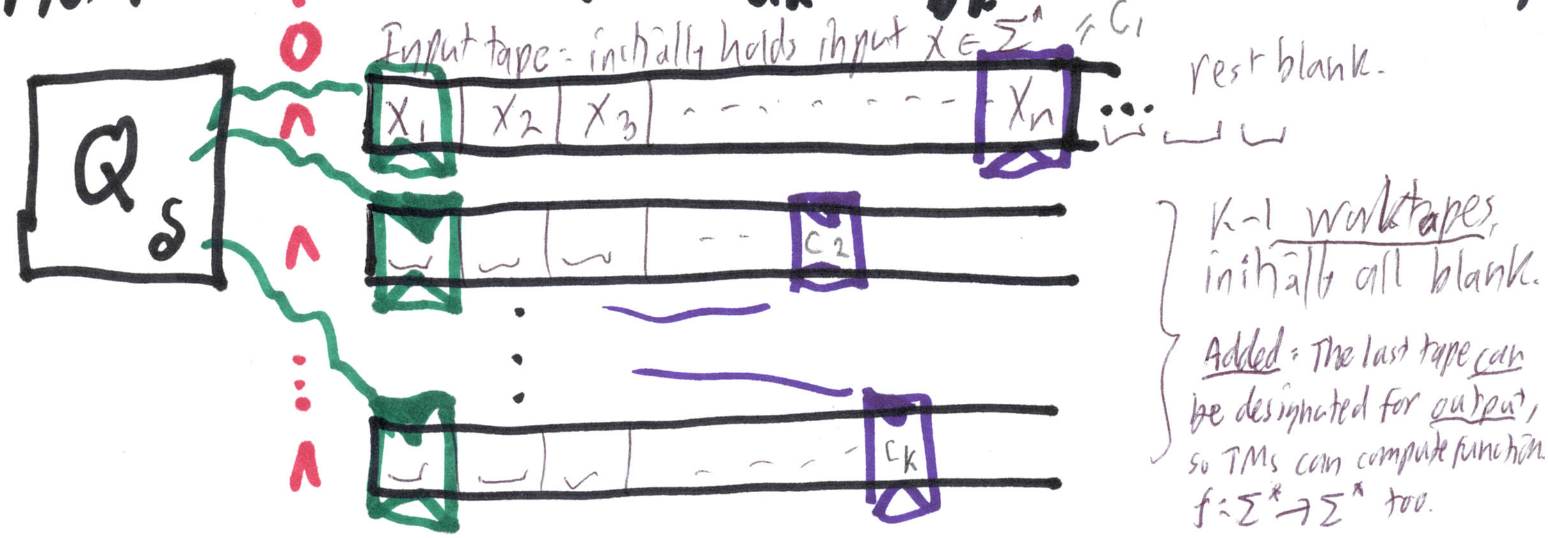
TopHat 1063

<u>Defn</u>: A <u>K</u>-tape TM ($K \geq 1$) has the same components $M = (Q, \Sigma, \Gamma, \delta, \sqcup, s, F)$

but $\delta \subseteq (Q \times \Gamma^K) \times (\Gamma^K \times \{L, R, S\}^K \times Q)$

Typical Instruction: $\left(p, \begin{matrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{matrix} \middle/ \begin{matrix} d_1 \\ d_2 \\ \vdots \\ d_K \end{matrix}, \begin{matrix} D_1 \\ D_2 \\ \vdots \\ D_K \end{matrix}, q\right)$ can draw $p \longrightarrow q$ arcs similarly

Picture.



The initial ID <u>$I_0(x)$</u> has all heads in column **1** with the first tape head scanning the first bit x_1 of x (or scanning a blank if $x = \epsilon$) and all other tapes are blank.

Defn of $I \vdash_M J$, $I \vdash_M^* K$, accepting ID, and $L(M)$: <u>same</u>

<u>Alternate convention</u>: Γ includes a dedicated "∧" char. Each tape has ∧ in cell 0. Code in δ always moves R off ∧. Never changes ∧, never moves L off ∧

For PDAs, ∧ acts as a bottom of stack marker.

- With multitape TMs, often the input tape[1] is <u>read-only</u>: in δ, always $d_1 = c_1$

Further, it is often <u>one-way</u>: (in δ, always $D_1 \neq L$, which essentially subsumes <u>RO</u> :)

- A tape j <u>behaves like a stack</u> if $D_j = L \Rightarrow d_j = \sqcup$ in all ca

<u>Def^n</u>: A pushdown automaton (PDA: DPDA det^c; NPDA nondet^c) <u>is</u> (equivalent to) a 2-tape TM whose input tape is one-way and whose sole worktape behaves as a stack

The one operational quirk is that shifting between push and pop steps needs one extra machine step. → Demo.

Proof of CFG-to-NPDA is nicest if you put G into ChN... A → c becomes $q \circlearrowleft \begin{pmatrix} c & c & R \\ A & \sqcup & L \end{pmatrix}$ Read c, pop A. All action is one state q... A → BC becomes: "Pop A, push <u>C</u>, then push B and go back to ...

The proof of NPDA-to-CFG is in text @2... but is ugly, skip

<u>Theorem</u>: Given any CFG G, we can build an <u>N</u>PDA N s.t. $L(N) = L(G)$, and <u>vice-versa</u>.

But, PAL and EVENPAL (without # markers) are examples of CFLs that have an NPDA but not a DPDA. (Proof of that fact is <u>hard</u>, not in text)

<u>Def^n</u>: A language A is a <u>DCFL</u> if there is a <u>D</u>PDA M such that $L(M) = A$.

Section 2.4 talks about "deterministic CFGs" but the ones it gives are not equivalent.

$\text{REG} \subsetneq \text{DCFL} \subsetneq \text{CFL} \subsetneq$ <u>decidable</u> languages. — Ch 4. next Tue.