

Foundations of Complexity Theory

Lecture 3: Foundational Relations II

October 28, 2026

Next Dates

LECTURE (WED 1:15PM)	
Oct 28th	Lecture 3: Fundamental relations II
Nov 4th	Lecture 4: Fundamental relations III
11th	Lecture 5: Reductions & complete problems I
18th	
	(public holiday)
25th	Lecture 6: Complete problems II
Dec 2nd	
	(dies academicus)
9th	Lecture 7: NP-complete problems
16th	Lecture 8: Complete problems III

LECTURE (WED 1:15PM)	
Dec 23rd	
30th	
Jan 6th	Lecture 9: Boolean circuits I
13th	Lecture 10: Boolean circuits II
20th	Lecture 11: Razborov's theorem I
27th	
Feb 3rd	Lecture 12: Razborov's theorem II

Contents

- 1 Basic complexity classes & their relations
- 2 Reductions & complete problems
- 3 Boolean circuits & Razborov's theorem
- 4 Randomization & interactive proof systems
- 5 Polynomial primality test

Complexity Classes

Standard classes

$$\mathbf{L} = \mathbf{DSpace}(\log x)$$

$$\mathbf{NL} = \mathbf{NSpace}(\log x)$$

$$\mathbf{P} = \bigcup_{i \in \mathbb{N}_+} \mathbf{DTIME}(x^i)$$

$$\mathbf{NP} = \bigcup_{i \in \mathbb{N}_+} \mathbf{NTIME}(x^i)$$

$$\mathbf{PSPACE} = \bigcup_{i \in \mathbb{N}_+} \mathbf{DSpace}(x^i) \stackrel{\dagger}{=} \bigcup_{i \in \mathbb{N}_+} \mathbf{NSpace}(x^i)$$

$$\mathbf{EXPTIME} = \bigcup_{i \in \mathbb{N}_+} \mathbf{DTIME}(2^{x^i})$$

$$\mathbf{NEXPTIME} = \bigcup_{i \in \mathbb{N}_+} \mathbf{NTIME}(2^{x^i})$$

$\dagger$ = Savitch's theorem

§3.1 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(f) \subseteq \text{NTIME}(f) \subseteq \text{DSpace}(f)$

Relations between Complexity Classes

§3.1 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n) \geq n$ for all $n \in \mathbb{N}$.
Then $\text{DTIME}(f) \subseteq \text{NTIME}(f) \subseteq \text{DSpace}(f)$

Proof

$\text{DTIME}(f) \subseteq \text{NTIME}(f)$ trivial.

Relations between Complexity Classes

§3.1 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(f) \subseteq \text{NTIME}(f) \subseteq \text{DSpace}(f)$

Proof

$\text{DTIME}(f) \subseteq \text{NTIME}(f)$ trivial. Let $M = (Q, \Sigma, \Gamma, \Sigma', \delta, \square, q_0, q_+, q_-)$ be f -time-bounded TM. Input $w \in \Sigma^*$ accepted by M iff exists protocol $(\rho_1, \dots, \rho_m)$ with $m \leq f(|w| + 2)$ such that computation is accepting.

Relations between Complexity Classes

§3.1 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ such that $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(f) \subseteq \text{NTIME}(f) \subseteq \text{DSpace}(f)$

Proof

$\text{DTIME}(f) \subseteq \text{NTIME}(f)$ trivial. Let $M = (Q, \Sigma, \Gamma, \Sigma', \delta, \square, q_0, q_+, q_-)$ be f -time-bounded TM. Input $w \in \Sigma^*$ accepted by M iff exists protocol $(\rho_1, \dots, \rho_m)$ with $m \leq f(|w| + 2)$ such that computation is accepting. Store those f -length-bounded protocols in space $O(f)$, length-lexicographically enumerate them, and execute them to check validity and acceptance. Using linear tape compression (Theorem §2.5) we achieve $L(M) \in \text{DSpace}(f)$. □

Space Hierarchy Theorem

§3.2 Theorem (Space hierarchy theorem)

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ space-constructible with $f(n) \geq \log n$ for all $n \in \mathbb{N}$.
Then $\text{DSPACE}(o(f)) \neq \text{DSPACE}(O(f))$

Space Hierarchy Theorem

§3.2 Theorem (Space hierarchy theorem)

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ space-constructible with $f(n) \geq \log n$ for all $n \in \mathbb{N}$.
Then $\text{DSPACE}(o(f)) \neq \text{DSPACE}(O(f))$

Notes

- $g \in o(f)$ if and only if (iff) $\lim_{n \rightarrow \infty} \frac{g(n)}{f(n)} = 0$
- Wlog. assume TMs only utilize tapes right of starting position

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of deterministic 1-TM $\mathcal{M}$.

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of deterministic 1-TM $\mathcal{M}$.

Use diagonalization with space-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right. \\ \left. \text{and utilizes only 4 tape symbols} \right\}$$

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0,1\}^*$ of deterministic 1-TM $\mathcal{M}$.

Use diagonalization with space-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right. \\ \left. \text{and utilizes only 4 tape symbols} \right\}$$

We first prove $L \notin \text{DSPACE}(o(f))$. Let $\mathcal{M}$ be det. TM with $L(\mathcal{M}) = L$ requiring only $o(f)$ space.

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of deterministic 1-TM $\mathcal{M}$.
Use diagonalization with space-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

We first prove $L \notin \text{DSPACE}(o(f))$. Let $\mathcal{M}$ be det. TM with $L(\mathcal{M}) = L$ requiring only $o(f)$ space. There also exists equivalent det. 1-TM $\mathcal{M}'$ that utilizes only 4 tape symbols and $o(f)$ space such that $L(\mathcal{M}') = L(\mathcal{M}) = L$.

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0,1\}^*$ of deterministic 1-TM $\mathcal{M}$.
Use diagonalization with space-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right. \\ \left. \text{and utilizes only 4 tape symbols} \right\}$$

We first prove $L \notin \text{DSPACE}(o(f))$. Let $\mathcal{M}$ be det. TM with $L(\mathcal{M}) = L$ requiring only $o(f)$ space. There also exists equivalent det. 1-TM $\mathcal{M}'$ that utilizes only 4 tape symbols and $o(f)$ space such that $L(\mathcal{M}') = L(\mathcal{M}) = L$. Hence TM $\mathcal{M}'$ utilizes strictly fewer space than $f(|\langle \text{bin}(\mathcal{M}'), w \rangle| + 2)$ on input $\langle \text{bin}(\mathcal{M}'), w \rangle$ for sufficiently long w .

Space Hierarchy Theorem

Proof (1/3)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of deterministic 1-TM $\mathcal{M}$.
Use diagonalization with space-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

We first prove $L \notin \text{DSPACE}(o(f))$. Let $\mathcal{M}$ be det. TM with $L(\mathcal{M}) = L$ requiring only $o(f)$ space. There also exists equivalent det. 1-TM $\mathcal{M}'$ that utilizes only 4 tape symbols and $o(f)$ space such that $L(\mathcal{M}') = L(\mathcal{M}) = L$. Hence TM $\mathcal{M}'$ utilizes strictly fewer space than $f(|\langle \text{bin}(\mathcal{M}'), w \rangle| + 2)$ on input $\langle \text{bin}(\mathcal{M}'), w \rangle$ for sufficiently long w . However, this yields contradiction

$$\langle \text{bin}(\mathcal{M}'), w \rangle \in L \iff \langle \text{bin}(\mathcal{M}'), w \rangle \notin L(\mathcal{M}')$$

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (2/3)

It remains to prove $L \in \text{DSPACE}(O(f))$. Use universal TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$.

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (2/3)

It remains to prove $L \in \text{DSPACE}(O(f))$. Use universal TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f is space-constructible we mark $f(n+2)$ cells on tape within space $f(n+2)$.

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right. \\ \left. \text{and utilizes only 4 tape symbols} \right\}$$

Proof (2/3)

It remains to prove $L \in \text{DSPACE}(O(f))$. Use universal TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f is space-constructible we mark $f(n+2)$ cells on tape within space $f(n+2)$. By Lemma §1.7 there exist constants $a, b \in \mathbb{N}$ such that at most $m = a \cdot n \cdot b^{f(n+2)}$ configurations are reachable from initial configuration.

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (2/3)

It remains to prove $L \in \text{DSPACE}(O(f))$. Use universal TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f is space-constructible we mark $f(n+2)$ cells on tape within space $f(n+2)$. By Lemma §1.7 there exist constants $a, b \in \mathbb{N}$ such that at most $m = a \cdot n \cdot b^{f(n+2)}$ configurations are reachable from initial configuration. Counter ℓ initialized to m stored in binary on another tape requires space $O(f(x) + \log x) = O(f)$.

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (2/3)

It remains to prove $L \in \text{DSPACE}(O(f))$. Use universal TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f is space-constructible we mark $f(n+2)$ cells on tape within space $f(n+2)$. By Lemma §1.7 there exist constants $a, b \in \mathbb{N}$ such that at most $m = a \cdot n \cdot b^{f(n+2)}$ configurations are reachable from initial configuration. Counter ℓ initialized to m stored in binary on another tape requires space $O(f(x) + \log x) = O(f)$.

Start simulation of $\mathcal{M}$ inside marked tape cells.

Decrement counter ℓ after each simulated step.

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (3/3)

Distinguish 3 outcomes

- If simulation leaves marked cells (before ℓ reaches 0), then reject because more than $f(n + 2)$ space utilized

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (3/3)

Distinguish 3 outcomes

- If simulation leaves marked cells (before ℓ reaches 0), then reject because more than $f(n + 2)$ space utilized
- If counter ℓ reaches 0, then accept because configuration was repeated, so $\mathcal{M}$ is in infinite loop

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \text{ and utilizes only 4 tape symbols} \right\}$$

Proof (3/3)

Distinguish 3 outcomes

- If simulation leaves marked cells (before ℓ reaches 0), then reject because more than $f(n + 2)$ space utilized
- If counter ℓ reaches 0, then accept because configuration was repeated, so $\mathcal{M}$ is in infinite loop
- Otherwise invert decision of $\mathcal{M}$

Space Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ within space } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right. \\ \left. \text{and utilizes only 4 tape symbols} \right\}$$

Proof (3/3)

Distinguish 3 outcomes

- If simulation leaves marked cells (before ℓ reaches 0), then reject because more than $f(n + 2)$ space utilized
- If counter ℓ reaches 0, then accept because configuration was repeated, so $\mathcal{M}$ is in infinite loop
- Otherwise invert decision of $\mathcal{M}$

Det. TM U thus accepts language L with space consumption $O(f)$. □

Theorem of Hennie & Stearns

Frederick C. Hennie III (* 1933)

- US computer scientist
- Worked at MIT on lower bounds for SAT
- Namesake for “Teaching Excellence Award”



© MIT

Richard Edwin Stearns (* 1936)

- US computer scientist
- Published result with Hennie in 1966
- Received ACM-Turing-Prize in 1993



© R.E. Stearns

§3.3 Theorem (Theorem of Hennie & Stearns)

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ be such that $f(n) \geq n$ for all $n \in \mathbb{N}$ and $L \in \text{DTIME}(f)$.
There exists $(f(x) \cdot \lceil \log f(x) \rceil)$ -time-bounded det. 2-TM $\mathcal{M}'$ with $L(\mathcal{M}') = L$

Theorem of Hennie & Stearns

§3.3 Theorem (Theorem of Hennie & Stearns)

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ be such that $f(n) \geq n$ for all $n \in \mathbb{N}$ and $L \in \text{DTIME}(f)$.
There exists $(f(x) \cdot \lceil \log f(x) \rceil)$ -time-bounded det. 2-TM M' with $L(M') = L$

Notes

- Know simulation of k -TM by 1-TM (quadratic overhead)
- Collection of tape symbols causes unnecessary runs over tape
- Avoid head divergence (keep all heads reasonably close)
- Proof by Pippenger & Fischer (1979)

Theorem of Hennie & Stearns

Proof (1/3)

Let k -TM M and w input of length n . We construct 2-TM with tape of k -tuples (as in Theorem §2.4) and additionally mark head positions. 2nd tape only needed for fast copying and as storage for call stack.

Theorem of Hennie & Stearns

Proof (1/3)

Let k -TM M and w input of length n . We construct 2-TM with tape of k -tuples (as in Theorem §2.4) and additionally mark head positions. 2nd tape only needed for fast copying and as storage for call stack. We call initial position on 1st tape *origin*. We keep heads close to origin.

Theorem of Hennie & Stearns

Proof (1/3)

Let k -TM M and w input of length n . We construct 2-TM with tape of k -tuples (as in Theorem §2.4) and additionally mark head positions. 2nd tape only needed for fast copying and as storage for call stack. We call initial position on 1st tape *origin*. We keep heads close to origin. We use `Simulate( $m$ )` with m power of 2 that simulates m steps of M .

- If $m = 1$, then simulate step of M and return to origin

Theorem of Hennie & Stearns

Proof (1/3)

Let k -TM M and w input of length n . We construct 2-TM with tape of k -tuples (as in Theorem §2.4) and additionally mark head positions. 2nd tape only needed for fast copying and as storage for call stack. We call initial position on 1st tape *origin*. We keep heads close to origin. We use `Simulate( $m$ )` with m power of 2 that simulates m steps of M .

- If $m = 1$, then simulate step of M and return to origin
- Otherwise execute

`Center( $m$ ) ; Simulate( $\frac{m}{2}$ ) ; Center( $m$ ) ; Simulate( $\frac{m}{2}$ ) ; Rewind( $m$ )`

(second $\frac{m}{2}$ steps continue simulation after first $\frac{m}{2}$ steps)

Theorem of Hennie & Stearns

Proof (2/3)

Call `Simulate( $m$ )` requires heads in interval $[-m, m]$ around origin.

Theorem of Hennie & Stearns

Proof (2/3)

Call $\text{Simulate}(m)$ requires heads in interval $[-m, m]$ around origin.

$\text{Center}(m)$ moves heads on all simulated tapes to interval $[-\frac{m}{2}, \frac{m}{2}]$.

Theorem of Hennie & Stearns

Proof (2/3)

Call $\text{Simulate}(m)$ requires heads in interval $[-m, m]$ around origin.

$\text{Center}(m)$ moves heads on all simulated tapes to interval $[-\frac{m}{2}, \frac{m}{2}]$.

We achieve this in linear time.

On each simulated tape search head position in interval $[-m, m]$ and decide direction of rotation of interval $[-2m, 2m]$ by m if necessary.

Theorem of Hennie & Stearns

Proof (2/3)

Call $\text{Simulate}(m)$ requires heads in interval $[-m, m]$ around origin.

$\text{Center}(m)$ moves heads on all simulated tapes to interval $[-\frac{m}{2}, \frac{m}{2}]$.

We achieve this in linear time.

On each simulated tape search head position in interval $[-m, m]$ and decide direction of rotation of interval $[-2m, 2m]$ by m if necessary. With 2nd tape cyclic rotation by m can be implemented in linear time (per tape and overall).

Theorem of Hennie & Stearns

Proof (2/3)

Call $\text{Simulate}(m)$ requires heads in interval $[-m, m]$ around origin.

$\text{Center}(m)$ moves heads on all simulated tapes to interval $[-\frac{m}{2}, \frac{m}{2}]$.

We achieve this in linear time.

On each simulated tape search head position in interval $[-m, m]$ and decide direction of rotation of interval $[-2m, 2m]$ by m if necessary. With 2nd tape cyclic rotation by m can be implemented in linear time (per tape and overall).

After simulation of $\frac{m}{2}$ steps heads remain in interval $[-m, m]$, but potentially outside $[-\frac{m}{2}, \frac{m}{2}]$ forcing another call to $\text{Center}(m)$.

Theorem of Hennie & Stearns

Proof (2/3)

Call $\text{Simulate}(m)$ requires heads in interval $[-m, m]$ around origin.

$\text{Center}(m)$ moves heads on all simulated tapes to interval $[-\frac{m}{2}, \frac{m}{2}]$.

We achieve this in linear time.

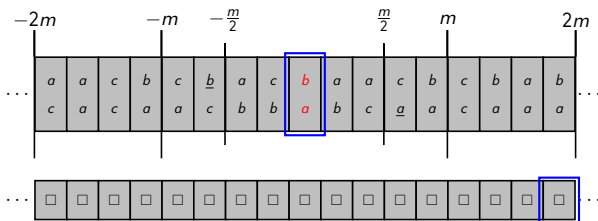
On each simulated tape search head position in interval $[-m, m]$ and decide direction of rotation of interval $[-2m, 2m]$ by m if necessary. With 2nd tape cyclic rotation by m can be implemented in linear time (per tape and overall).

After simulation of $\frac{m}{2}$ steps heads remain in interval $[-m, m]$, but potentially outside $[-\frac{m}{2}, \frac{m}{2}]$ forcing another call to $\text{Center}(m)$.

$\text{Rewind}(m)$ undoes all rotations (requires remembering which rotations were performed to which tapes).

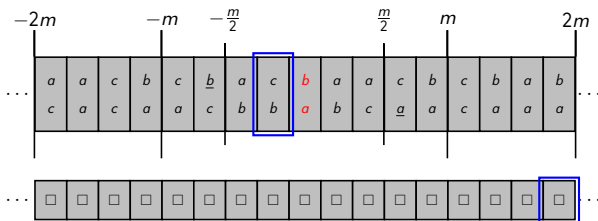
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



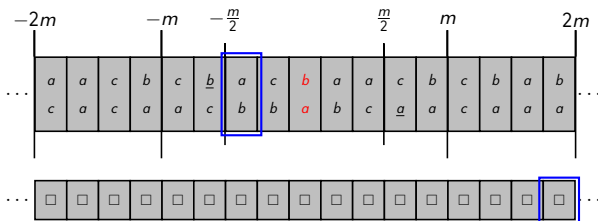
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



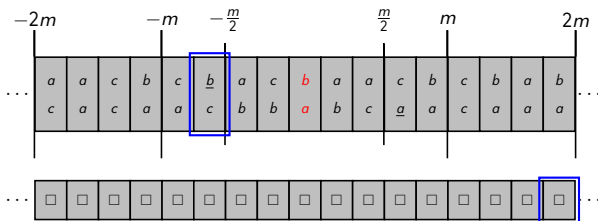
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



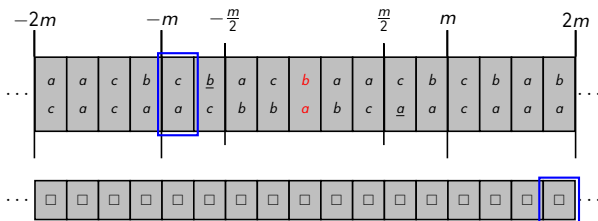
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



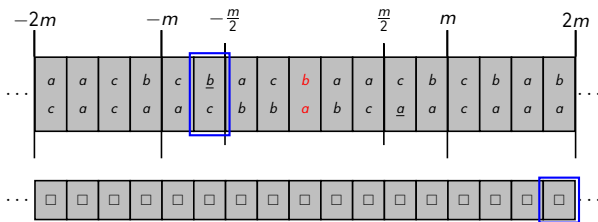
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



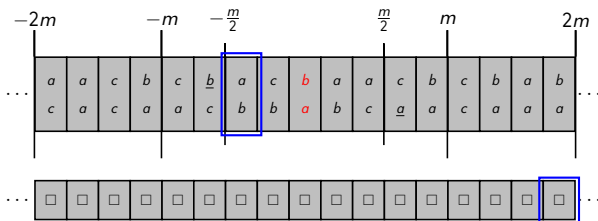
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



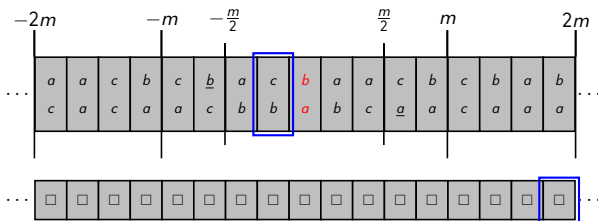
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



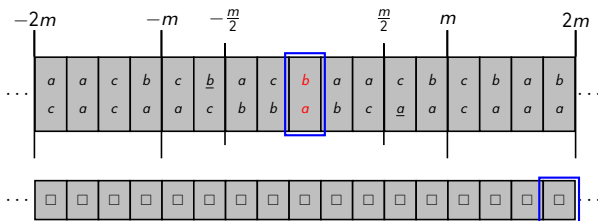
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



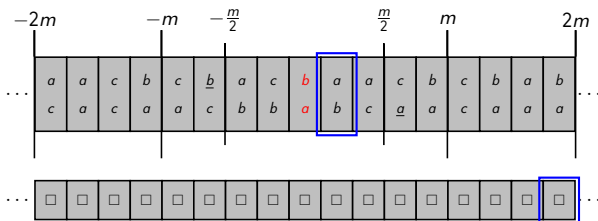
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



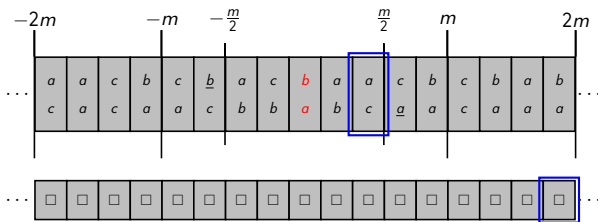
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



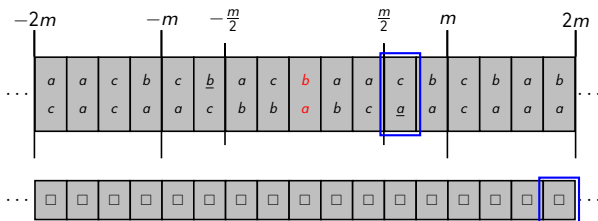
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



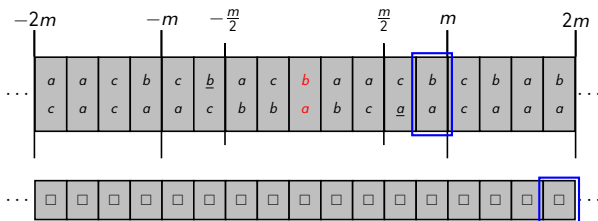
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



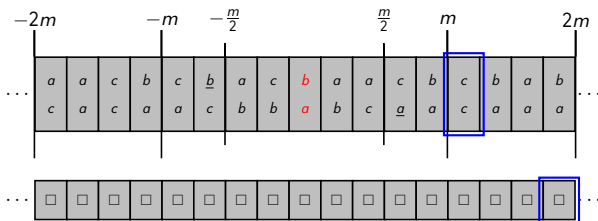
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



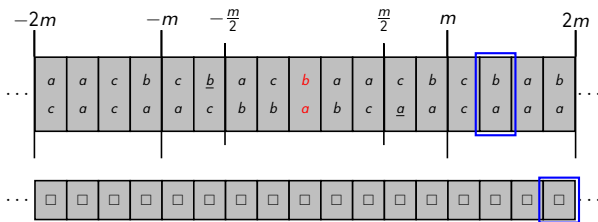
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



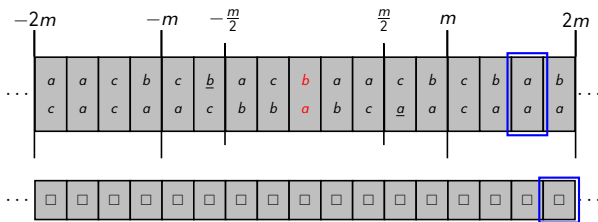
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



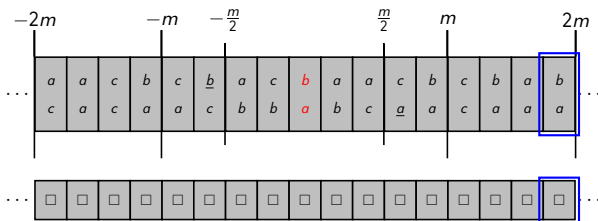
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



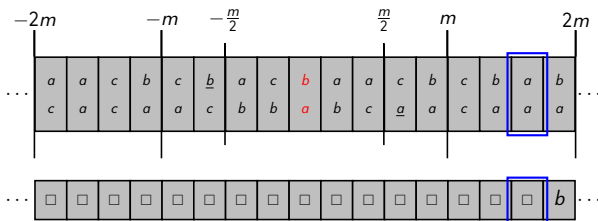
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



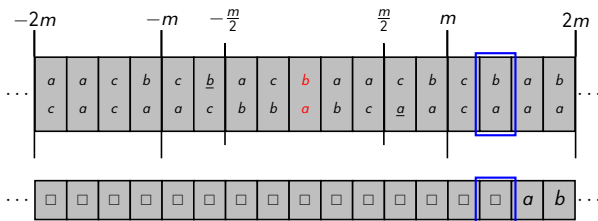
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



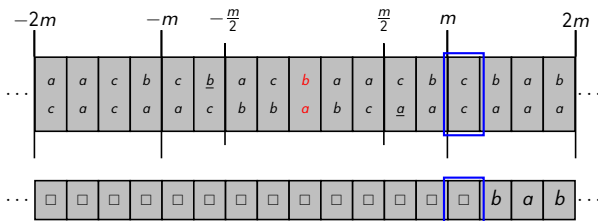
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



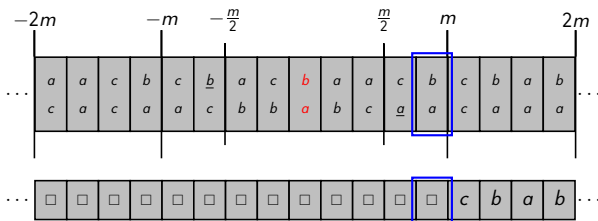
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



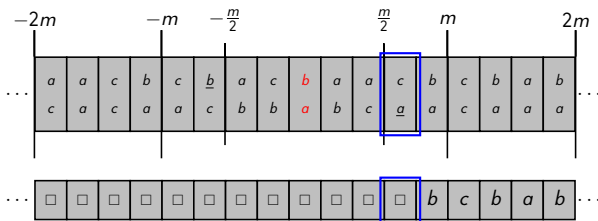
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



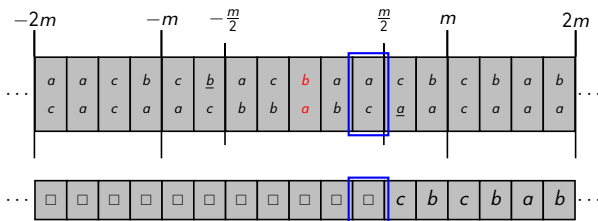
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



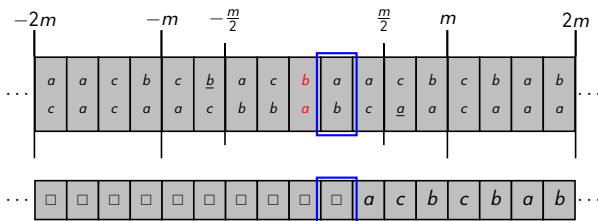
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



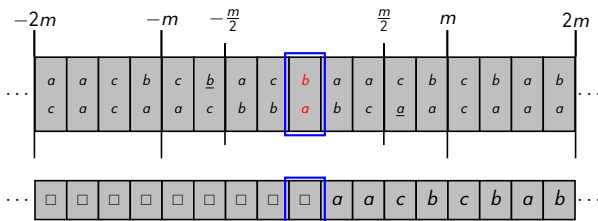
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



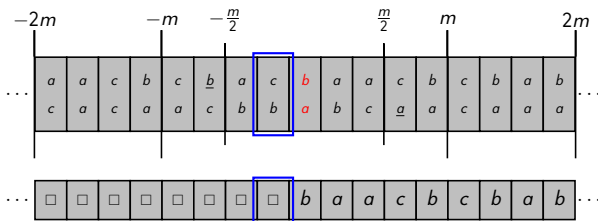
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



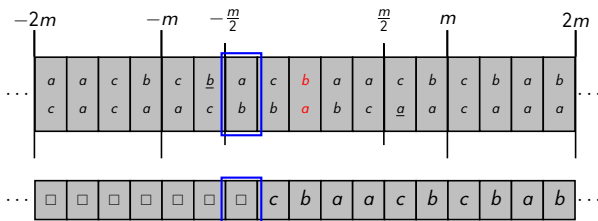
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



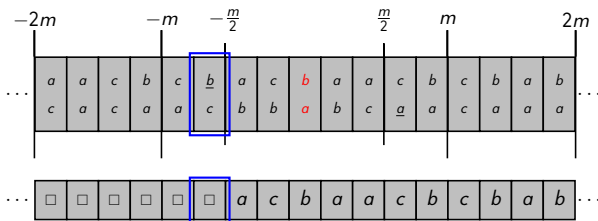
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



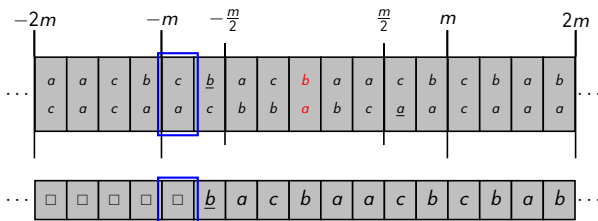
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



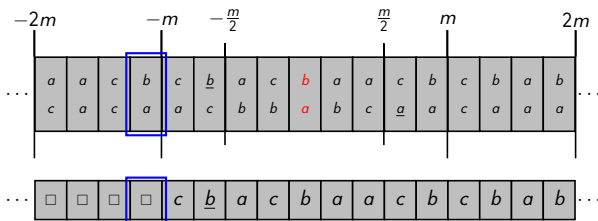
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



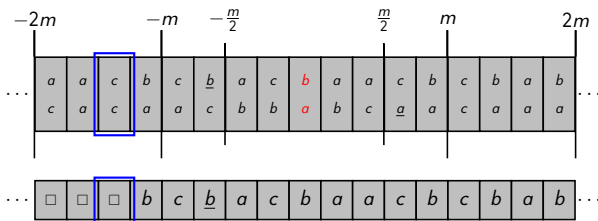
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



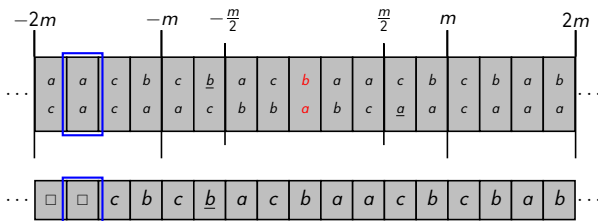
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



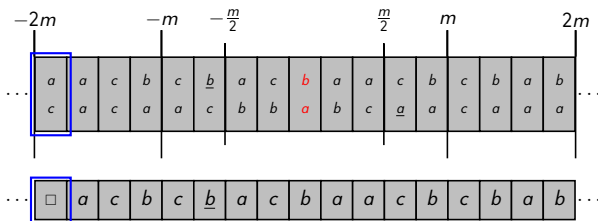
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



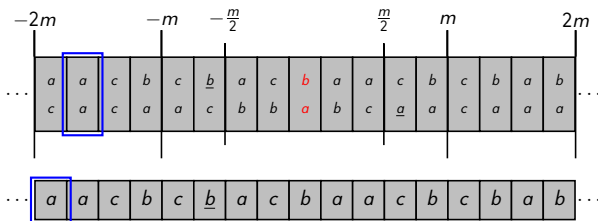
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



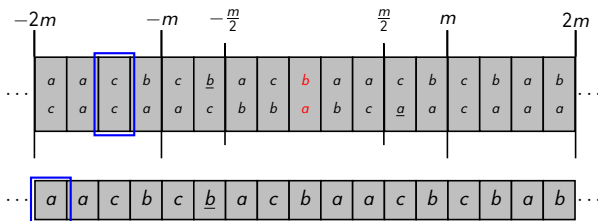
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



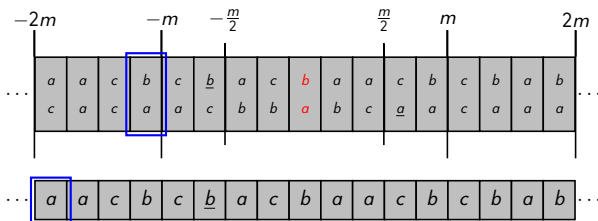
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



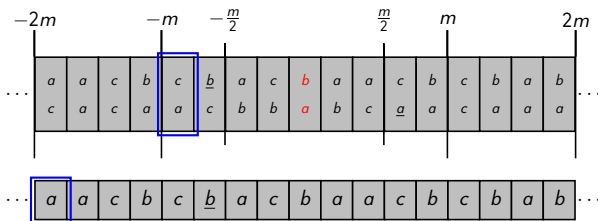
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



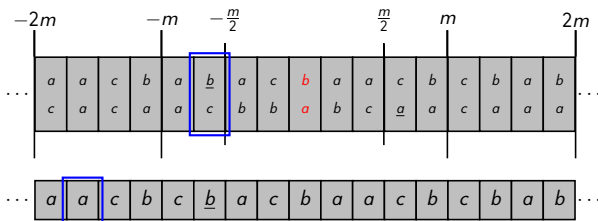
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



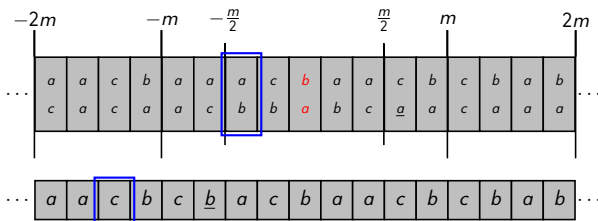
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



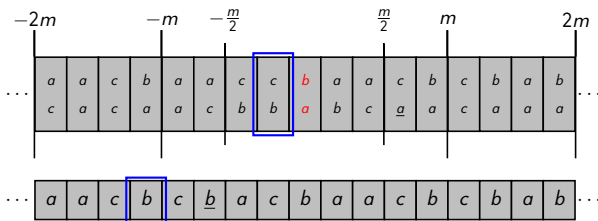
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



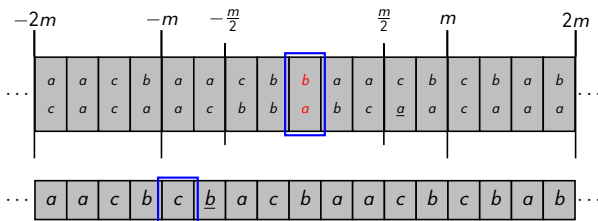
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



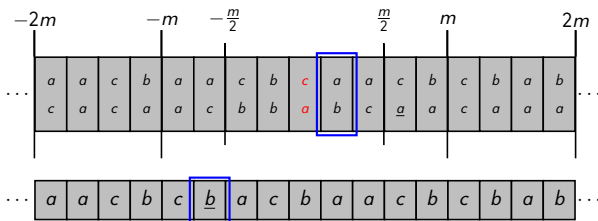
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



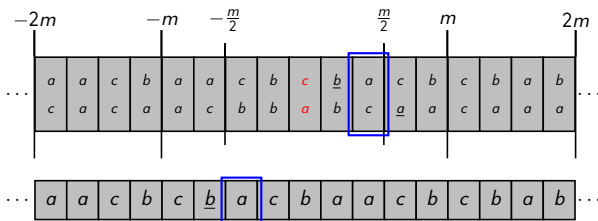
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



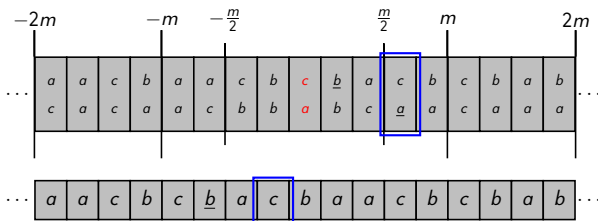
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



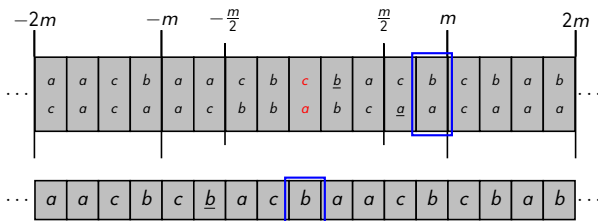
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



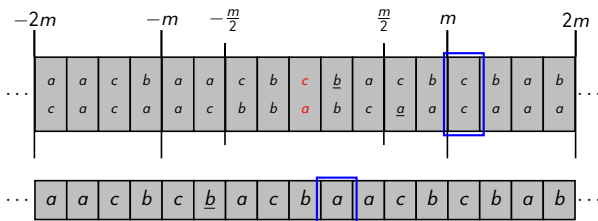
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



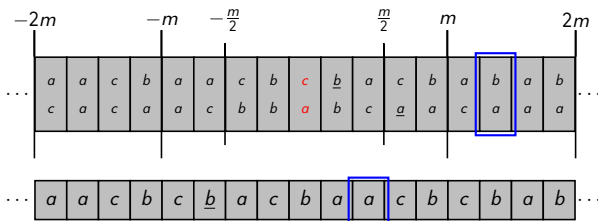
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



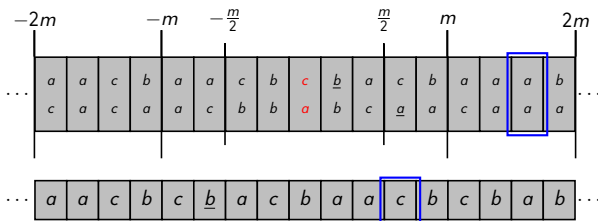
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



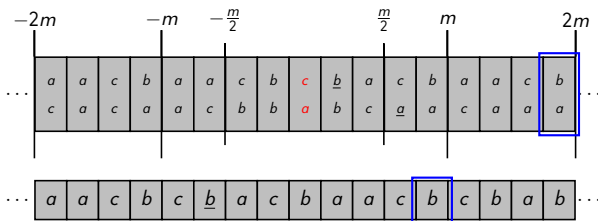
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



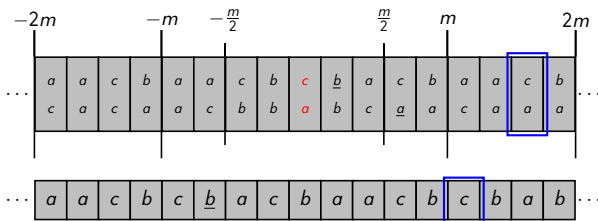
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



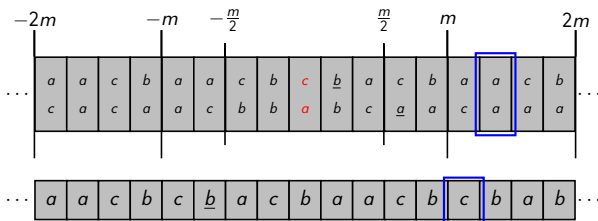
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



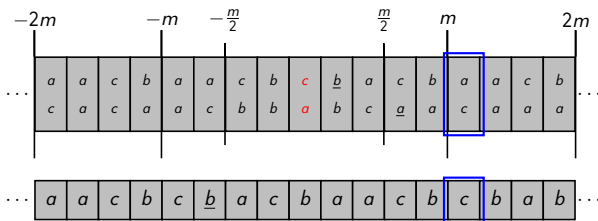
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



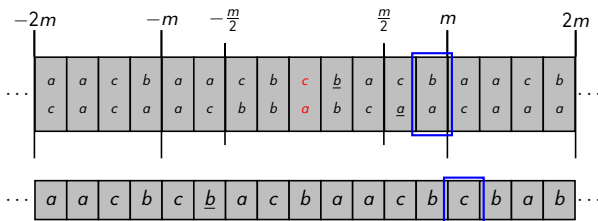
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



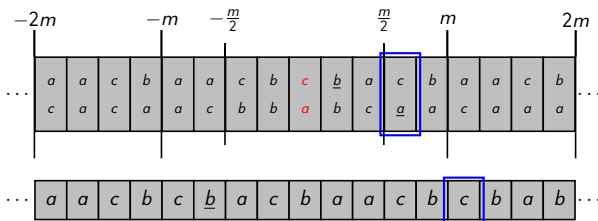
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



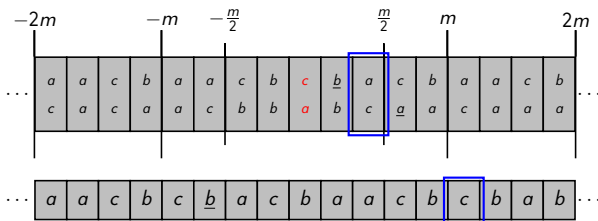
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



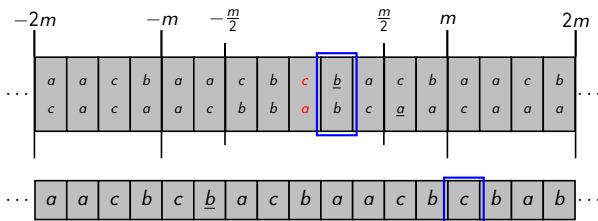
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



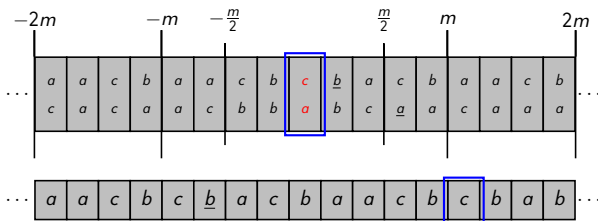
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



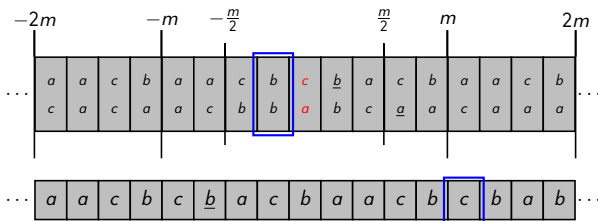
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



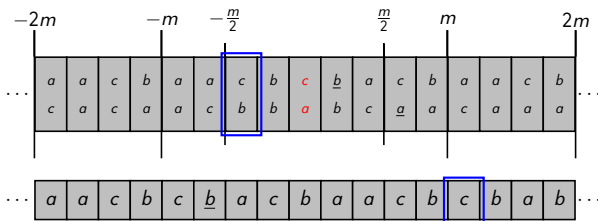
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



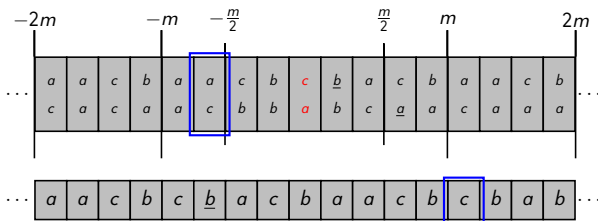
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



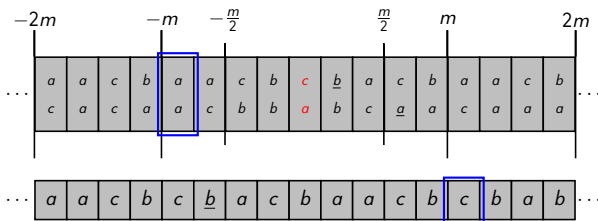
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



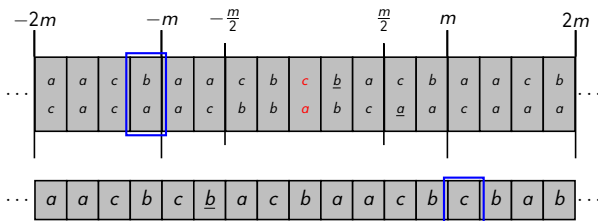
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



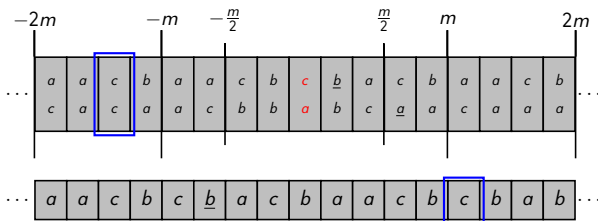
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



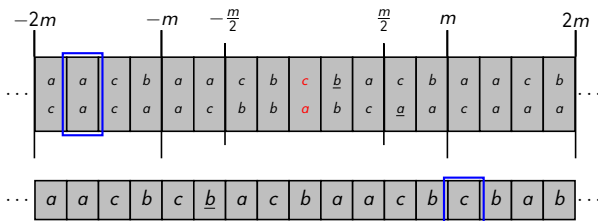
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



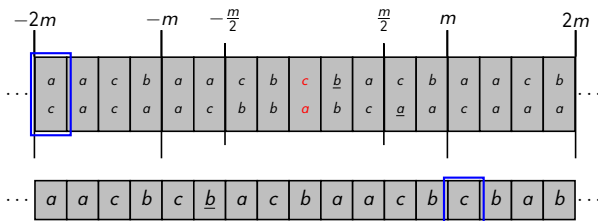
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



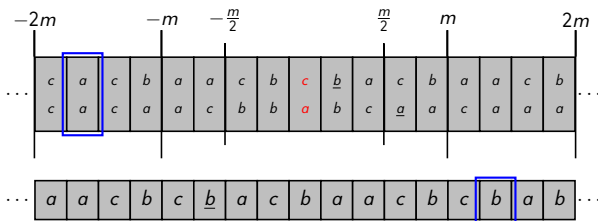
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



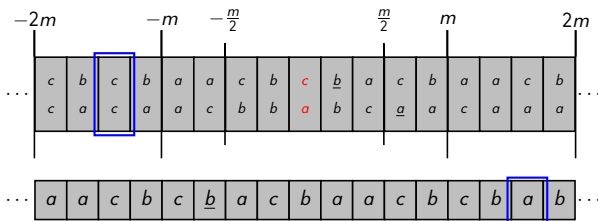
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



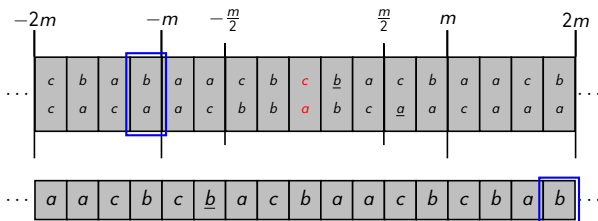
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



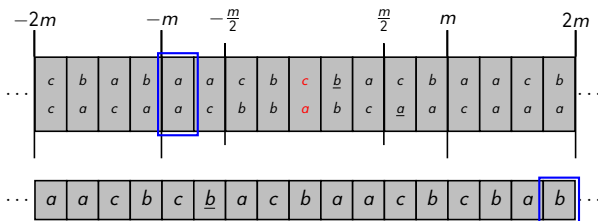
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



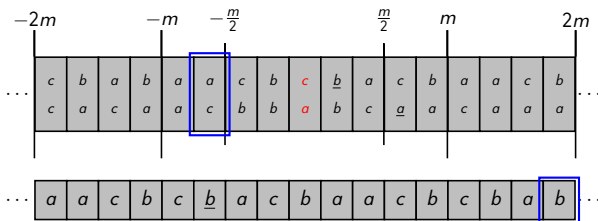
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



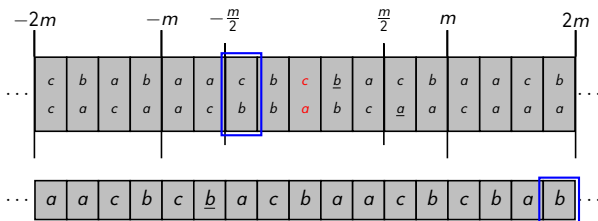
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



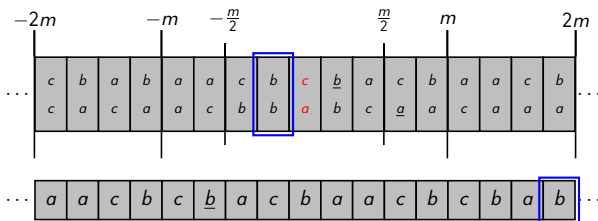
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



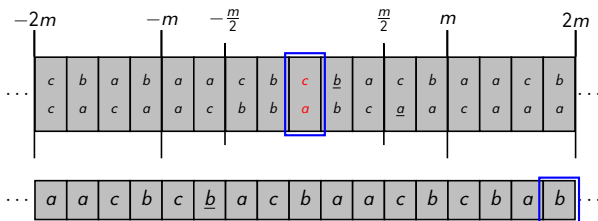
Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



Theorem of Hennie & Stearns

Illustration of centering by rotation of 1st simulated tape to right



Theorem of Hennie & Stearns

Proof (3/3)

Correctness of $\text{Simulate}(m)$ obvious. We use $\text{SimulateFromStart}(m)$ to indicate calls of $\text{Simulate}(m)$ on the initial configuration and run

- ① Let $m = 1$
- ② While M not stopped, execute $\text{SimulateFromStart}(m)$; $m \leftarrow 2m$

Theorem of Hennie & Stearns

Proof (3/3)

Correctness of $\text{Simulate}(m)$ obvious. We use $\text{SimulateFromStart}(m)$ to indicate calls of $\text{Simulate}(m)$ on the initial configuration and run

- ① Let $m = 1$
- ② While M not stopped, execute $\text{SimulateFromStart}(m) ; m \leftarrow 2m$

Let $t(m)$ be number of steps for call $\text{Simulate}(m)$. For sufficiently large m there is $k \in \mathbb{N}$ such that $t(m) \leq 2t(\frac{m}{2}) + km$ and thus $t \in O(x \log x)$.

Theorem of Hennie & Stearns

Proof (3/3)

Correctness of $\text{Simulate}(m)$ obvious. We use $\text{SimulateFromStart}(m)$ to indicate calls of $\text{Simulate}(m)$ on the initial configuration and run

- ① Let $m = 1$
- ② While M not stopped, execute $\text{SimulateFromStart}(m) ; m \leftarrow 2m$

Let $t(m)$ be number of steps for call $\text{Simulate}(m)$. For sufficiently large m there is $k \in \mathbb{N}$ such that $t(m) \leq 2t(\frac{m}{2}) + km$ and thus $t \in O(x \log x)$. Hence for large n there exists $k' \in \mathbb{N}$ such that overall time-consumption is

$$\sum_{i=0}^{\lceil \log f(n+2) \rceil} t(2^i) \leq k' \cdot \sum_{i=0}^{\lceil \log f(n+2) \rceil} 2^i \cdot i \leq 2k' \cdot f(n+2) \cdot \lceil \log f(n+2) \rceil$$

and thus in $O(f(x) \lceil \log f(x) \rceil)$. Applying linear speed-up, we obtain $f(x) \lceil \log f(x) \rceil$ -time-consumption. □

§3.4 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ with $f(n) \geq \log n$ for all $n \in \mathbb{N}$.
Then $\text{DSPACE}(f) \subseteq \text{NSPACE}(f) \subseteq \text{DTIME}(2^{O(f)})$

Proof

Only 2nd inclusion needs proof. Let M be f -space-bounded TM and w input of length n . By Lemma §1.7 there exists $g \in O(f)$ such that number of configurations reachable from initial configuration I_w is at most $2^{g(n+2)}$.

§3.4 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ with $f(n) \geq \log n$ for all $n \in \mathbb{N}$.
Then $\text{DSPACE}(f) \subseteq \text{NSPACE}(f) \subseteq \text{DTIME}(2^{O(f)})$

Proof

Only 2nd inclusion needs proof. Let M be f -space-bounded TM and w input of length n . By Lemma §1.7 there exists $g \in O(f)$ such that number of configurations reachable from initial configuration I_w is at most $2^{g(n+2)}$. We close set $\{I_w\}$ under successors and accept if and only if accepting configuration reached.

§3.4 Theorem

Let $f: \mathbb{N} \rightarrow \mathbb{N}$ with $f(n) \geq \log n$ for all $n \in \mathbb{N}$.
Then $\text{DSPACE}(f) \subseteq \text{NSPACE}(f) \subseteq \text{DTIME}(2^{O(f)})$

Proof

Only 2nd inclusion needs proof. Let M be f -space-bounded TM and w input of length n . By Lemma §1.7 there exists $g \in O(f)$ such that number of configurations reachable from initial configuration I_w is at most $2^{g(n+2)}$. We close set $\{I_w\}$ under successors and accept if and only if accepting configuration reached. Time-consumption is eventually proportional to $2^{g(n+2)} \cdot 2^{g(n+2)} \cdot g(n+2)$, which proves time-bound $2^{O(f)}$. (closure after $\leq 2^{g(n+2)}$ steps, search $\leq 2^{g(n+2)}$ elements for configuration with outside successor, configuration length $\leq g(n+2)$) $\square$

First conclusions

- $L \subseteq NL \subseteq DTIME(2^{O(\log x)}) = P$

(1st inclusion trivial and 2nd inclusion by Theorem §3.4)

First conclusions

- $L \subseteq NL \subseteq DTIME(2^{O(\log x)}) = P$

(1st inclusion trivial and 2nd inclusion by Theorem §3.4)

- $NL \subsetneq CSL = LBA = NSPACE(x) \subseteq DTIME(2^{O(x)})$

(**CSL** are context-sensitive languages &

LBA languages of linear-bounded TM

strictness of 1st inclusion by space hierarchy — Theorem §3.2

last inclusion by Theorem §3.4)

First conclusions

- $L \subseteq NL \subseteq DTIME(2^{O(\log x)}) = P$
(1st inclusion trivial and 2nd inclusion by Theorem §3.4)
- $NL \subsetneq CSL = LBA = NSPACE(x) \subseteq DTIME(2^{O(x)})$
(**CSL** are context-sensitive languages &
LBA languages of linear-bounded TM
strictness of 1st inclusion by space hierarchy — Theorem §3.2
last inclusion by Theorem §3.4)
- $DSPACE(x^2) \subseteq DTIME(2^{O(x^2)})$
(by Theorem §3.4)

Time Hierarchy Theorem

§3.5 Theorem (Time hierarchy theorem)

Let f time-constructible with $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(o(f)) \subsetneq \text{DTIME}(O(f; g))$ with $g(n) = n \log n$ for all $n \in \mathbb{N}$

Time Hierarchy Theorem

§3.5 Theorem (Time hierarchy theorem)

Let f time-constructible with $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(o(f)) \subsetneq \text{DTIME}(O(f; g))$ with $g(n) = n \log n$ for all $n \in \mathbb{N}$

Proof (1/2)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of det. k -TM $\mathcal{M}$ and diagonalize with time-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Time Hierarchy Theorem

§3.5 Theorem (Time hierarchy theorem)

Let f time-constructible with $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(o(f)) \subsetneq \text{DTIME}(O(f; g))$ with $g(n) = n \log n$ for all $n \in \mathbb{N}$

Proof (1/2)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of det. k -TM $\mathcal{M}$ and diagonalize with time-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

First prove $L \notin \text{DTIME}(o(f))$. Let $\mathcal{M}$ det. TM with $L(\mathcal{M}) = L$ that runs in time $o(f)$. Hence $\mathcal{M}$ runs in time $< f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2)$ on input $\langle \text{bin}(\mathcal{M}), w \rangle$ for sufficiently long w .

Time Hierarchy Theorem

§3.5 Theorem (Time hierarchy theorem)

Let f time-constructible with $f(n) \geq n$ for all $n \in \mathbb{N}$.

Then $\text{DTIME}(o(f)) \subsetneq \text{DTIME}(O(f; g))$ with $g(n) = n \log n$ for all $n \in \mathbb{N}$

Proof (1/2)

Select suitable coding $\text{bin}(\mathcal{M}) \in \{0, 1\}^*$ of det. k -TM $\mathcal{M}$ and diagonalize with time-bounded TMs.

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

First prove $L \notin \text{DTIME}(o(f))$. Let $\mathcal{M}$ det. TM with $L(\mathcal{M}) = L$ that runs in time $o(f)$. Hence $\mathcal{M}$ runs in time $< f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2)$ on input $\langle \text{bin}(\mathcal{M}), w \rangle$ for sufficiently long w . Then we obtain contradiction

$$\langle \text{bin}(\mathcal{M}), w \rangle \in L \iff \langle \text{bin}(\mathcal{M}), w \rangle \notin L(\mathcal{M})$$

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$.

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f time-constructible, we setup counter with value $f(n + 2)$ in time $f(n + 2)$ on 3rd tape.

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f time-constructible, we setup counter with value $f(n + 2)$ in time $f(n + 2)$ on 3rd tape. We simulate $\mathcal{M}$ and decrease counter on 3rd tape after each simulated step.

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f time-constructible, we setup counter with value $f(n + 2)$ in time $f(n + 2)$ on 3rd tape. We simulate $\mathcal{M}$ and decrease counter on 3rd tape after each simulated step.

- Reject if counter reaches 0 without halting since time-consumption exceeded

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f time-constructible, we setup counter with value $f(n + 2)$ in time $f(n + 2)$ on 3rd tape. We simulate $\mathcal{M}$ and decrease counter on 3rd tape after each simulated step.

- Reject if counter reaches 0 without halting since time-consumption exceeded
- Otherwise invert decision of $\mathcal{M}$

Time Hierarchy Theorem

$$L = \left\{ \langle \text{bin}(\mathcal{M}), w \rangle \mid \mathcal{M} \text{ rejects } \langle \text{bin}(\mathcal{M}), w \rangle \text{ in time } f(|\langle \text{bin}(\mathcal{M}), w \rangle| + 2) \right\}$$

Proof (2/2)

It remains to prove $L \in \text{DTIME}(O(f; g))$. Use universal 3-TM U that decodes input $\langle \text{bin}(\mathcal{M}), w \rangle$ and then simulates $\mathcal{M}$ according to Theorem §3.3 (Hennie & Stearns) on 2 tapes. Let $n = |\langle \text{bin}(\mathcal{M}), w \rangle|$. Since f time-constructible, we setup counter with value $f(n + 2)$ in time $f(n + 2)$ on 3rd tape. We simulate $\mathcal{M}$ and decrease counter on 3rd tape after each simulated step.

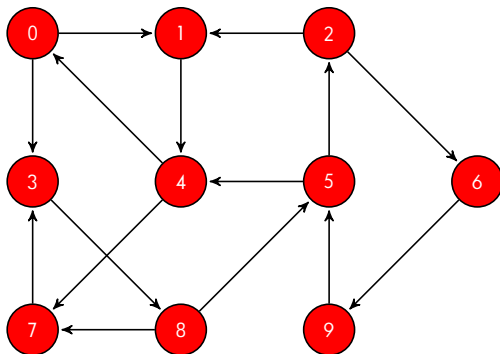
- Reject if counter reaches 0 without halting since time-consumption exceeded
- Otherwise invert decision of $\mathcal{M}$

Det. TM U accepts L in time $O(f; g)$ by Theorem §3.3. □

Graph Reachability

Decision variant GAP

- **Input:** finite directed graph (V, E) and vertices $s, t \in V$
- **Output:** Is there path from s to t ?



Algorithm of Savitch

- Every path of length $\leq 2^m$ splits into 2 paths of length $\leq 2^{m-1}$ and middle vertex
- Space-efficient (but time-inefficient) solution via recursion

Walter Savitch (* 1943)

- US computer scientist
- Complexity theory & programming languages
- Ph.D. student of Stephen Cook at UC Berkeley



Graph Reachability

Finite directed graph (V, E) and vertices $s, t \in V$

We implement predicate $\text{Reach}(v, i, v')$ for $v, v' \in V$ and $i \in \mathbb{N}$, which tells whether v' is reachable from v in 2^i steps

- 1 If $i = 0$, then return $(v = v') \vee ((v, v') \in E)$

Graph Reachability

Finite directed graph (V, E) and vertices $s, t \in V$

We implement predicate $\text{Reach}(v, i, v')$ for $v, v' \in V$ and $i \in \mathbb{N}$, which tells whether v' is reachable from v in 2^i steps

- ① If $i = 0$, then return $(v = v') \vee ((v, v') \in E)$
- ② Otherwise for all $v'' \in V$ (in arbitrary order)
 - Return **true**, if $\text{Reach}(v, i - 1, v'') \wedge \text{Reach}(v'', i - 1, v')$
- ③ Return **false**

Initial call $\text{Reach}(s, \lceil \log |V| \rceil, t)$

Graph Reachability

Finite directed graph (V, E) and vertices $s, t \in V$

We implement predicate $\text{Reach}(v, i, v')$ for $v, v' \in V$ and $i \in \mathbb{N}$, which tells whether v' is reachable from v in 2^i steps

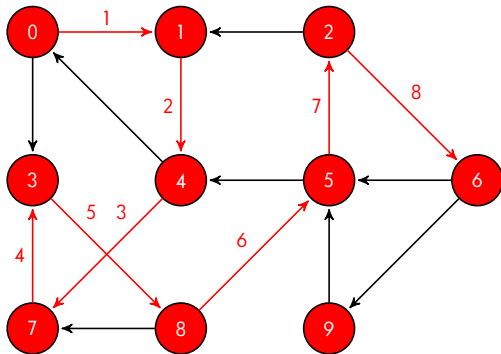
- ① If $i = 0$, then return $(v = v') \vee ((v, v') \in E)$
- ② Otherwise for all $v'' \in V$ (in arbitrary order)
 - ▶ Return **true**, if $\text{Reach}(v, i - 1, v'') \wedge \text{Reach}(v'', i - 1, v')$
- ③ Return **false**

Initial call $\text{Reach}(s, \lceil \log |V| \rceil, t)$

Input size

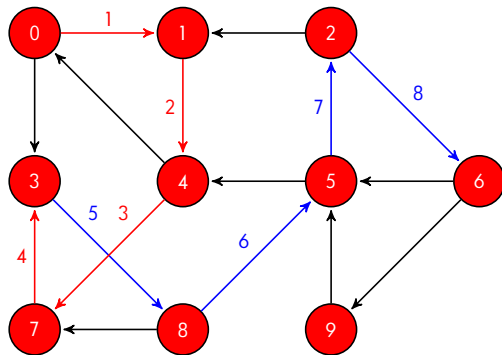
- $|V|^2 + 2 \log |V|$ as binary matrix (matrix & 2 vertices)
- $|E| \cdot (2 \log |V|) + 2 \log |V|$ as list of edges (list of edges & 2 vertices)

Graph Reachability



Open calls
 $\text{Reach}(0, 3, 6)$

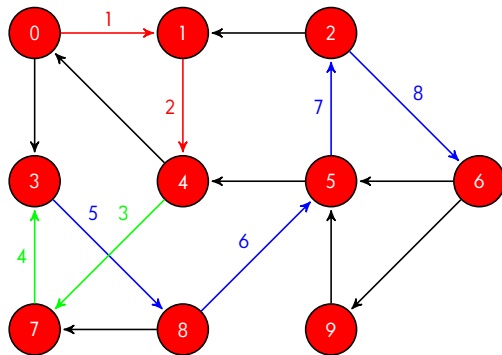
Graph Reachability



Open calls

$\text{Reach}(0, 3, 6)$ $\text{Reach}(0, 2, 3)$

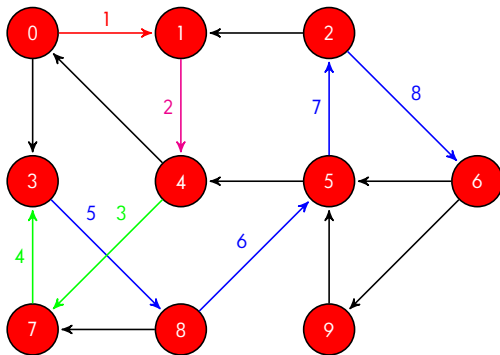
Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(0, 1, 4)

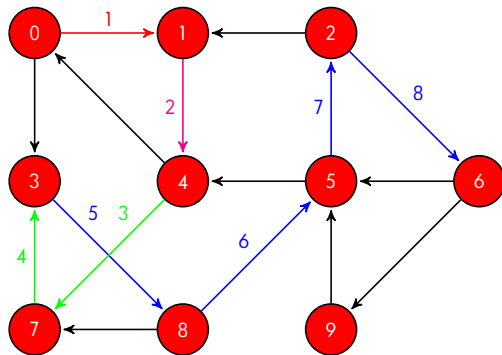
Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(0, 1, 4) Reach(0, 0, 1)

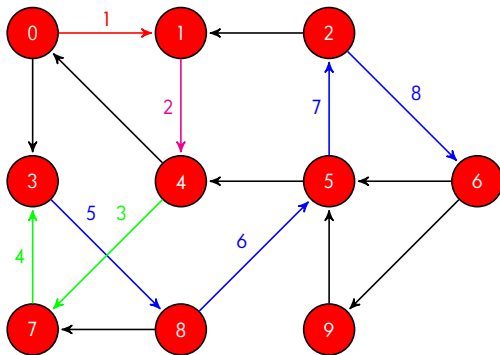
Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(0, 1, 4) Reach(1, 0, 4)

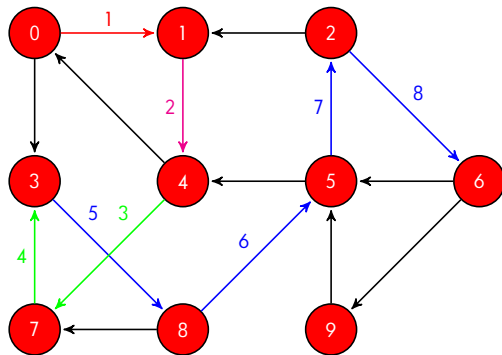
Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(4, 1, 3)

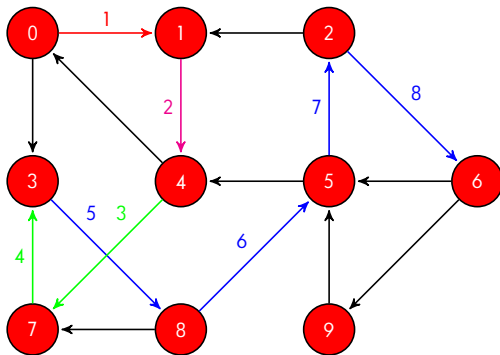
Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(4, 1, 3) Reach(4, 0, 7)

Graph Reachability



Open calls

Reach(0, 3, 6) Reach(0, 2, 3) Reach(4, 1, 3) Reach(7, 0, 3)

Graph Reachability

Proof of correctness

Let (V, E) graph and $s, t \in V$. There exists path from s to t iff such path of length $\leq |V|$ exists (visit at most each vertex once).

Graph Reachability

Proof of correctness

Let (V, E) graph and $s, t \in V$. There exists path from s to t iff such path of length $\leq |V|$ exists (visit at most each vertex once). We show invariant of call $\text{Reach}(v, i, v')$ by induction on i .

Induction base at $i = 0$ obvious.

Graph Reachability

Proof of correctness

Let (V, E) graph and $s, t \in V$. There exists path from s to t iff such path of length $\leq |V|$ exists (visit at most each vertex once). We show invariant of call $\text{Reach}(v, i, v')$ by induction on i .

Induction base at $i = 0$ obvious. Let $i \geq 1$ and $v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_m$ path of length $m \leq 2^i$ from $v_0 = v$ to $v_m = v'$. Then

$$v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_{\lfloor \frac{m}{2} \rfloor} \quad \text{and} \quad v_{\lfloor \frac{m}{2} \rfloor} \rightarrow v_{\lfloor \frac{m}{2} \rfloor + 1} \rightarrow \dots \rightarrow v_m$$

path from v to $v'' = v_{\lfloor \frac{m}{2} \rfloor}$ and from v'' to v' of length $\leq 2^{i-1}$ because

$$\lfloor \frac{m}{2} \rfloor \leq \frac{m}{2} \leq \frac{2^i}{2} = 2^{i-1} \quad \text{and} \quad m - \lfloor \frac{m}{2} \rfloor = \lceil \frac{m}{2} \rceil \leq \lceil \frac{2^i}{2} \rceil = 2^{i-1}$$

Graph Reachability

Proof of correctness

Let (V, E) graph and $s, t \in V$. There exists path from s to t iff such path of length $\leq |V|$ exists (visit at most each vertex once). We show invariant of call $\text{Reach}(v, i, v')$ by induction on i .

Induction base at $i = 0$ obvious. Let $i \geq 1$ and $v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_m$ path of length $m \leq 2^i$ from $v_0 = v$ to $v_m = v'$. Then

$$v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_{\lfloor \frac{m}{2} \rfloor} \quad \text{and} \quad v_{\lfloor \frac{m}{2} \rfloor} \rightarrow v_{\lfloor \frac{m}{2} \rfloor + 1} \rightarrow \dots \rightarrow v_m$$

path from v to $v'' = v_{\lfloor \frac{m}{2} \rfloor}$ and from v'' to v' of length $\leq 2^{i-1}$ because

$$\lfloor \frac{m}{2} \rfloor \leq \frac{m}{2} \leq \frac{2^i}{2} = 2^{i-1} \quad \text{and} \quad m - \lfloor \frac{m}{2} \rfloor = \lceil \frac{m}{2} \rceil \leq \lceil \frac{2^i}{2} \rceil = 2^{i-1}$$

By IH calls $\text{Reach}(v, i-1, v'')$ and $\text{Reach}(v'', i-1, v')$ return true, which proves invariant. If no path exists, then invariant also true. □

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$
 - ▶ If $v \neq w$, then delete successful call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as successful and reinspect call stack

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$
 - ▶ If $v \neq w$, then delete successful call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as successful and reinspect call stack
- When call not successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$
 - ▶ If $v \neq w$, then delete successful call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as successful and reinspect call stack
- When call not successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$ and $w' + 1$ valid, then replace failed call $\text{Reach}(w, k-1, w')$ by $\text{Reach}(w, k-1, w' + 1)$

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$
 - ▶ If $v \neq w$, then delete successful call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as successful and reinspect call stack
- When call not successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$ and $w' + 1$ valid, then replace failed call $\text{Reach}(w, k-1, w')$ by $\text{Reach}(w, k-1, w' + 1)$
 - ▶ If $v = w$ and $w' + 1$ invalid, then delete failed call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as failed and reinspect call stack

Graph Reachability

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- When call successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$, then delete successful call $\text{Reach}(w, k-1, w')$ and continue with call $\text{Reach}(w', k-1, v')$
 - ▶ If $v \neq w$, then delete successful call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as successful and reinspect call stack
- When call not successful, inspect last 2 calls on stack
 $\text{Reach}(v, k, v'), \text{Reach}(w, k-1, w')$
 - ▶ If $v = w$ and $w' + 1$ valid, then replace failed call $\text{Reach}(w, k-1, w')$ by $\text{Reach}(w, k-1, w' + 1)$
 - ▶ If $v = w$ and $w' + 1$ invalid, then delete failed call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as failed and reinspect call stack
 - ▶ If $v \neq w$ and $w + 1$ valid, then replace failed call $\text{Reach}(w, k-1, w')$ by $\text{Reach}(v, k-1, w + 1)$
 - ▶ If $v \neq w$ and $w + 1$ invalid, then delete failed call $\text{Reach}(w, k-1, w')$ and mark call $\text{Reach}(v, k, v')$ as failed and reinspect call stack

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- Neighboring entries differ in depth i

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- Neighboring entries differ in depth i
- At most $\log |V| + 1$ open calls

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- Neighboring entries differ in depth i
- At most $\log |V| + 1$ open calls
- Space complexity $3 \log |V| \cdot (\log |V| + 1) \in O(\log^2 x)$

Space complexity

- Store inputs of call v, i, v' in space $3 \log |V|$ since $i \leq \lceil \log |V| \rceil$
- Neighboring entries differ in depth i
- At most $\log |V| + 1$ open calls
- Space complexity $3 \log |V| \cdot (\log |V| + 1) \in O(\log^2 x)$

§3.6 Theorem

$\text{GAP} \in \text{DSpace}(\log^2 x)$

Summary

- Space hierarchy
- Theorem of Hennie & Stearns
- Time hierarchy & other relations between complexity classes
- Space-efficient graph reachability

Third exercise sheet available in Moodle