Lecture 21

Hierarchy Theorems

Machine as Strings and Universal TM

Machine as Strings and Universal TM

Recall:

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings.

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$.

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$.

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$. Moreover,

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$. Moreover, if M_α halts on x within T steps,

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$. Moreover, if M_α halts on x within T steps, then U on (α, x) halts in $CT \log T$ steps,

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$. Moreover, if M_α halts on x within T steps, then U on (α, x) halts in $CT \log T$ steps, where C depends

Machine as Strings and Universal TM

Recall:

- $\langle M \rangle$ denotes the encoding of M.
- M_α denotes the TM that binary string α represents.
- Every TM is represented by infinitely many strings. (Add 1s in the end that are ignored.)

output of M_α on x

Theorem: There exists a TM U such that $\forall x, \alpha \in \{0,1\}^*$, $U(x, \alpha) = M_\alpha(x)$. Moreover, if M_α halts on x within T steps, then U on (α, x) halts in $CT \log T$ steps, where C depends only on $M'_\alpha s$ alphabet size, number of tapes, and number of states.

Diagonalization

Diagonalization

Diagonalization is any technique that relies solely upon the following properties of TMs:

Diagonalization

Diagonalization is any technique that relies solely upon the following properties of TMs:

- The existence of an effective representation of Turing machines by strings.

Diagonalization

Diagonalization is any technique that relies solely upon the following properties of TMs:

- The existence of an effective representation of Turing machines by strings.
- The ability of one TM to simulate any another without much overhead in running time or space.

Diagonalization

Diagonalization is any technique that relies solely upon the following properties of TMs:

- The existence of an effective representation of Turing machines by strings.
- The ability of one TM to simulate any another without much overhead in running time or space.

Diagonalization is used in **Hierarchy theorems** and **Ladner's theorem**.

Deterministic Time Hierarchy

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying

$f(n) \log f(n) = o(g(n))$

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathsf{DTIME}(f(n)) \subset \mathsf{DTIME}(g(n))$.

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\text{DTIME}(f(n)) \subset \text{DTIME}(g(n))$.

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\text{DTIME}(f(n)) \subset \text{DTIME}(g(n))$.

***o*-notation:**

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

o**-notation:** $\forall c > 0, \exists n_0 > 0$

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then DTIME$(f(n)) \subset$ DTIME$(g(n))$.

o**-notation:** $\forall c > 0, \exists n_0 > 0$ such that $c \cdot f(n) \log f(n) < g(n)$,

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\text{DTIME}(f(n)) \subset \text{DTIME}(g(n))$.

***o*-notation:** $\forall c > 0, \exists n_0 > 0$ such that $c \,.\, f(n) \log f(n) < g(n)$, for all $n > n_0$.

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\text{DTIME}(f(n)) \subset \text{DTIME}(g(n))$.

...

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof:

...

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

...

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time,

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \rightarrow \mathbb{N}, g : \mathbb{N} \rightarrow \mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Let $L(D)$ denote the language decided by D.

…

Deterministic Time Hierarchy

Theorem: If $f:\mathbb{N}\to\mathbb{N}, g:\mathbb{N}\to\mathbb{N}$ are time-constructible functions satisfying $f(n)\log f(n) = o(g(n))$, then $\mathrm{DTIME}(f(n)) \subset \mathrm{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Let $L(D)$ denote the language decided by D.

Claim 1: $L(D) \in \mathrm{DTIME}(g(n))$.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then DTIME$(f(n)) \subset$ DTIME$(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Let $L(D)$ denote the language decided by D.

Claim 1: $L(D) \in$ DTIME$(g(n))$.

Claim 2: $L(D) \notin$ DTIME$(f(n))$.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then DTIME$(f(n)) \subset$ DTIME$(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Let $L(D)$ denote the language decided by D.

Claim 1**:** $L(D) \in$ DTIME$(g(n))$. **Proof:** By definition … using time-constructibility…

Claim 2**:** $L(D) \notin$ DTIME$(f(n))$.

…

Deterministic Time Hierarchy

Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n) \log f(n) = o(g(n))$, then $\text{DTIME}(f(n)) \subset \text{DTIME}(g(n))$.

Proof: Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Let $L(D)$ denote the language decided by D.

Claim 1: $L(D) \in \text{DTIME}(g(n))$. **Proof:** By definition … using time-constructibility…

Claim 2: $L(D) \notin \text{DTIME}(f(n))$. **Proof:** By contradiction …

…

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \mathsf{DTIME}(f(n))$.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \mathsf{DTIME}(f(n))$.

Proof:

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$.

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin \mathrm{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin \mathrm{DTIME}(f(n))$.

Proof: Suppose ∃ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin \mathsf{DTIME}(f(n))$.

Proof: Suppose ∃ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin$ DTIME$(f(n))$.

Proof: Suppose ∃ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

Recall D**:**

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin$ DTIME$(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

Recall D**:**

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin$ DTIME$(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.

Recall D**:**

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.
- If M_x accepts x,

Recall D:

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.
- If M_x accepts x,
- If M_x rejects x,

Recall D:

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin$ DTIME$(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.
- If M_x accepts x, then D rejects x.
- If M_x rejects x,

Recall D**:**

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2**:** $L(D) \notin \mathrm{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.
- If M_x accepts x, then D rejects x.
- If M_x rejects x, then D accepts x.

Recall D**:**

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

Deterministic Time Hierarchy

Claim 2: $L(D) \notin \text{DTIME}(f(n))$.

Proof: Suppose $\exists$ a DTM M with running time $O(f(n))$ that decides $L(D)$.

- M on any input x halts within $cf(|x|)$ steps, where c is a constant.
- UTM U can simulate M on input x within $c'f(|x|)\log f(|x|)$ steps, where c' is a constant…

We know that $\exists n_0$ such that $\forall n \geq n_0$, $c'f(n)\log f(n) < g(n)$. $(\because f(n)\log f(n) = o(g(n)))$

Let x be a binary representation of M whose length is at least n_0.

What happens when D gets x as input?

- M_x halts on x within $g(|x|)$ steps of U.
- If M_x accepts x, then D rejects x.
- If M_x rejects x, then D accepts x.

Recall D:

Consider a DTM D that on input x:

- Runs UTM U on (x, x) for $g(|x|)$ steps.
- If M_x halts on x and writes some bits on the output tape within this time, then D outputs the opposite of the first bit.
- Else, D outputs 0.

■

Other Hierarchy Theorems

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n+1) = o(g(n))$, then

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n+1) = o(g(n))$, then

$$\mathrm{NTIME}(f(n)) \subset \mathrm{NTIME}(g(n))$$

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n+1) = o(g(n))$, then

$$\text{NTIME}(f(n)) \subset \text{NTIME}(g(n))$$

Space Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are space-constructible functions

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n+1) = o(g(n))$, then

$$\text{NTIME}(f(n)) \subset \text{NTIME}(g(n))$$

Space Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are space-constructible functions satisfying $f(n) = o(g(n))$, then

Other Hierarchy Theorems

Nondeterministic Time Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are time-constructible functions satisfying $f(n+1) = o(g(n))$, then

$$\text{NTIME}(f(n)) \subset \text{NTIME}(g(n))$$

Space Hierarchy Theorem: If $f : \mathbb{N} \to \mathbb{N}, g : \mathbb{N} \to \mathbb{N}$ are space-constructible functions satisfying $f(n) = o(g(n))$, then

$$\text{DSPACE}(f(n)) \subset \text{DSPACE}(g(n))$$