Ambiguity/growth of tree automata/transducers made easy via MSO queries

Lê Thành Dũng (Tito) Nguyễn — *nltd@nguyentito.eu* — ÉNS Lyon
joint work with Paul Gallot (Bremen) & Nathan Lhote (Aix–Marseille)

Séminaire LX, LaBRI, Bordeaux — 8 février 2023

What is an interesting class of finite-state computable functions?

Regular languages ($L \subseteq \Sigma^*$): a robust notion

deterministic finite automata $\iff$ nondeterministic FA $\iff$ two-way FA $\iff$ regular expressions $\iff$ monadic second-order logic (MSO) $\iff$ ...

What about *functions* $f \colon \Sigma^* \to \Gamma^*$? $\rightsquigarrow$ consider *transducers*: automata with output

What is an interesting class of finite-state computable functions?

Regular languages ($L \subseteq \Sigma^*$): a robust notion

deterministic finite automata $\iff$ nondeterministic FA $\iff$ two-way FA $\iff$ regular expressions $\iff$ monadic second-order logic (MSO) $\iff$ ...

What about *functions* $f\colon \Sigma^* \to \Gamma^*$? $\rightsquigarrow$ consider *transducers*: automata with output

Some equivalences don't hold anymore, e.g. DFT $\subsetneq$ NFT! Several usual classes:

- Linear growth: $|f(w)| = O(|w|)$ for $f\colon \Sigma^* \to \Gamma^*$ rational (NFT) / regular (MSO)
- Or hyperexponential growth (L-systems, iterated pushdown transducers, ...)

What is an interesting class of finite-state computable functions?

Regular languages ($L \subseteq \Sigma^*$): a robust notion

deterministic finite automata $\iff$ nondeterministic FA $\iff$ two-way FA $\iff$ regular expressions $\iff$ monadic second-order logic (MSO) $\iff$ ...

What about *functions* $f\colon \Sigma^* \to \Gamma^*$? $\rightsquigarrow$ consider *transducers*: automata with output

Some equivalences don't hold anymore, e.g. DFT $\subsetneq$ NFT! Several usual classes:

- Linear growth: $|f(w)| = O(|w|)$ for $f\colon \Sigma^* \to \Gamma^*$ rational (NFT) / regular (MSO)
- Or hyperexponential growth (L-systems, iterated pushdown transducers, ...)

Complexity theory: feasible $= \mathsf{P}$. What is the finite-state counterpart?

Proposal (Bojańczyk 2018): polyregular functions

Robust class of string functions, computed by *pebble transducers* (early 2000s)

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$

$\triangleright$	a	b	c	$\#$	b	a	c	$\#$	c	b	$\triangleleft$

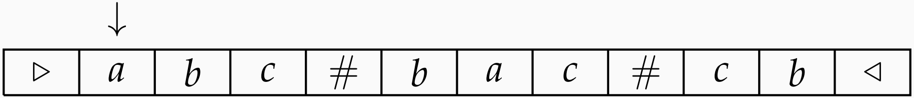
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



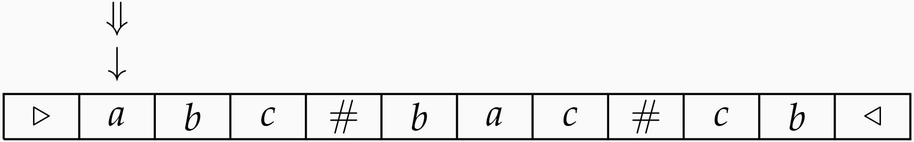
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



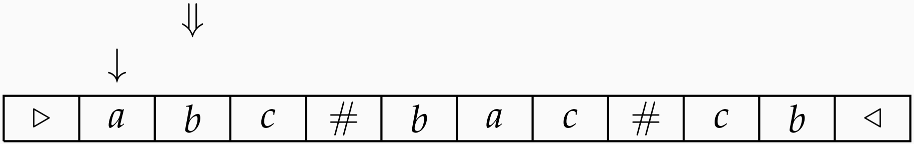
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



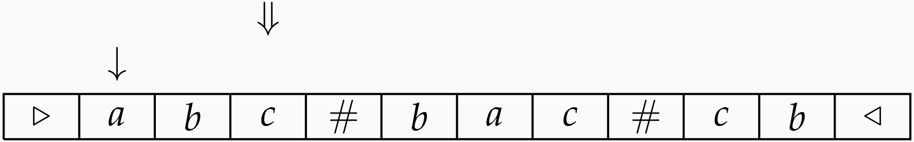
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



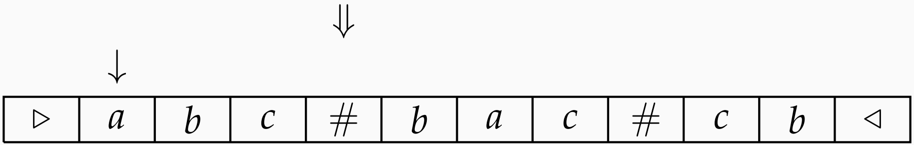
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



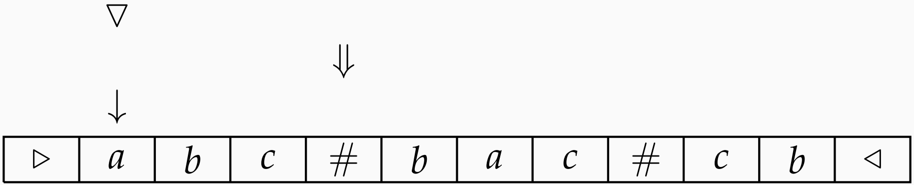
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



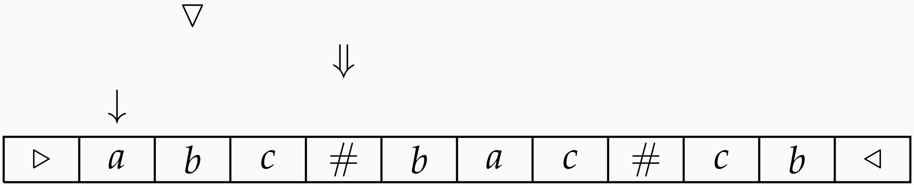
Output:

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



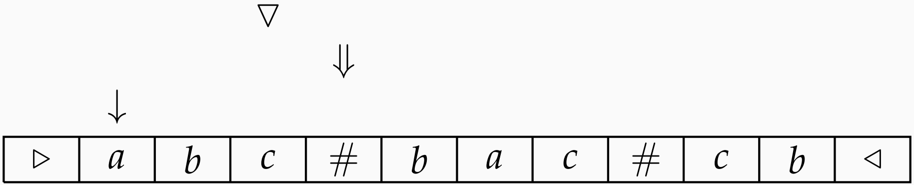
Output: a

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



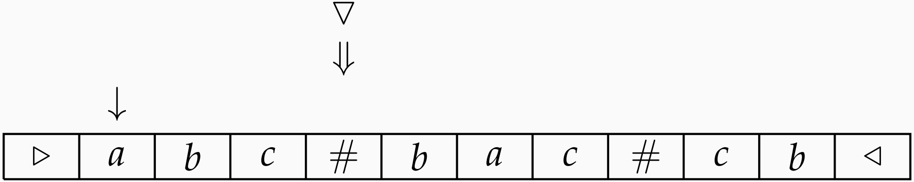
Output: *ab*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>***k*-pebble transducers**</u> $(k \geq 1)$

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\dots\#w_n \longmapsto (w_0)^n\#\dots\#(w_n)^n$



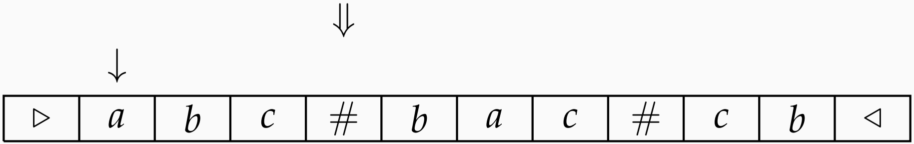
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



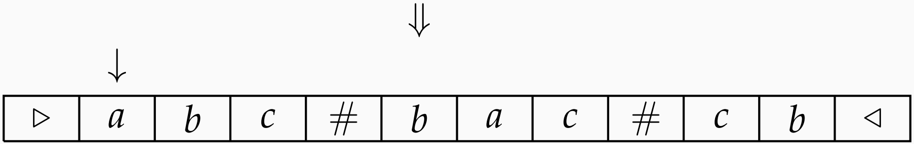
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$

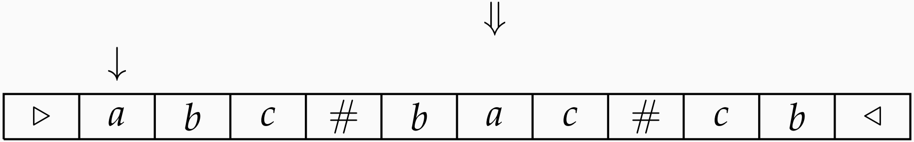


Output: *abc*

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



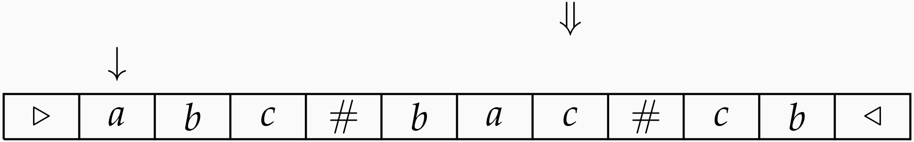
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



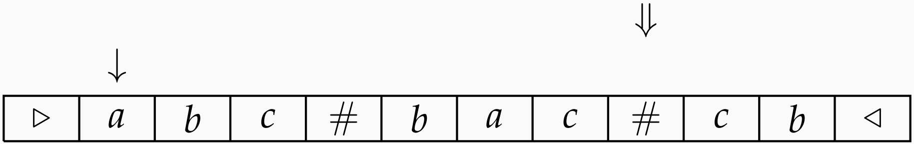
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



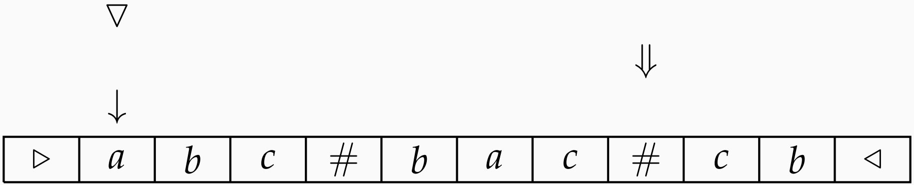
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



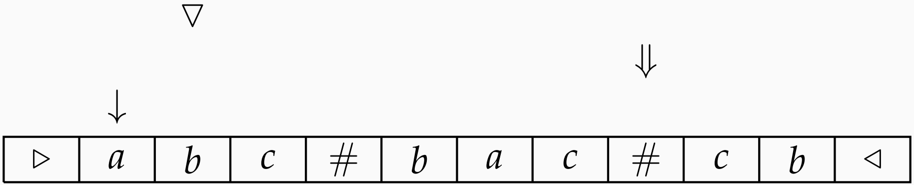
Output: *abc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



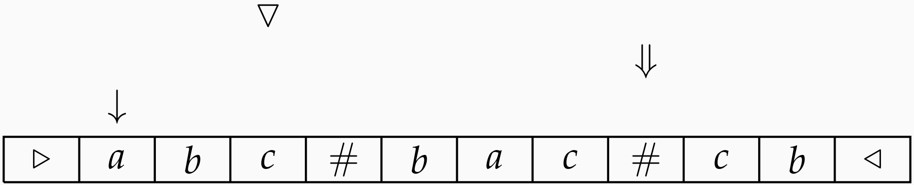
Output: *abca*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



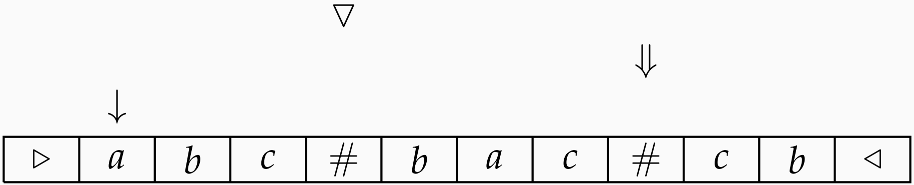
Output: *abcab*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



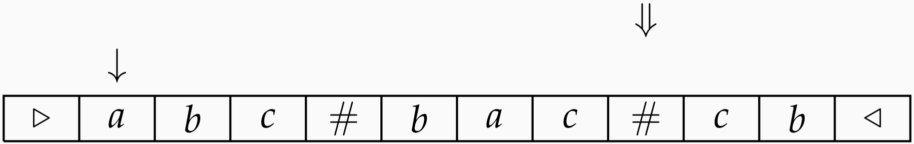
Output: *abcabc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



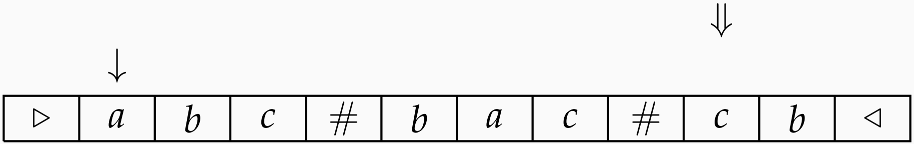
Output: *abcabc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



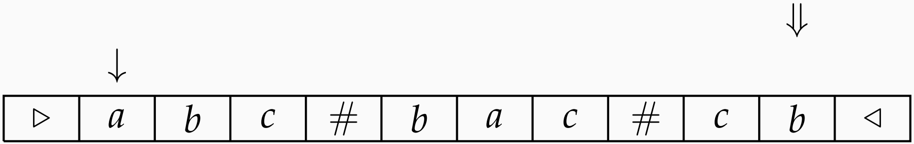
Output: *abcabc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



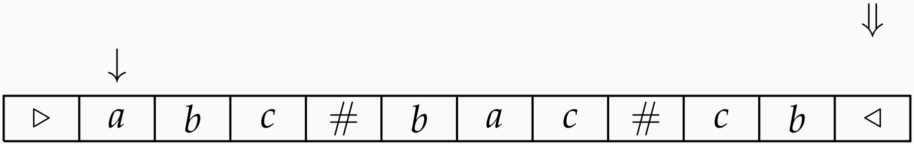
Output: *abcabc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



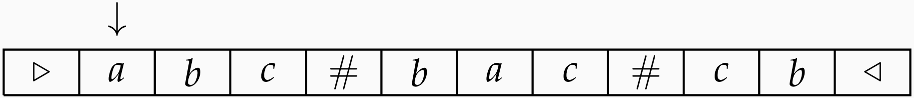
Output: *abcabc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



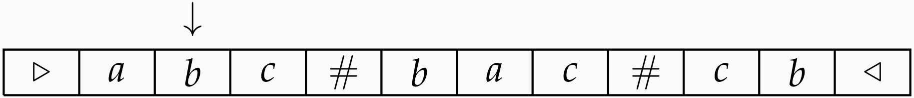
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



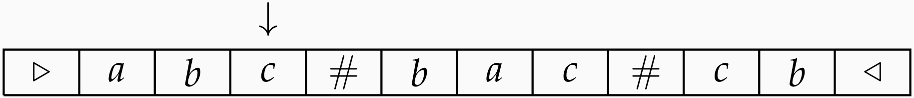
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



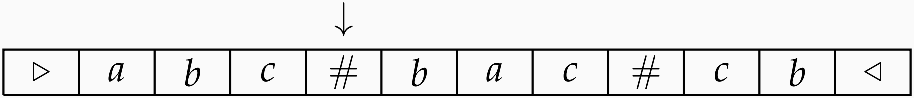
Output: $abcabc\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



Output: $abcabc\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$

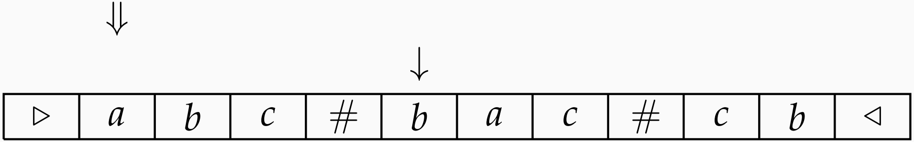
					$\downarrow$						
$\triangleright$	a	b	c	$\#$	b	a	c	$\#$	c	b	$\triangleleft$

Output: $abcabc\#$

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



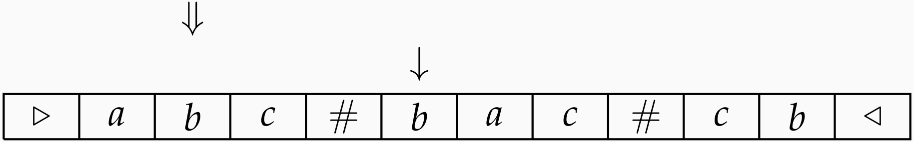
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



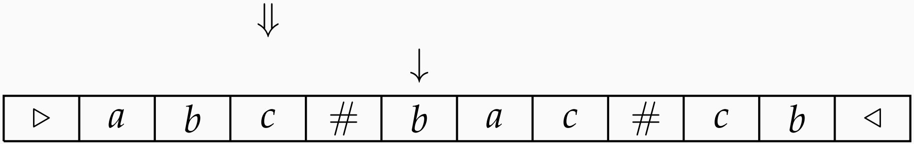
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



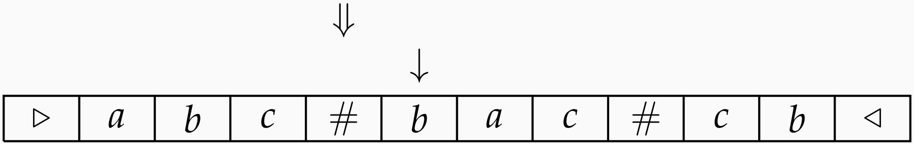
Output: $abcabc\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



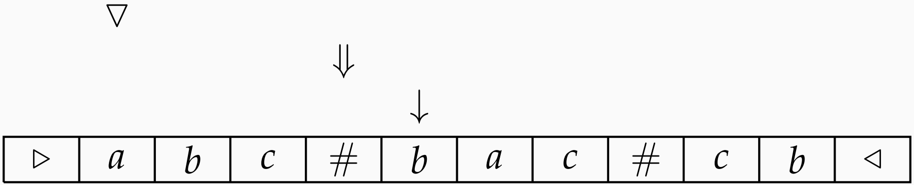
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



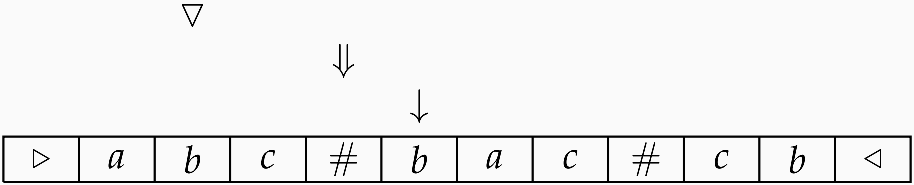
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



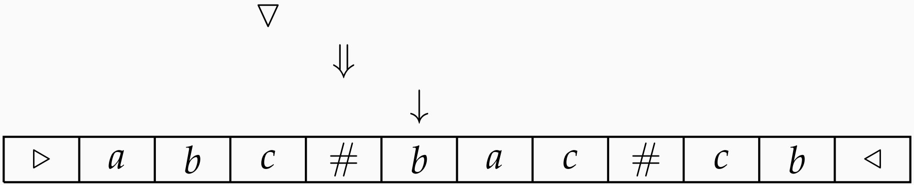
Output: *abcabc#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$

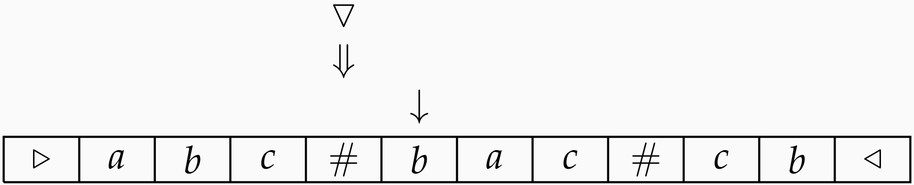


Output: *abcabc#*

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



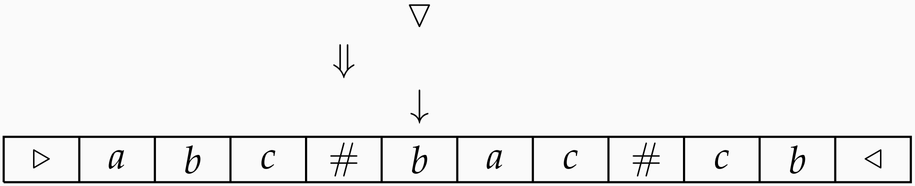
Output: $abcabc\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



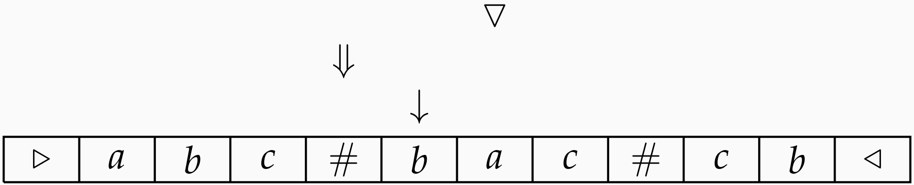
Output: $abcabc\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



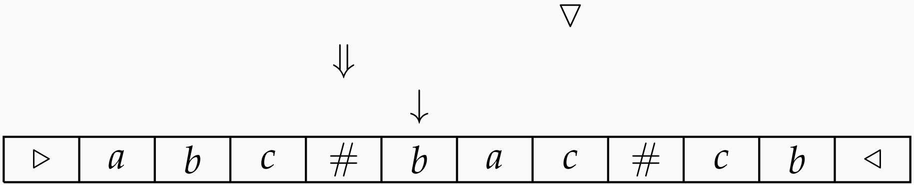
Output: $abcabc\#b$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



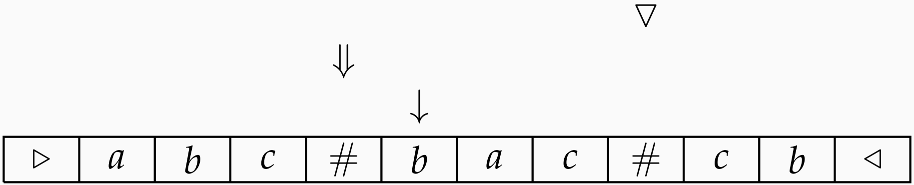
Output: $abcabc\#ba$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



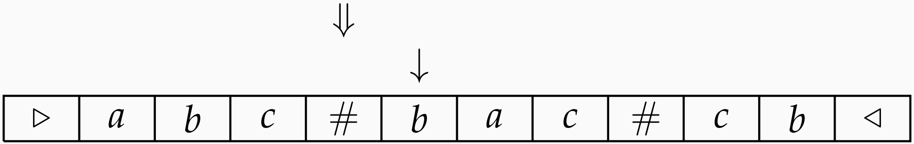
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



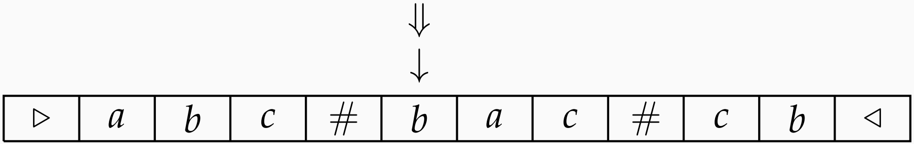
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



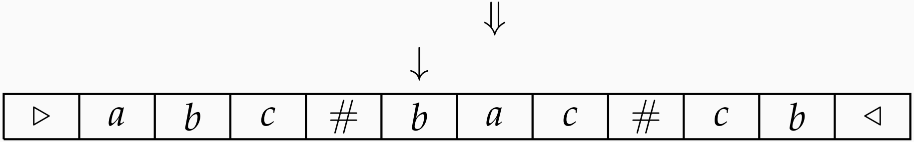
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



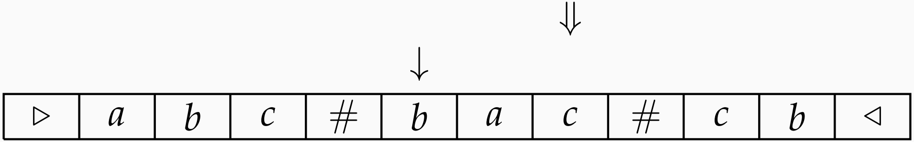
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



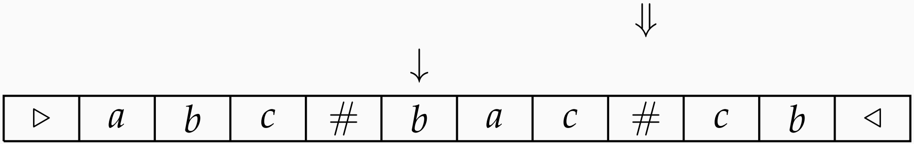
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



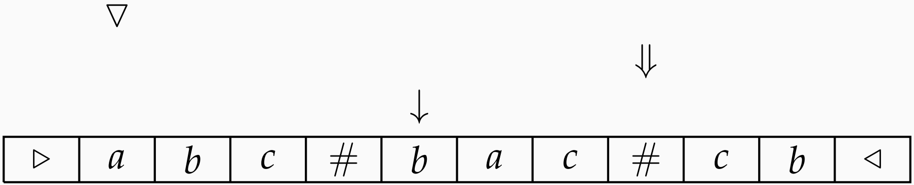
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



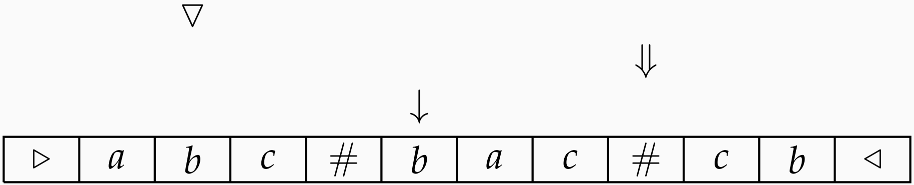
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$

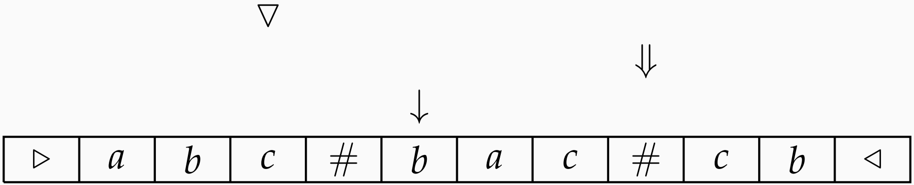


Output: *abcabc#bac*

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



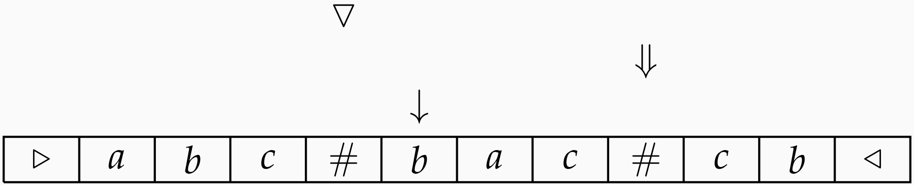
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



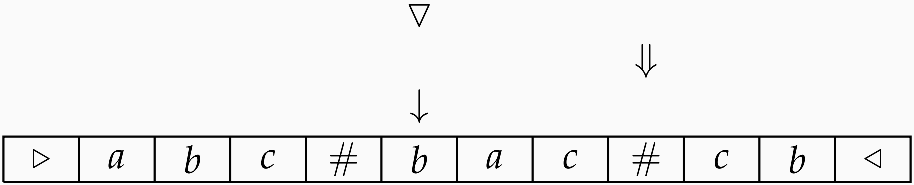
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



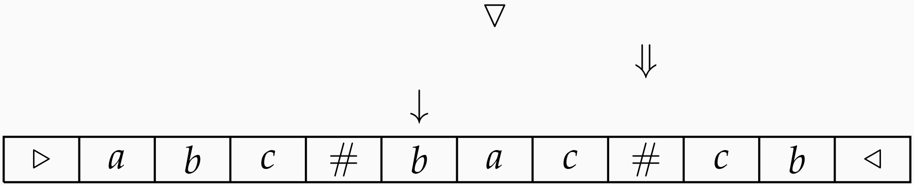
Output: *abcabc#bac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



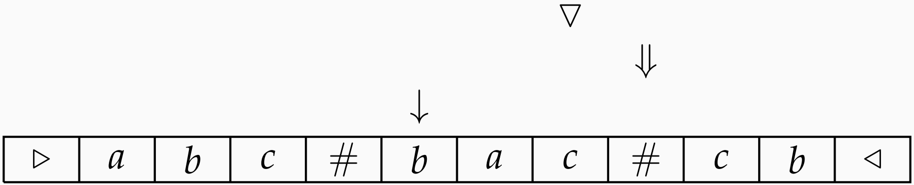
Output: $abcabc\#bacb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



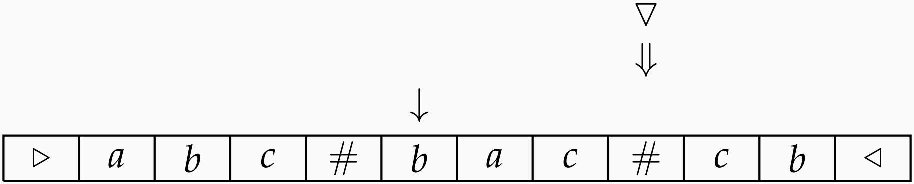
Output: *abcabc#bacba*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



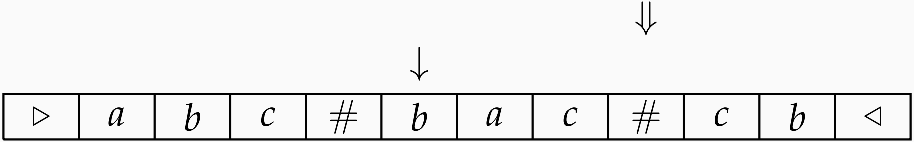
Output: *abcabc#bacbac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



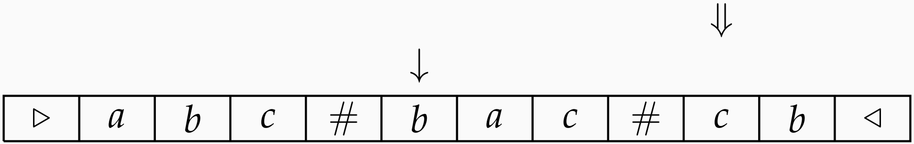
Output: *abcabc#bacbac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



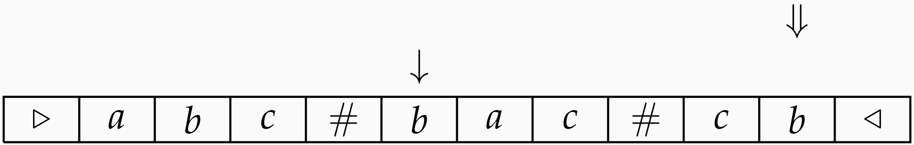
Output: *abcabc#bacbac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



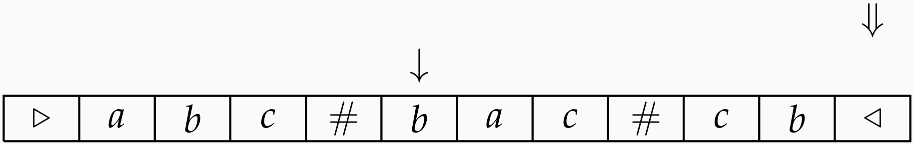
Output: $abcabc\#bacbac$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



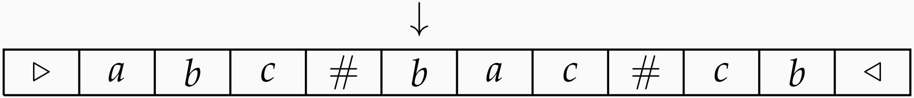
Output: *abcabc#bacbac*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$

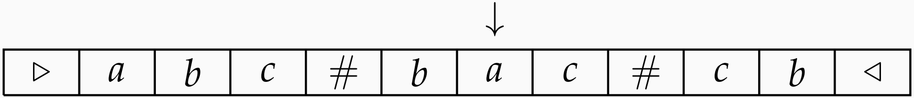


Output: $abcabc\#bacbac\#$

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



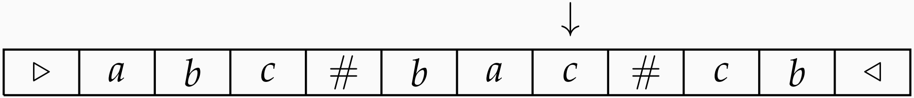
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



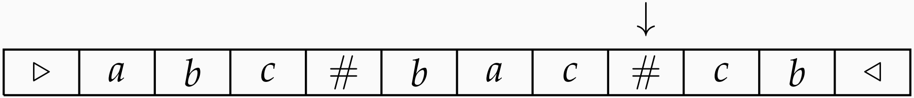
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



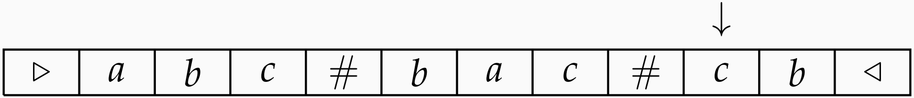
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers $(k \geq 1)$

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$

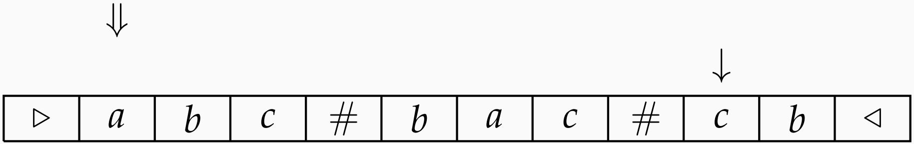


Output: *abcabc#bacbac#*

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



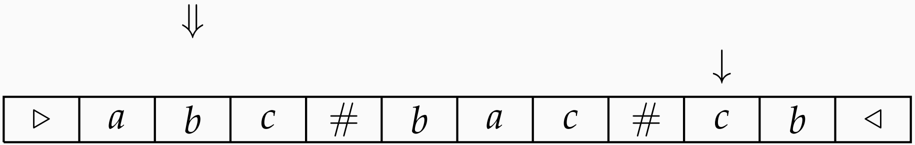
Output: $abcabc\#bacbac\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



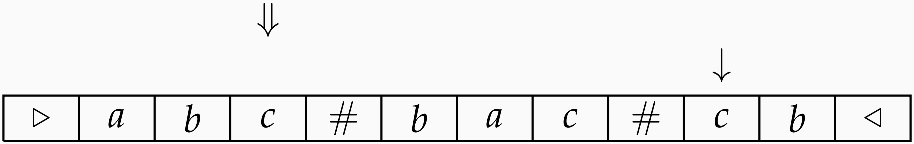
Output: $abcabc\#bacbac\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



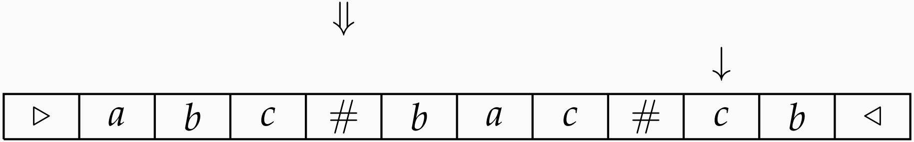
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



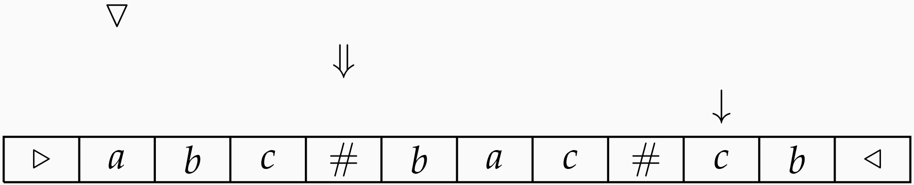
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$

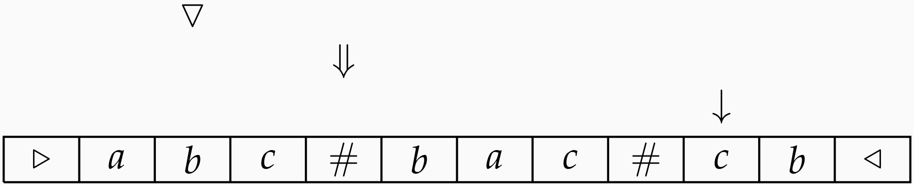


Output: $abcabc\#bacbac\#$

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



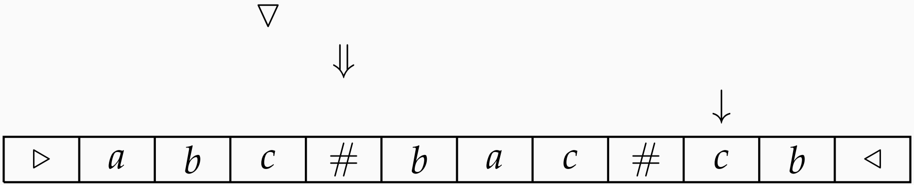
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



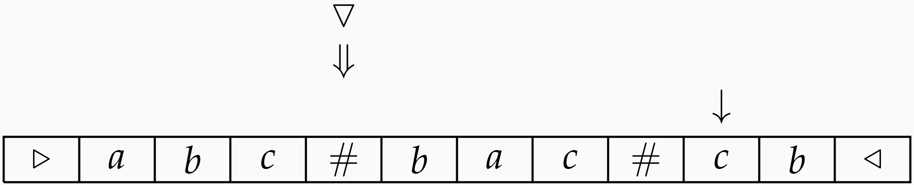
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



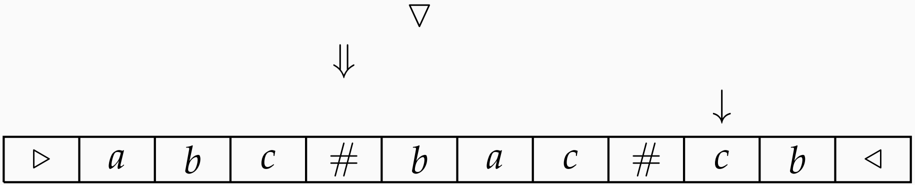
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



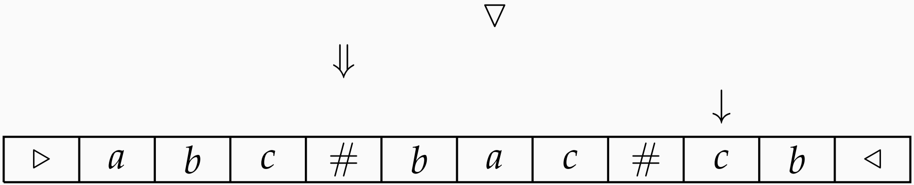
Output: $abcabc\#bacbac\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



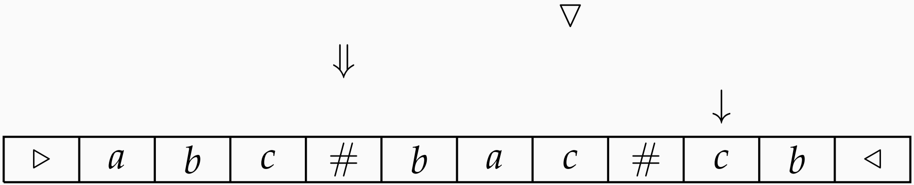
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$

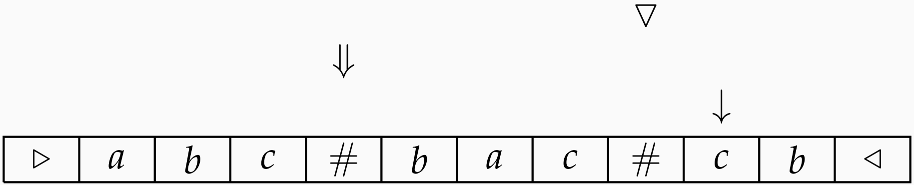


Output: *abcabc#bacbac#*

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



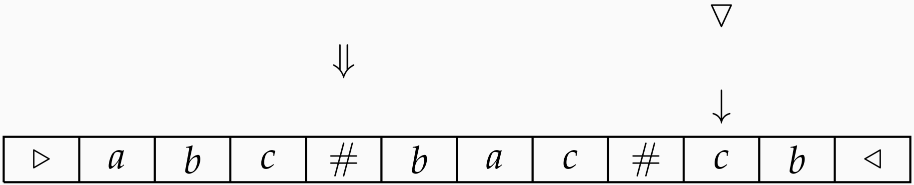
Output: $abcabc\#bacbac\#$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



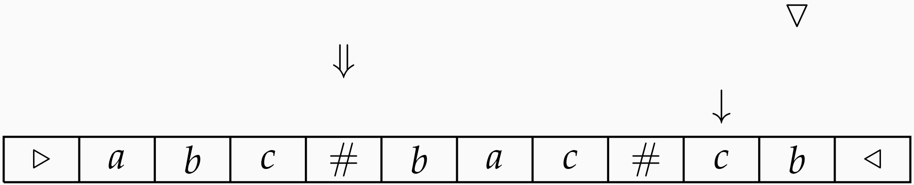
Output: *abcabc#bacbac#*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



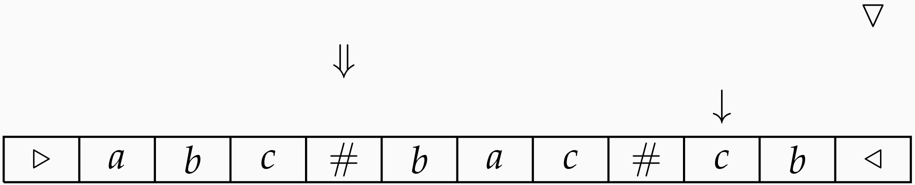
Output: *abcabc#bacbac#c*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



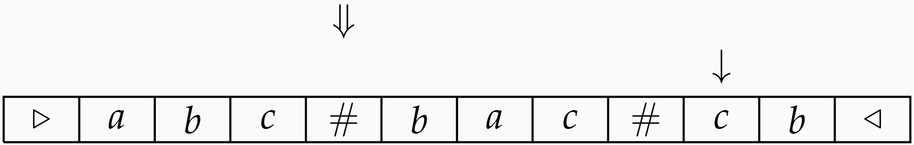
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



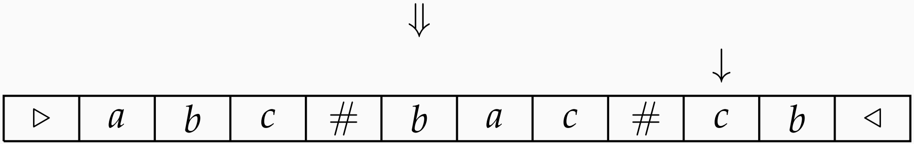
Output: $abcabc\#bacbac\#cb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



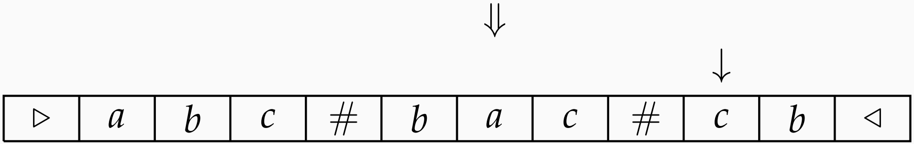
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



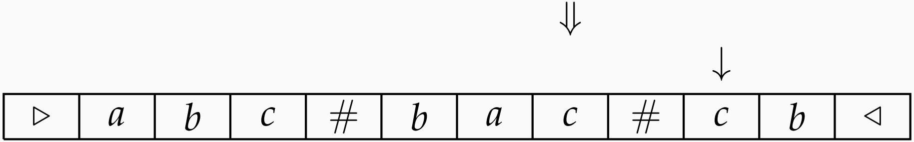
Output: $abcabc\#bacbac\#cb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \#w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



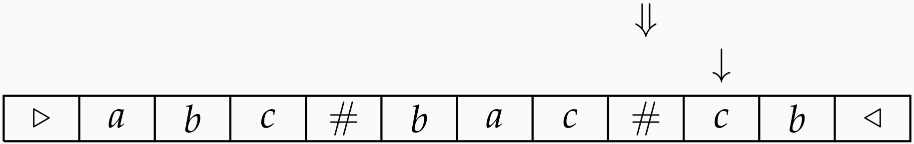
Output: $abcabc\#bacbac\#cb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



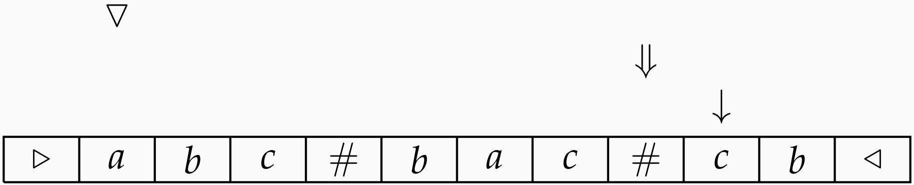
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



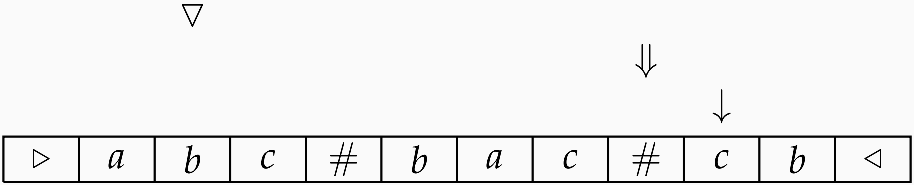
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



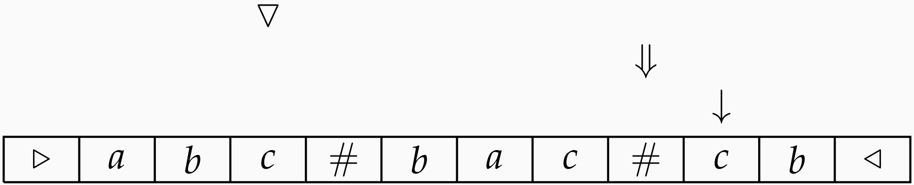
Output: $abcabc\#bacbac\#cb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



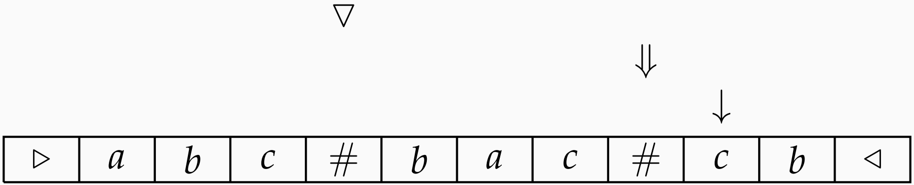
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$

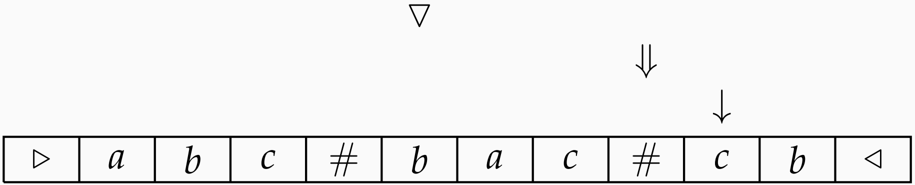


Output: *abcabc#bacbac#cb*

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$

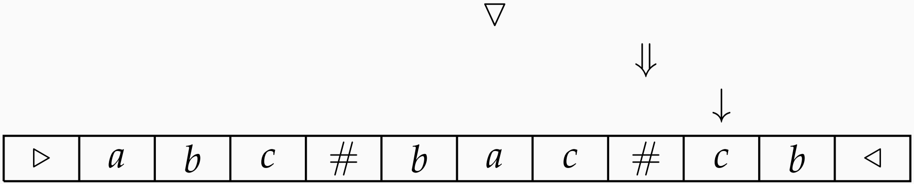


Output: $abcabc\#bacbac\#cb$

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



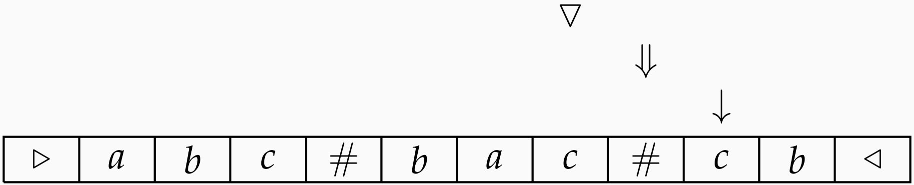
Output: $abcabc\#bacbac\#cb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



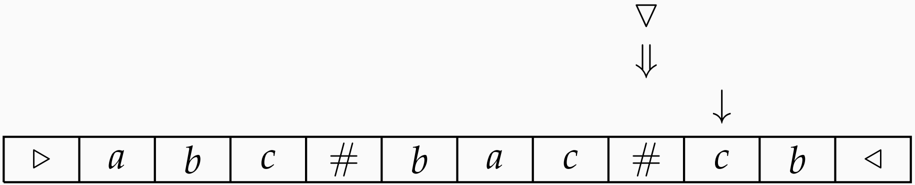
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$



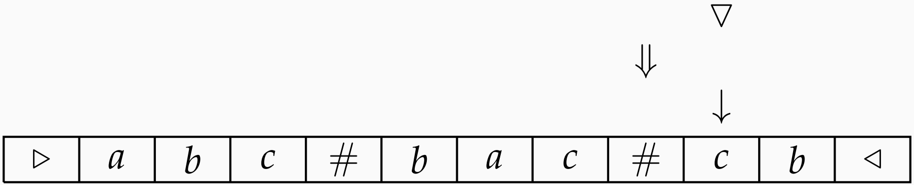
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



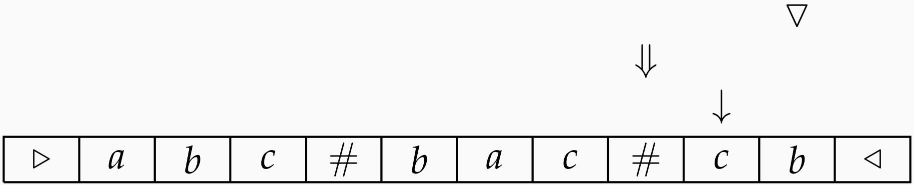
Output: *abcabc#bacbac#cb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}: w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



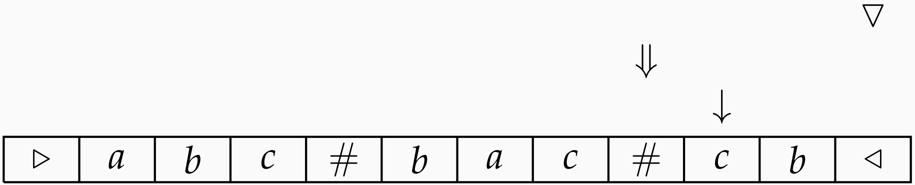
Output: *abcabc#bacbac#cbc*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



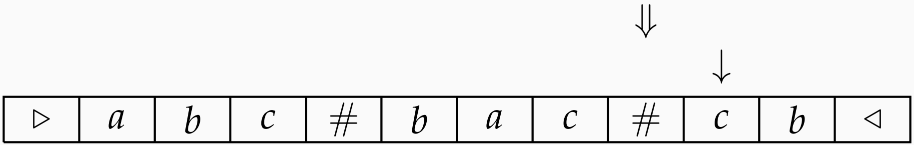
Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$



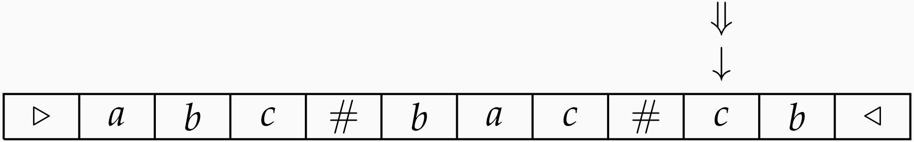
Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



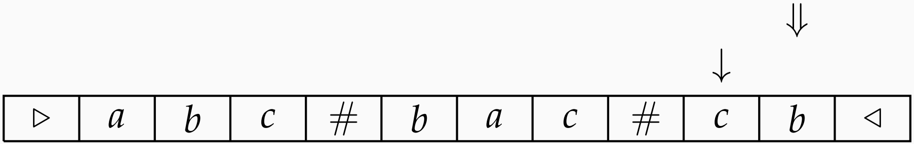
Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \# w_n \longmapsto (w_0)^n \# \dots \# (w_n)^n$

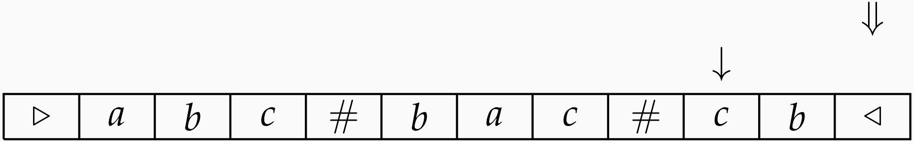


Output: *abcabc#bacbac#cbcb*

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads (“pebbles”)

“Inner squaring” $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



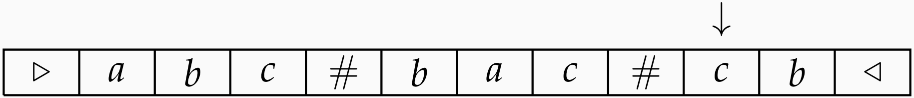
Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\#\ldots\#w_n \longmapsto (w_0)^n\#\ldots\#(w_n)^n$



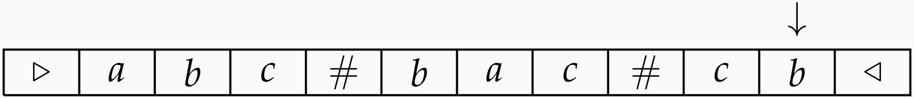
Output: $abcabc\#bacbac\#cbcb$

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by k-pebble transducers ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\mathtt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n\# \ldots \#(w_n)^n$



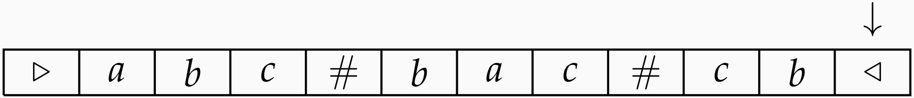
Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> ($k \geq 1$)

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \dots \#w_n \longmapsto (w_0)^n\# \dots \#(w_n)^n$



Output: *abcabc#bacbac#cbcb*

Pebble transducers [Milo, Suciu & Vianu 2000; Engelfriet & Maneth 2002]

Polyregular functions = computed by <u>k-pebble transducers</u> $(k \geq 1)$

DFA (hidden in drawing) + *stack* of height $\leqslant k$ of heads ("pebbles")

"Inner squaring" $\texttt{innsq}\colon w_0\# \ldots \# w_n \longmapsto (w_0)^n \# \ldots \# (w_n)^n$

$\triangleright$	a	b	c	$\#$	b	a	c	$\#$	c	b	$\triangleleft$

Output: $abcabc\#bacbac\#cbcb$

Not shown here: heads are *two-way* $\rightsquigarrow$ can compute e.g. $\texttt{reverse}$

Polyregular functions and their growth

- Closed under composition [Engelfriet & Maneth 2002; Engelfriet 2015]
  - L regular $\implies f^{-1}(L)$ regular
- Several alternative definitions in the last few years → revived interest

  [Bojańczyk 2018, 2023; Bojańczyk, Kiefer & Lhote 2019]
- Polynomial growth: k pebbles $\implies O(n^k)$ growth

Polyregular functions and their growth

- Closed under composition [Engelfriet & Maneth 2002; Engelfriet 2015]
  - L regular $\implies f^{-1}(L)$ regular
- Several alternative definitions in the last few years → revived interest

  [Bojańczyk 2018, 2023; Bojańczyk, Kiefer & Lhote 2019]
- Polynomial growth: k pebbles $\implies O(n^k)$ growth

What about the converse?

For $w = w_0\# \dots \#w_n$, $|\mathtt{innsq}(w)| = |(w_0)^n\# \dots \#(w_n)^n| = O(|w|^2)$

$\longrightarrow$ could $\mathtt{innsq}$ be computed with only 2 pebbles instead of 3?

Polyregular functions and their growth

- Closed under composition [Engelfriet & Maneth 2002; Engelfriet 2015]
  - L regular $\implies f^{-1}(L)$ regular
- Several alternative definitions in the last few years $\to$ revived interest

  [Bojańczyk 2018, 2023; Bojańczyk, Kiefer & Lhote 2019]
- Polynomial growth: k pebbles $\implies O(n^k)$ growth

What about the converse?

For $w = w_0\# \dots \# w_n$, $|\mathtt{innsq}(w)| = |(w_0)^n \# \dots \# (w_n)^n| = O(|w|^2)$

$\longrightarrow$ could $\mathtt{innsq}$ be computed with only 2 pebbles instead of 3?

- Main theorem of a LICS'20 paper: $O(n^k) \implies$ computable with k pebbles

Polyregular functions and their growth

- Closed under composition [Engelfriet & Maneth 2002; Engelfriet 2015]
  - L regular $\implies f^{-1}(L)$ regular
- Several alternative definitions in the last few years $\to$ revived interest [Bojańczyk 2018, 2023; Bojańczyk, Kiefer & Lhote 2019]
- Polynomial growth: k pebbles $\implies O(n^k)$ growth

What about the converse?

For $w = w_0\# \dots \# w_n$, $|\mathtt{innsq}(w)| = |(w_0)^n\# \dots \#(w_n)^n| = O(|w|^2)$
$\longrightarrow$ could $\mathtt{innsq}$ be computed with only 2 pebbles instead of 3?

- Main theorem of a LICS'20 paper: $O(n^k) \implies$ computable with k pebbles
- But actually, $\mathtt{innsq}$ requires 3 pebbles! [Bojańczyk 2023; Kiefer, N. & Pradic 2023]

Polyregular functions and their growth

- Closed under composition [Engelfriet & Maneth 2002; Engelfriet 2015]
  - L regular $\implies f^{-1}(L)$ regular
- Several alternative definitions in the last few years → revived interest

  [Bojańczyk 2018, 2023; Bojańczyk, Kiefer & Lhote 2019]
- Polynomial growth: k pebbles $\implies O(n^k)$ growth

What about the converse?

For $w = w_0\# \ldots \# w_n$, $|\mathtt{innsq}(w)| = |(w_0)^n\# \ldots \#(w_n)^n| = O(|w|^2)$

$\longrightarrow$ could $\mathtt{innsq}$ be computed with only 2 pebbles instead of 3?

- Main theorem of a LICS'20 paper: $O(n^k) \implies$ computable with k pebbles
- But actually, $\mathtt{innsq}$ requires 3 pebbles! [Bojańczyk 2023; Kiefer, N. & Pradic 2023]
- We will be able to realize $\mathtt{innsq}$ with 2 "pointers to input" using *logic*

Example: $w \mapsto a^{|w|} \cdot \texttt{reverse}(w)$

$$\begin{array}{cccc} a & b & a & c \\ 1 & 2 & 3 & 4 \end{array} \quad \rightsquigarrow \quad \begin{array}{cccccccc} a & a & a & a & c & a & b & a \\ \lambda 1 & \lambda 2 & \lambda 3 & \lambda 4 & \rho 4 & \rho 3 & \rho 2 & \rho 1 \end{array}$$

$I = \{\lambda, \rho\}$ $\quad a(\lambda i) = \texttt{true}, a(\rho i) = (w[i] = a)$ $\quad \lambda 1 \prec \lambda 2 \prec \cdots \prec \rho 2 \prec \rho 1$

Logical transductions

Example: $w \mapsto a^{|w|} \cdot \texttt{reverse}(w)$

$$\begin{array}{cccc} a & b & a & c \\ 1 & 2 & 3 & 4 \end{array} \quad \rightsquigarrow \quad \begin{array}{cccccccc} a & a & a & a & c & a & b & a \\ \lambda 1 & \lambda 2 & \lambda 3 & \lambda 4 & \rho 4 & \rho 3 & \rho 2 & \rho 1 \end{array}$$

$I = \{\lambda, \rho\}$ $\quad a(\lambda i) = \texttt{true}, a(\rho i) = (w[i] = a)$ $\quad \lambda 1 \prec \lambda 2 \prec \cdots \prec \rho 2 \prec \rho 1$

Idea: for an input word $w \in \Gamma^*$, define over $z, z' \in I \times \{1, \dots, |w|\}$

unary relations $a(z)$ for $a \in \Sigma$ $\quad + \quad$ a binary relation $z \prec z'$

if we're lucky, the result is isomorphic to an output word $f(w) \in \Sigma^*$

MSO transductions

Reminder on Monadic Second-Order logic

MSO formula: $\varphi(x_1, \ldots, x_n)$, the x_k refer to *positions* of a word w ($1 \leq x_k \leq |w|$)

$$\varphi, \psi ::= \underbrace{a(x)}_{\text{position } x \text{ has label } a} \mid x < y \mid \exists x.\, \varphi \mid \underbrace{\exists X.\, \varphi}_{X \subseteq \text{positions}} \mid x \in X \mid \varphi \wedge \psi \mid \neg\varphi$$

1960s: $L \subseteq \Gamma^*$ regular language $\iff \exists \varphi.\ L = \{w \in \Sigma^* \mid w \vDash \varphi\}$ (for $n = 0$)

MSO transduction = finite set I + $\varphi^i_a(x)$ + $\varphi^{i,j}_{\prec}(x, y)$ for $a \in \Sigma$ and $i, j \in I$

MSO transductions

Reminder on Monadic Second-Order logic

MSO formula: $\varphi(x_1, \dots, x_n)$, the x_k refer to *positions* of a word w ($1 \le x_k \le |w|$)

$$\varphi, \psi ::= \underbrace{a(x)}_{\text{position } x \text{ has label } a} \mid x < y \mid \exists x.\, \varphi \mid \underbrace{\exists X.\, \varphi}_{X \subseteq \text{positions}} \mid x \in X \mid \varphi \wedge \psi \mid \neg\varphi$$

1960s: $L \subseteq \Gamma^*$ regular language $\iff \exists \varphi.\ L = \{w \in \Sigma^* \mid w \vDash \varphi\}$ (for $n = 0$)

MSO transduction = finite set I + $\varphi^i_a(x)$ + $\varphi^{i,j}_{\prec}(x, y)$ for $a \in \Sigma$ and $i, j \in I$

Theorem [Engelfriet & Hoogeboom 2001]

String-to-string MSO transductions $\equiv$ 1-pebble (i.e. “two-way”) transducers

(Not too hard once you know these transducers are closed under composition)

$\longrightarrow$ this is called the class of *regular functions*

MSO interpretations in higher dimension

MSO interpretation $\Gamma^* \to \Sigma^*$ = choose *dimension* $k \in \mathbb{N}$, a finite set I & formulas

$$\varphi_a^i(x_1, \ldots, x_k) \text{ for } a \in \Sigma \qquad \varphi_{\prec}^{i,j}(x_1, \ldots, x_k, y_1, \ldots, y_k) \qquad i, j \in I$$

$w \in \Gamma^* \longmapsto$ relations $a(-)$ and $\prec$ over $I \times \{1, \ldots, |w|\}^k$

again, if we're lucky, this structure is isomorphic to some $f(w) \in \Sigma^*$

MSO interpretations in higher dimension

MSO interpretation $\Gamma^* \to \Sigma^*$ = choose *dimension* $k \in \mathbb{N}$, a finite set I & formulas

$$\varphi^i_a(x_1, \ldots, x_k) \text{ for } a \in \Sigma \qquad \varphi^{i,j}_{\prec}(x_1, \ldots, x_k, y_1, \ldots, y_k) \qquad i,j \in I$$

$w \in \Gamma^* \longmapsto$ relations $a(-)$ and $\prec$ over $I \times \{1, \ldots, |w|\}^k$

again, if we're lucky, this structure is isomorphic to some $f(w) \in \Sigma^*$

Theorem [Bojańczyk, Kiefer & Lhote 2019]

String-to-string MSO interpretations = polyregular functions

- Highly technical proof using finite model theory
- Somewhat "unnatural": no reason *a priori* for MSO interpretations to preserve regular languages by inverse image

  whereas MSO transductions (1-dim.) compose by syntactic substitution

Example of MSO interpretation

$\texttt{innsq}' : w_0\# \ldots \# w_n \longmapsto (w_0)^n \ldots (w_n)^n$ has a dim. 2 (optimal) interpretation:

$$
\begin{array}{cccc}
 & acab\#abba\#c & & \\
\vdots & & \ldots & \\
\# & acab & abba & c \\
\vdots & & \ldots & \\
\# & acab & abba & c \\
\vdots & & \ldots &
\end{array}
\longrightarrow (acab)(acab)(abba)(abba)(c)(c)
$$

Example of MSO interpretation

$\texttt{innsq}' : w_0\# \dots \# w_n \longmapsto (w_0)^n \dots (w_n)^n$ has a dim. 2 (optimal) interpretation:

$$\begin{array}{cccc} & acab\#abba\#c & & \\ \vdots & & \dots & \\ \# & acab & abba & c \\ \vdots & & \dots & \\ \# & acab & abba & c \\ \vdots & & \dots & \end{array} \longrightarrow (acab)(acab)(abba)(abba)(c)(c)$$

- $\varphi_a(x_1, x_2) = a(x_1) \wedge \#(x_2)$
- $\varphi_{\prec}(x_1, x_2, y_1, y_2) = \exists x_3, y_3.$ which begin blocks containing resp. x_1, y_1
  and $(x_3, x_2, x_1) < (y_3, y_2, y_1)$ lex. $\longrightarrow$ pebbles $\downarrow, \Downarrow, \triangledown$

Dimension minimisation

Theorem [Bojańczyk 2023]

MSO interpretations *of dim.* k on strings = polyregular fn *with growth* $O(n^k)$

Dimension minimisation

Theorem [Bojańczyk 2023]

MSO interpretations *of dim.* k on strings = polyregular fn *with growth* $O(n^k)$

Fundamentally, it's not about interpretations, it's about *queries*:

Main lemma

Let $\varphi(x_1, \ldots, x_\ell)$ be an MSO formula over Γ^*. One can compute:

- the least $k \in \mathbb{N}$ such that $|\{(i_1, \ldots, i_\ell) \mid w \models \varphi(i_1, \ldots, i_\ell)\}| = O(|w|^k)$ (so $k \leq \ell$);
- $\psi(x_1, \ldots, x_\ell, z_1, \ldots, z_k)$ and $B \in \mathbb{N}$ such that for every $w \in \Gamma^*$,
  - $\forall j_1, \ldots, j_k,\ |\{(i_1, \ldots, i_\ell) \mid w \models \psi(i_1, \ldots, i_\ell, j_1, \ldots, j_k)\}| \leq B$;
  - $\forall i_1, \ldots, i_\ell,\ w \models \varphi(i_1, \ldots, i_\ell) \implies |\{(j_1, \ldots, j_k) \mid w \models \psi(i_1, \ldots, i_\ell, j_1, \ldots, j_k)\}| = 1$.

Suffices to derive the theorem by simple syntactic "reparametrization"

MSO query reparametrization made easy

- Bojańczyk proves (something more precise than) the Main Lemma via compositionality of MSO + factorisation forests
- This is overkill: the Main Lemma reduces to a structure theorem on *polynomially ambiguous automata*, obtained by "simple-minded" pumping

  origin: [Seidl & Weber 1991]; convenient variant: [Douéneau-Tabot, Filiot & Gastin 2020]

MSO query reparametrization made easy

- Bojańczyk proves (something more precise than) the Main Lemma via compositionality of MSO + factorisation forests
- This is overkill: the Main Lemma reduces to a structure theorem on *polynomially ambiguous automata*, obtained by "simple-minded" pumping

  origin: [Seidl & Weber 1991]; convenient variant: [Douéneau-Tabot, Filiot & Gastin 2020]

Connection between MSO queries and ambiguous automata

$\varphi(x_1, \ldots, x_\ell) \rightsquigarrow$ DFA recognizing words with ℓ marked positions

$\xrightarrow[\text{projection } (\Gamma\times\{0,1\}^\ell)^*\to\Gamma^*]{}$ NFA recognizing words without marks

Ambiguity (nb of runs) of NFA on $w \in \Gamma^*$ = nb of "w+marks" accepted by DFA

= nb of query matches on w

MSO set queries and set interpretations

Connection between MSO set queries and ambiguous automata

$\varphi(\underbrace{X_1, \ldots, X_\ell}_{\text{variables ranging over } \textit{subsets} \text{ of positions}}) \rightsquigarrow$ DFA recognizing words with $\{0,1\}^\ell$-coloring

$\longrightarrow$ NFA recognizing words without colors

$\underbrace{\text{Ambiguity}}_{\text{now possibly exponential}}$ of NFA on $w \in \Gamma^*$ = nb of "w+colors" accepted by DFA

= nb of query matches on w

MSO set queries and set interpretations

Connection between MSO set queries and ambiguous automata

$\underbrace{\varphi(X_1, \ldots, X_\ell)}_{\text{variables ranging over } \textit{subsets} \text{ of positions}} \rightsquigarrow$ DFA recognizing words with $\{0,1\}^\ell$-coloring

$\longrightarrow$ NFA recognizing words without colors

$\underbrace{\text{Ambiguity}}_{\text{now possibly exponential}}$ of NFA on $w \in \Gamma^*$ = nb of "w+colors" accepted by DFA

= nb of query matches on w

Structure thm of poly. amb. NFA $\implies$ can determine whether nb of matches of φ is $O(n^k)$, and if so, compute reparametrization $\psi(X_1, \ldots, X_\ell, z_1, \ldots, z_k)$

Corollary: generalization of Bojańczyk's dimension minimization theorem

MSO set interpretation of growth $O(n^k) \equiv$ MSO interpretation of dim. k

def: specified by $\varphi_a(X_1, \ldots, X_\ell) + \varphi_\prec(X_1, \ldots, X_\ell, Y_1, \ldots, Y_\ell)$ [Colcombet & Löding 2007]

Generalization to trees

- Simple pumping (pigeonhole principle) instead of factorization forests
  $\longrightarrow$ can hope for extension from strings to *ranked trees*

Generalization to trees

- Simple pumping (pigeonhole principle) instead of factorization forests
  $\longrightarrow$ can hope for extension from strings to *ranked trees*
- Reuse ideas from Erik Paul's master thesis (Univ. Leipzig, 2015)
  $\longrightarrow$ proof "from scratch" of "main lemma" in a few pages
  $\longrightarrow$ dimension minimization for *tree-to-anything* MSO set interpretations follows by same syntactic argument as before

Generalization to trees

- Simple pumping (pigeonhole principle) instead of factorization forests
  $\longrightarrow$ can hope for extension from strings to *ranked trees*
- Reuse ideas from Erik Paul's master thesis (Univ. Leipzig, 2015)
  $\longrightarrow$ proof "from scratch" of "main lemma" in a few pages
  $\longrightarrow$ dimension minimization for *tree-to-anything* MSO set interpretations follows by same syntactic argument as before
- No fully black-box reduction to known literature…
  but this "main lemma" on MSO set queries on trees entails a new(??) result:

Corollary

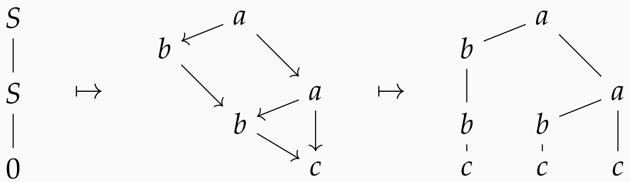
Given a tree automaton as input, the least $k \in \mathbb{N}$ such that it is $O(n^k)$-ambiguous is computable. (also poly/exp ambiguity dichotomy: was explicitly stated by E. Paul)

Important examples of MSO set interpretation over trees (1)

Proposition

If $f\colon \mathrm{Tree}(\Gamma) \to \mathrm{Tree}(\Sigma)$ is defined by an *MSO transduction with sharing*, then it is also defined by some MSO set interpretation.

i.e. $f = (\mathrm{Tree}(\Gamma) \xrightarrow[\text{i.e. 1-dim. MSO interpretation}]{\text{some MSO transduction}} \mathrm{rootedDAG}(\Sigma) \xrightarrow{\texttt{unfold}} \mathrm{Tree}(\Sigma))$

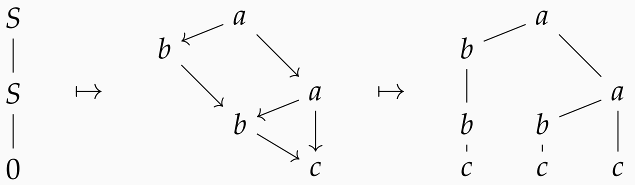


Important examples of MSO set interpretation over trees (1)

Proposition

If $f\colon \mathrm{Tree}(\Gamma) \to \mathrm{Tree}(\Sigma)$ is defined by an *MSO transduction with sharing*,
then it is also defined by some MSO set interpretation.

i.e. $f = \left(\mathrm{Tree}(\Gamma) \xrightarrow[\text{i.e. 1-dim. MSO interpretation}]{\text{some MSO transduction}} \mathrm{rootedDAG}(\Sigma) \xrightarrow{\texttt{unfold}} \mathrm{Tree}(\Sigma)\right)$



$|f(t)| = \Theta(|t|^2)$ here; $\exists$ example of growth $\Theta(2^n)$ (complete binary tree)

Important examples of MSO set interpretation over trees (2)

Proposition

If f is defined by an *MSO transduction with sharing* i.e. $\texttt{unfold} \circ$ [MSO trans.]

(or equivalently: by an attribute grammar / a tree-walking transducer with regular lookaround)

then it is also defined by some MSO set interpretation.

- $\texttt{unfold}$ is defined by a DAG-to-tree MSO set interpretation

  (idea: output nodes = input paths from the root)

- on arbitrary structures: [MSO set interp.] $\circ$ [MSO trans.] $\subseteq$ [MSO set interp.]

  (by the usual syntactic substitution argument)

Important examples of MSO set interpretation over trees (2)

Proposition

If f is defined by an *MSO transduction with sharing* i.e. `unfold` $\circ$ [MSO trans.]

(or equivalently: by an attribute grammar / a tree-walking transducer with regular lookaround)

then it is also defined by some MSO set interpretation.

- `unfold` is defined by a DAG-to-tree MSO set interpretation

  (idea: output nodes = input paths from the root)

- on arbitrary structures: [MSO set interp.] $\circ$ [MSO trans.] $\subseteq$ [MSO set interp.]

  (by the usual syntactic substitution argument)

$\Longrightarrow$ **The growth rate theorem applies!** new result on MSO trans. w/ sharing
e.g. the example on previous slide admits a 2-dim. interpretation

The linear growth case

In particular...

If $f\colon \mathrm{Tree}(\Gamma) \to \mathrm{Tree}(\Sigma)$ is defined by an MSO transduction w/ sharing (MSOTS) and $|f(t)| = O(|t|)$, then it is also defined by an MSO transduction (MSOT).

Existing result, but the only known proof [Engelfriet & Maneth 2003] is very technical, has weaker assumption "f computed by some macro tree transducer"

$$\text{known:} \quad \text{MSOTS} \subseteq \text{macro tree transducer} \subseteq \text{MSOTS} \circ \text{MSOTS}$$

The linear growth case

In particular...

If $f\colon \mathrm{Tree}(\Gamma) \to \mathrm{Tree}(\Sigma)$ is defined by an MSO transduction w/ sharing (MSOTS) and $|f(t)| = O(|t|)$, then it is also defined by an MSO transduction (MSOT).

Existing result, but the only known proof [Engelfriet & Maneth 2003] is very technical, has weaker assumption "f computed by some macro tree transducer"

known: MSOTS $\subseteq$ macro tree transducer $\subseteq$ MSOTS $\circ$ MSOTS

Proposition

MSOTS $\circ$ MSOTS $\subseteq$ `unfold` $\circ$ [(tree-to-DAG) MSO set interpretation]

$$\text{MSOT} \circ \texttt{unfold} \circ \text{MSOT} \equiv \text{FOT} \circ \text{MSO relabeling} \circ \texttt{unfold} \circ \text{MSOT}$$

(FOT = first-order transductions) $\subseteq$ FOT $\circ$ (`unfold` $\circ$ MSOT) $\circ$ MSOT

unfold vs relabeling "commutation lemma": $\exists$ something similar in Carayol's PhD

A linear growth argument

macro tree transducer $\subseteq$ $\mathtt{unfold} \circ$ [(tree-to-DAG) MSO set interpretation]

Theorem – generalizing [Engelfriet & Maneth 2003] thanks to the above

If $f = \mathtt{unfold} \circ$ [some MSO set interpretation] and $|f(t)| = O(|t|)$,

then f is defined by some MSO transduction.

A linear growth argument

macro tree transducer $\subseteq$ $\texttt{unfold} \circ$ [(tree-to-DAG) MSO set interpretation]

Theorem – generalizing [Engelfriet & Maneth 2003] thanks to the above

If $f = \texttt{unfold} \circ$ [some MSO set interpretation] and $|f(t)| = O(|t|)$,

then f is defined by some MSO transduction.

Since $|\texttt{unfold}(G)| \geq |G|$, the MSO set interpretation in the statement is $O(n)$

$\Rightarrow$ by growth rate theorem on set interpretation, it's equivalent to an MSOT

$\Rightarrow f = \texttt{unfold} \circ$ [some MSOT] and $|f(t)| = O(|t|)$

$\rightsquigarrow$ conclude using theorem on MSOT w/ sharing = $\texttt{unfold} \circ$ MSOT! $\square$

A linear growth argument

macro tree transducer $\subseteq$ $\texttt{unfold} \circ$ [(tree-to-DAG) MSO set interpretation]

Theorem – generalizing [Engelfriet & Maneth 2003] thanks to the above

If $f = \texttt{unfold} \circ$ [some MSO set interpretation] and $|f(t)| = O(|t|)$,
then f is defined by some MSO transduction.

Since $|\texttt{unfold}(G)| \geq |G|$, the MSO set interpretation in the statement is $O(n)$
$\Rightarrow$ by growth rate theorem on set interpretation, it's equivalent to an MSOT
$\Rightarrow f = \texttt{unfold} \circ$ [some MSOT] and $|f(t)| = O(|t|)$
$\rightsquigarrow$ conclude using theorem on MSOT w/ sharing = $\texttt{unfold} \circ$ MSOT! $\square$

Future work

Reprove the generalization of [Engelfriet, Inaba & Maneth 2021] to entire composition hierarchy of MSOT w/ sharing, using similarly "clean" arguments macro tree transducer $\subseteq$ $\mathtt{unfold} \circ$ [(tree-to-DAG) MSO set interpretation]

Theorem – generalizing [Engelfriet & Maneth 2003] thanks to the above

If $f = \mathtt{unfold} \circ$ [some MSO set interpretation] and $|f(t)| = O(|t|)$, then f is defined by some MSO transduction.

Since $|\mathtt{unfold}(G)| \geq |G|$, the MSO set interpretation in the statement is $O(n)$
$\Rightarrow$ by growth rate theorem on set interpretation, it's equivalent to an MSOT
$\Rightarrow f = \mathtt{unfold} \circ$ [some MSOT] and $|f(t)| = O(|t|)$
$\rightsquigarrow$ conclude using theorem on MSOT w/ sharing = $\mathtt{unfold} \circ$ MSOT! $\square$

Future work

Reprove the generalization of [Engelfriet, Inaba & Maneth 2021] to entire composition hierarchy of MSOT w/ sharing, using similarly "clean" arguments