CMPE 322/327 - Theory of Computation

Week 12: Halting Problem & Reduction

Burak Ekici

May 16-20, 2022

Outline

1 A Quick Recap

2 Halting Problem

3 Reduction

Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

1. Q: finite set of states

Definition

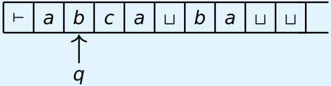
Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

1. Q: finite set of states
2. Σ: input alphabet

Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

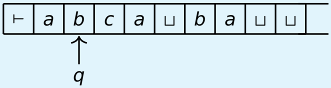
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet



Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker



Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

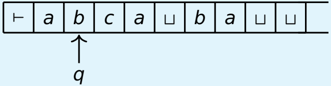
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol

$\vdash$	a	b	c	a	$\sqcup$	b	a	$\sqcup$	$\sqcup$
		$\uparrow$							
		q							

Definition

Turning machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

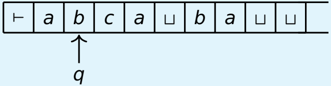
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function



Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

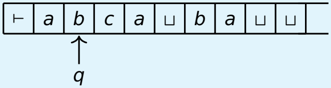
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function
7. $s \in Q$: start state



Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

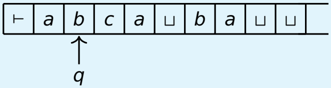
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function
7. $s \in Q$: start state
8. $t \in Q$: accept state



Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

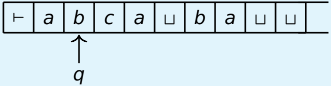
1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function
7. $s \in Q$: start state
8. $t \in Q$: accept state
9. $r \in Q - \{t\}$: reject state



Definition

Turning machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function
7. $s \in Q$: start state
8. $t \in Q$: accept state
9. $r \in Q - \{t\}$: reject state



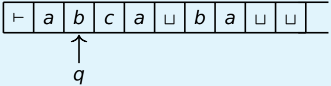
such that

$$\forall\, a \in \Gamma \quad \exists\, b, c \in \Gamma \quad \exists\, d, e \in \{L, R\} : \ \delta(t, a) = (t, b, d) \quad \text{and} \quad \delta(r, a) = (r, c, e)$$

Definition

Turing machine (TM) is 9-tuple $M = (Q, \Sigma, \Gamma, \vdash, \sqcup, \delta, s, t, r)$ with

1. Q: finite set of states
2. Σ: input alphabet
3. $\Gamma \supseteq \Sigma$: tape alphabet
4. $\vdash \in \Gamma - \Sigma$: left endmarker
5. $\sqcup \in \Gamma - \Sigma$: blank symbol
6. $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$: (partial) transition function
7. $s \in Q$: start state
8. $t \in Q$: accept state
9. $r \in Q - \{t\}$: reject state



such that

$$\forall\, a \in \Gamma \;\; \exists\, b, c \in \Gamma \;\; \exists\, d, e \in \{L, R\}\colon\; \delta(t, a) = (t, b, d) \quad \text{and} \quad \delta(r, a) = (r, c, e)$$

$$\forall\, p \in Q \;\; \exists\, q \in Q\colon\; \delta(p, \vdash) = (q, \vdash, R)$$

Decision Problems

- halting problem for TMs

instance:	TM M, string x
question:	does M halt on x?

Decision Problems

- halting problem for TMs

  instance: TM M, string x
  question: does M halt on x?

- uniform halting problem for TMs

  instance: TM M
  question: does M halt on all inputs?

Decision Problems

- halting problem for TMs

  instance: TM M, string x
  question: does M halt on x?

- uniform halting problem for TMs

  instance: TM M
  question: does M halt on all inputs?

- membership problem for CFGs

  instance: CFG G, string x
  question: $x \in L(G)$?

Decision Problems

- halting problem for TMs

  instance: TM M, string x
  question: does M halt on x?

- uniform halting problem for TMs

  instance: TM M
  question: does M halt on all inputs?

- membership problem for CFGs

  instance: CFG G, string x
  question: $x \in L(G)$?

- equivalence problem for regular expressions

  instance: regular expressions α, β
  question: $L(\alpha) = L(\beta)$?

Decision Problems

- halting problem for TMs

  instance: TM M, string x
  question: does M halt on x?

- uniform halting problem for TMs

  instance: TM M
  question: does M halt on all inputs?

- membership problem for CFGs

  instance: CFG G, string x
  question: $x \in L(G)$?

- equivalence problem for regular expressions

  instance: regular expressions α, β
  question: $L(\alpha) = L(\beta)$?

Decision Problems as Membership Problems

- code instance of problem as string over some alphabet

Decision Problems

- halting problem for TMs

  instance: TM M, string x
  question: does M halt on x?

- uniform halting problem for TMs

  instance: TM M
  question: does M halt on all inputs?

- membership problem for CFGs

  instance: CFG G, string x
  question: $x \in L(G)$?

- equivalence problem for regular expressions

  instance: regular expressions α, β
  question: $L(\alpha) = L(\beta)$?

Decision Problems as Membership Problems

- code instance of problem as string over some alphabet
- language is set of all strings that correspond to yes instances

Definition (Encoding of Membership Problem for Turing Machines)

$\text{MP} = \{\text{enc}(M) \# \text{enc}(x) \mid x \in L(M)\}$

Definition (Encoding of Membership Problem for Turing Machines)

MP $= \{ \quad M \; \# \quad x \; \mid x \in L(M) \}$

Definition (Encoding of Membership Problem for Turing Machines)

$\mathrm{MP} = \{ \quad M \ \# \quad x \ \mid x \in L(M) \}$

Theorem (Universal Turing Machine)

$\exists$ TM U such that $L(U) = \mathrm{MP}$

Definition (Encoding of Membership Problem for Turing Machines)

$\mathrm{MP} = \{ \; M \; \# \; x \; \mid x \in L(M)\}$

Theorem (Universal Turing Machine)

$\exists$ TM U such that $L(U) = \mathrm{MP}$

universal TM can simulate any TM

Definition (Encoding of Membership Problem for Turing Machines)

$\mathrm{MP} = \{ \quad M \; \# \quad x \mid x \in L(M) \}$

Theorem (Universal Turing Machine)

$\exists$ TM U such that $L(U) = \mathrm{MP}$

universal TM can simulate any TM

Corollary

MP is r.e.

How to Prove Undecidability?

- diagonalization
- reduction

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \dots\}$ is countable

$$
\begin{array}{c|cccc}
 & A_0 & A_1 & A_2 & \cdots \\
\hline
x_0 & & & & \\
x_1 & & & & \\
x_2 & & & & \\
\vdots & & & &
\end{array}
$$

Theorem

$\Sigma \neq \emptyset \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable

	A_0	A_1	A_2	$\cdots$
x_0				
x_1				
x_2				
$\vdots$				

Theorem

$\Sigma \neq \emptyset \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

$$
\begin{array}{c|cccc}
 & A_0 & A_1 & A_2 & \cdots \\
\hline
x_0 & & & & \\
x_1 & & & & \\
x_2 & & & & \\
\vdots & & & &
\end{array}
$$

Theorem

$\Sigma \neq \emptyset \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\times$			
x_1	$\checkmark$			
x_2	$\times$			
$\vdots$				

- $A_0 = \{x_1\}$

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\times$	$\checkmark$		
x_1	$\checkmark$	$\times$		
x_2	$\times$	$\checkmark$		
$\vdots$				

- $A_0 = \{x_1\}$ $\quad A_1 = \{x_0, x_2, x_7, x_8, \ldots\}$

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\times$	$\checkmark$	$\times$	
x_1	$\checkmark$	$\times$	$\times$	
x_2	$\times$	$\checkmark$	$\checkmark$	
$\vdots$				

- $A_0 = \{x_1\}$ $A_1 = \{x_0, x_2, x_7, x_8, \ldots\}$ $A_2 = \{x_2, x_5\}$

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\times$	$\checkmark$	$\times$	
x_1	$\checkmark$	$\times$	$\times$	
x_2	$\times$	$\checkmark$	$\checkmark$	
$\vdots$				

- $A_0 = \{x_1\} \quad A_1 = \{x_0, x_2, x_7, x_8, \ldots\} \quad A_2 = \{x_2, x_5\}$
- define $B = \{x_i \mid x_i \notin A_i\} \in 2^{\Sigma^*}$

Theorem

$\Sigma \neq \varnothing \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\checkmark$	$\checkmark$	$\times$	
x_1	$\checkmark$	$\checkmark$	$\times$	
x_2	$\times$	$\checkmark$	$\times$	
$\vdots$				

- $A_0 = \{x_1\}$ $\quad A_1 = \{x_0, x_2, x_7, x_8, \ldots\}$ $\quad A_2 = \{x_2, x_5\}$
- define $B = \{x_i \mid x_i \notin A_i\} \in 2^{\Sigma^*}$
- $B \neq A_i \ \forall\, i$

Theorem

$\Sigma \neq \emptyset \implies 2^{\Sigma^*}$ is uncountable

Proof. (Diagonalization)

- suppose $2^{\Sigma^*} = \{A_0, A_1, A_2, \ldots\}$ is countable
- $\Sigma^* = \{x_0, x_1, x_2, \ldots\}$ is countable
- create infinite table

	A_0	A_1	A_2	$\cdots$
x_0	$\checkmark$	$\checkmark$	$\times$	
x_1	$\checkmark$	$\checkmark$	$\times$	
x_2	$\times$	$\checkmark$	$\times$	
$\vdots$				

- $A_0 = \{x_1\}$ $\quad A_1 = \{x_0, x_2, x_7, x_8, \ldots\}$ $\quad A_2 = \{x_2, x_5\}$
- define $B = \{x_i \mid x_i \notin A_i\} \in 2^{\Sigma^*}$
- $B \neq A_i \ \forall i$ ↯

Outline

1 A Quick Recap

2 Halting Problem

3 Reduction

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose $\text{HP} = \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. (   )

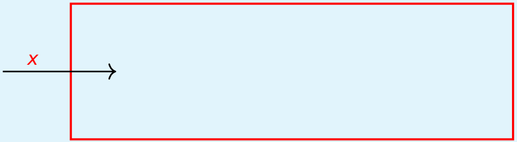
- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$

Theorem

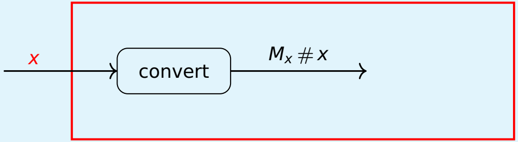
halting problem (HP) for TMs is undecidable (not recursive)

Proof. (   )

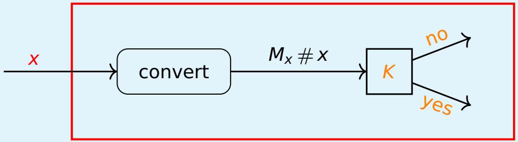
- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts



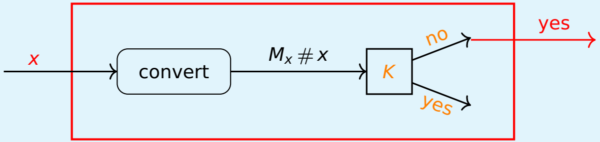
design of TM N



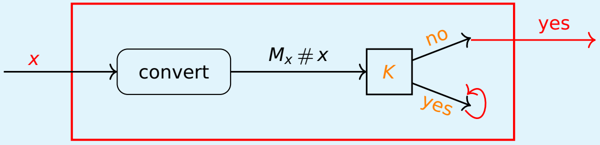
design of TM N



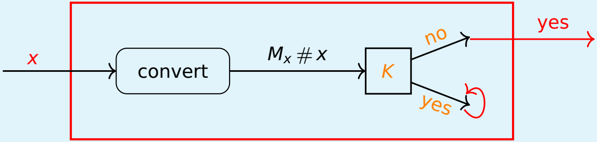
design of TM N



design of TM N



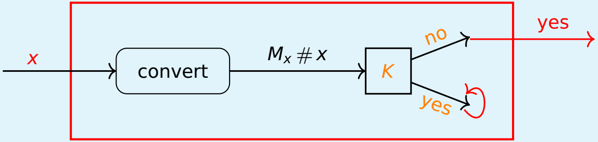
design of TM N



design of TM N

	ε	0	1	00	01	10	$\cdots$
M_ε	$\times$	$\checkmark$	$\times$	$\times$	$\checkmark$	$\times$	$\cdots$
M_0	$\checkmark$	$\checkmark$	$\times$	$\times$	$\times$	$\checkmark$	
M_1	$\times$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	
M_{00}	$\times$	$\checkmark$	$\checkmark$	$\times$	$\checkmark$	$\times$	
M_{01}	$\times$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	
M_{10}	$\checkmark$	$\times$	$\checkmark$	$\times$	$\times$	$\times$	
$\vdots$	$\vdots$						

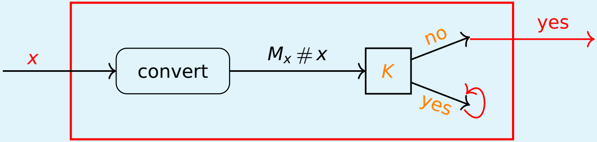
$\checkmark$: halts
$\times$: loops



design of TM N

	ε	0	1	00	01	10	$\cdots$
M_ε	$\times$	$\checkmark$	$\times$	$\times$	$\checkmark$	$\times$	$\cdots$
M_0	$\checkmark$	$\checkmark$	$\times$	$\times$	$\times$	$\checkmark$	
M_1	$\times$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	
M_{00}	$\times$	$\checkmark$	$\checkmark$	$\times$	$\checkmark$	$\times$	
M_{01}	$\times$	$\times$	$\checkmark$	$\times$	$\checkmark$	$\times$	
M_{10}	$\checkmark$	$\times$	$\checkmark$	$\times$	$\times$	$\times$	
$\vdots$	$\vdots$						

$\checkmark$: accepts
$\times$: rejects



design of TM N

	ε	0	1	00	01	10	$\cdots$
M_ε	$\checkmark$	$\checkmark$	$\times$	$\times$	$\checkmark$	$\times$	$\cdots$
M_0	$\checkmark$	$\times$	$\times$	$\times$	$\times$	$\checkmark$	
M_1	$\times$	$\times$	$\times$	$\checkmark$	$\checkmark$	$\checkmark$	
M_{00}	$\times$	$\checkmark$	$\checkmark$	$\checkmark$	$\checkmark$	$\times$	
M_{01}	$\times$	$\times$	$\checkmark$	$\times$	$\times$	$\times$	
M_{10}	$\checkmark$	$\times$	$\checkmark$	$\times$	$\times$	$\checkmark$	
$\vdots$	$\vdots$						

$\checkmark$: halts

$\times$: loops

$L(N) = \{\varepsilon, 00, 10, \ldots\}$

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose $\mathrm{HP} = \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- $\mathrm{HP} = L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts
- for all inputs x $\quad N$ halts on $x \iff K$ rejects $M_x \# x$

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts
- for all inputs x $\quad N$ halts on $x \iff K$ rejects $M_x \# x \iff M_x$ does not halt on x

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose $\mathrm{HP} = \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- $\mathrm{HP} = L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts
- for all inputs x $\quad N$ halts on $x \iff K$ rejects $M_x \# x \iff M_x$ does not halt on x
- N is different from all M_x

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. ( )

- suppose $\text{HP} = \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- $\text{HP} = L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts
- for all inputs x $\quad N$ halts on $x \iff K$ rejects $M_x \# x \iff M_x$ does not halt on x
- N is different from all M_x ↯

Theorem

halting problem (HP) for TMs is undecidable (not recursive)

Proof. (diagonalization)

- suppose HP $= \{M \# x \mid \text{TM } M \text{ halts on input } x\}$ is recursive
- HP $= L(K)$ for some total TM K
- construct TM N that on input x
  - constructs M_x from x
  - runs K on input $M_x \# x$
  - accepts if K rejects and loops if K accepts
- for all inputs x $\quad N$ halts on $x \iff K$ rejects $M_x \# x \iff M_x$ does not halt on x
- N is different from all M_x $\quad \lightning$

Membership problem for TMs

instance: TM M, string x
question: does M accept x ?

Membership problem for TMs

instance:	TM M, string x
question:	does M accept x ?

Theorem

1. membership problem for TMs is undecidable:

   $\mathrm{MP} = \{M \# x \mid x \in L(M)\}$ is not recursive

Membership problem for TMs

instance: TM M, string x
question: does M accept x ?

Theorem

1. membership problem for TMs is undecidable:

   $\mathrm{MP} = \{M \# x \mid x \in L(M)\}$ is not recursive

2. membership problem for TMs is semi-decidable (MP is r.e.):

   $\mathrm{MP} = L(U)$

Membership problem for TMs

instance: TM M, string x
question: does M accept x ?

Theorem

1. membership problem for TMs is undecidable:

   $\mathrm{MP} = \{M \# x \mid x \in L(M)\}$ is not recursive

2. membership problem for TMs is semi-decidable (MP is r.e.):

   $\mathrm{MP} = L(U)$

Proof.

1. reduction from halting problem for TMs

Membership problem for TMs

instance: TM M, string x
question: does M accept x ?

Theorem

1. membership problem for TMs is undecidable:

   $\mathrm{MP} = \{M \# x \mid x \in L(M)\}$ is not recursive

2. membership problem for TMs is semi-decidable (MP is r.e.):

   $\mathrm{MP} = L(U)$

Proof.

1. reduction from halting problem for TMs (next section)

Outline

1. A Quick Recap
2. Halting Problem
3. Reduction

Definition

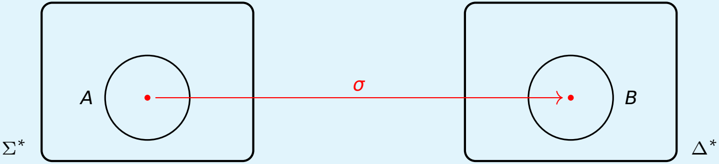
(many-one) reduction of set $A \subseteq \Sigma^*$ to set $B \subseteq \Delta^*$ is total computable function $\sigma\colon \Sigma^* \to \Delta^*$ such that

$$x \in A \iff \sigma(x) \in B$$

Definition

(many-one) reduction of set $A \subseteq \Sigma^*$ to set $B \subseteq \Delta^*$ is total computable function $\sigma\colon \Sigma^* \to \Delta^*$ such that

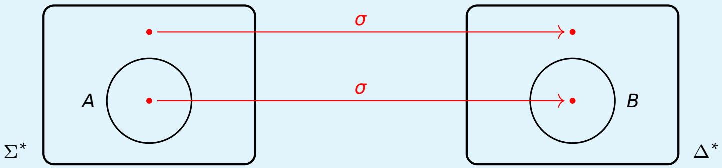
$$x \in A \quad \Longleftrightarrow \quad \sigma(x) \in B$$



Definition

(many-one) reduction of set $A \subseteq \Sigma^*$ to set $B \subseteq \Delta^*$ is total computable function $\sigma \colon \Sigma^* \to \Delta^*$ such that

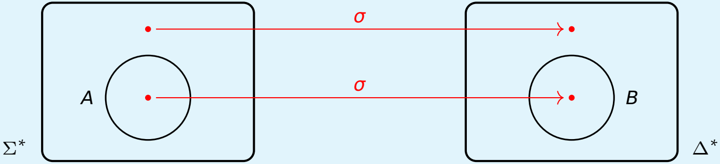
$$x \in A \iff \sigma(x) \in B$$



Definition

(many-one) reduction of set $A \subseteq \Sigma^*$ to set $B \subseteq \Delta^*$ is total computable function $\sigma\colon \Sigma^* \to \Delta^*$ such that

$$x \in A \iff \sigma(x) \in B$$



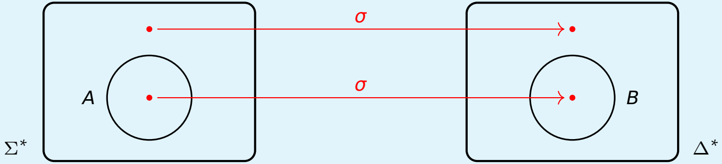
Notation

$A \leq_m B$

Definition

(many-one) reduction of set $A \subseteq \Sigma^*$ to set $B \subseteq \Delta^*$ is total computable function $\sigma\colon \Sigma^* \to \Delta^*$ such that

$$x \in A \iff \sigma(x) \in B$$



Notation

$A \leq_m B$ $\qquad$ $\sigma\colon A \to_m B$

Theorem

if $A \leq_{\mathrm{m}} B$ and B is r.e. then A is r.e.

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

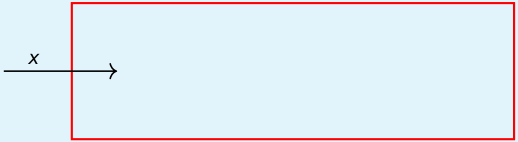
- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$

Theorem

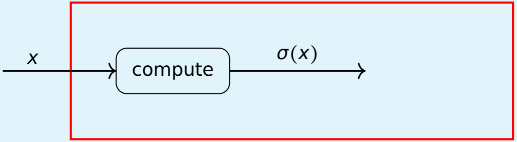
if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

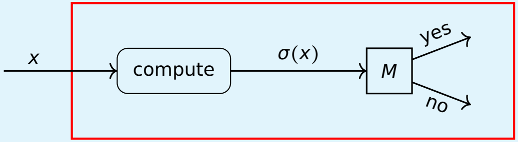
- $\sigma : A \rightarrow_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects



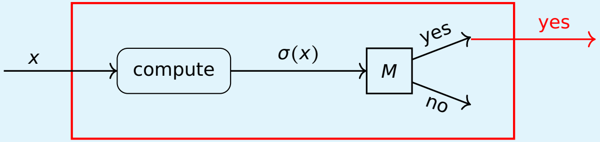
design of TM N



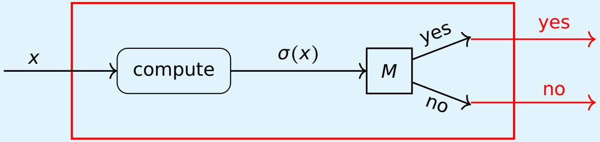
design of TM N



design of TM N



design of TM N



design of TM N

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x)$$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x) \iff \sigma(x) \in B$$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x) \iff \sigma(x) \in B \iff x \in A$$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x) \iff \sigma(x) \in B \iff x \in A$$

- $L(N) = A$

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x) \iff \sigma(x) \in B \iff x \in A$$

- $L(N) = A \implies A$ is r.e.

Theorem

if $A \leq_m B$ and B is r.e. then A is r.e.

Proof.

- $\sigma : A \to_m B$
- $B = L(M)$ for some TM M
- construct TM N that on input x
  1. computes $\sigma(x)$
  2. runs M on input $\sigma(x)$
  3. accepts if M accepts and rejects if M rejects
- for all inputs x

$$N \text{ accepts } x \iff M \text{ accepts } \sigma(x) \iff \sigma(x) \in B \iff x \in A$$

- $L(N) = A \implies A$ is r.e.

Theorem (Equivalent)

if $A \leq_m B$ and A is not r.e. then B is not r.e.

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Proof.

- $\sim A \ \leq_m \sim B$

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Proof.

- $\sim A \leq_m \sim B$
- B and $\sim B$ are r.e.

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Proof.

- $\sim A \ \leq_m \sim B$
- B and $\sim B$ are r.e.
- A and $\sim A$ are r.e. (by previous theorem)

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Proof.

- $\sim A \ \leq_m \sim B$
- B and $\sim B$ are r.e.
- A and $\sim A$ are r.e. (by previous theorem)
- A is recursive

Theorem

if $A \leq_m B$ and B is recursive then A is recursive

Proof.

- $\sim A \ \leq_m \sim B$
- B and $\sim B$ are r.e.
- A and $\sim A$ are r.e. (by previous theorem)
- A is recursive

Theorem (Equivalent)

if $A \leq_m B$ and A is not recursive then B is not recursive

Theorem

HP $\leq_m$ MP

Theorem

HP $\leq_m$ MP

Proof.

reduction σ: HP $\to_m$ MP

Theorem

HP $\leq_m$ MP

Proof.

reduction σ: HP $\to_m$ MP

- σ transforms $M \# x$ into $N \# x$

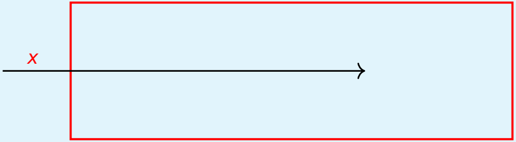
Theorem

$\mathrm{HP} \le_m \mathrm{MP}$

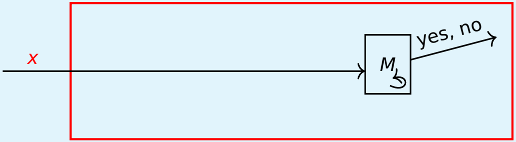
Proof.

reduction $\sigma\colon \mathrm{HP} \to_m \mathrm{MP}$

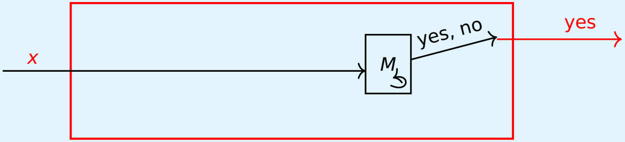
- σ transforms $M \# x$ into $N \# x$
- TM N is exactly like M but it accepts if M accepts or rejects



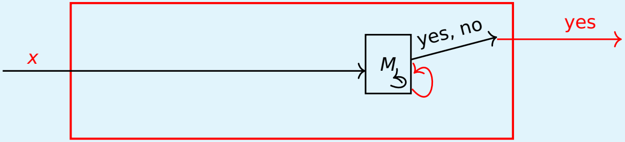
design of TM N



design of TM N



design of TM N



design of TM N

Theorem

HP $\leq_m$ MP

Proof.

reduction $\sigma\colon \mathrm{HP} \to_m \mathrm{MP}$

- σ transforms $M \# x$ into $N \# x$
- TM N is exactly like M but it accepts if M accepts or rejects
- $M \# x \in \mathrm{HP} \iff M$ halts on x

Theorem

HP $\leq_m$ MP

Proof.

reduction σ: HP $\rightarrow_m$ MP

- σ transforms $M \# x$ into $N \# x$
- TM N is exactly like M but it accepts if M accepts or rejects
- $M \# x \in$ HP $\iff$ M halts on x

  $\iff$ N accepts x

Theorem

HP $\leq_m$ MP

Proof.

reduction $\sigma\colon \mathrm{HP} \to_m \mathrm{MP}$

- σ transforms $M \# x$ into $N \# x$
- TM N is exactly like M but it accepts if M accepts or rejects
- $M\#x \in \mathrm{HP} \iff M$ halts on x

$$\iff N \text{ accepts } x$$

$$\iff N\#x \in \mathrm{MP}$$

Theorem

HP $\leq_m$ MP

Proof.

reduction $\sigma\colon \mathrm{HP} \to_m \mathrm{MP}$

- σ transforms $M \# x$ into $N \# x$
- TM N is exactly like M but it accepts if M accepts or rejects
- $$\begin{aligned} M\#x \in \mathrm{HP} &\iff M \text{ halts on } x \\ &\iff N \text{ accepts } x \\ &\iff N\#x \in \mathrm{MP} \\ &\iff \sigma(M\#x) \in \mathrm{MP} \end{aligned}$$

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. A $\leq_m$ B and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. $A \leq_m B$ and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. $A \leq_m B$ and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. A $\leq_m$ B and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. A $\leq_m$ B and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)
5. HP $\leq_m$ MP (by theorem on slides #17 – 18)

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. $A \leq_m B$ and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. $A \leq_m B$ and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)
5. $HP \leq_m MP$ (by theorem on slides #17 – 18)

Theorem

MP is not recursive

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. $A \leq_m B$ and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. $A \leq_m B$ and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)
5. $HP \leq_m MP$ (by theorem on slides #17 – 18)

Theorem

MP is not recursive

Proof.

by statements 5, contrapositive of 4 and 2

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. $A \le_m B$ and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. $A \le_m B$ and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)
5. $HP \le_m MP$ (by theorem on slides #17 – 18)

Theorem

MP is not recursive

Proof.

by statements 5, contrapositive of 4 and 2

Theorem

HP is r.e.

All Together

1. MP is r.e. (by corollary in w11.pdf on slide #29)
2. HP is not recursive (by theorem on slide #9)
3. $A \leq_m B$ and B is r.e $\implies$ A is r.e. (by the first theorem on slides #14 – 15)
4. $A \leq_m B$ and B is recursive $\implies$ A is recursive (by the first theorem on slide #16)
5. $HP \leq_m MP$ (by theorem on slides #17 – 18)

Theorem

MP is not recursive

Proof.

by statements 5, contrapositive of 4 and 2

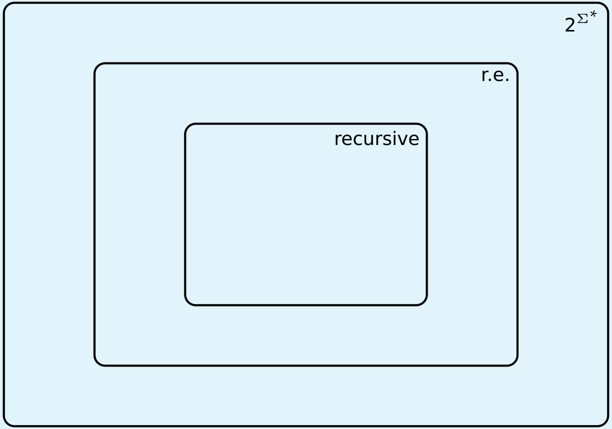
Theorem

HP is r.e.

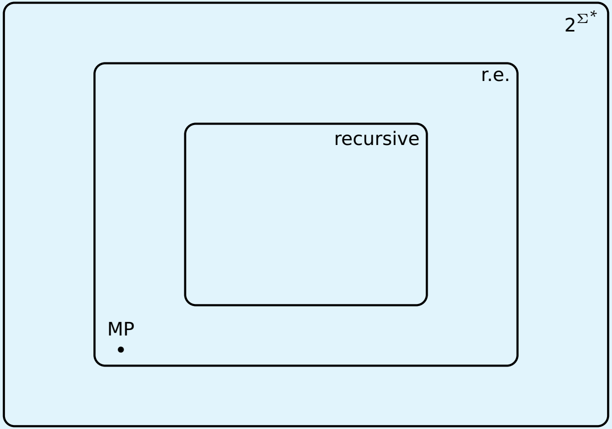
Proof.

by statements 5, 3 and 1

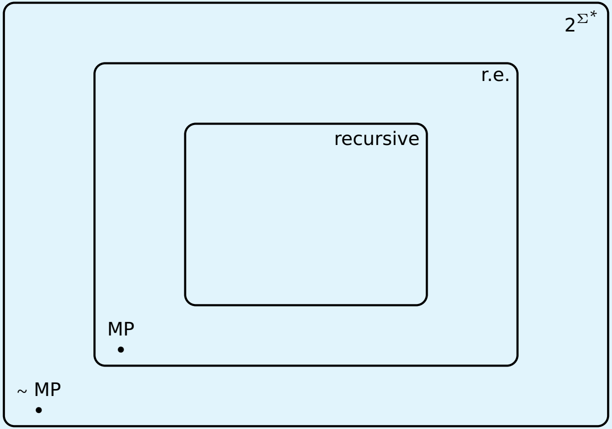
Summary



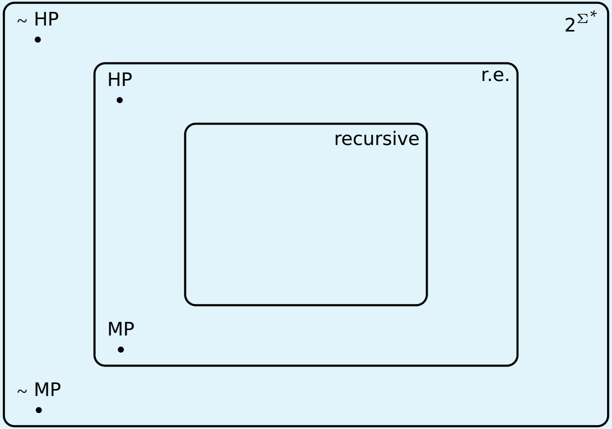
Summary



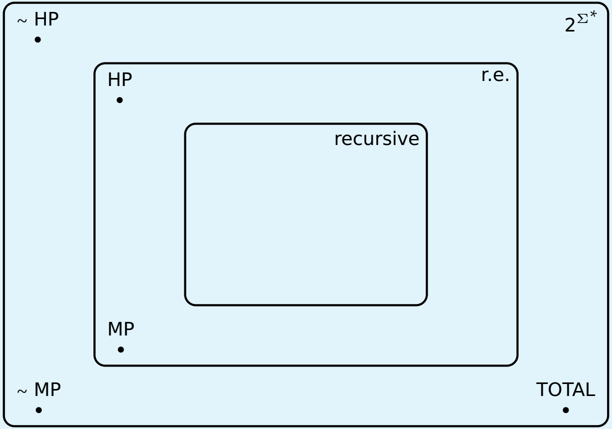
Summary



Summary



Summary



TOTAL = $\{M \mid \text{TM } M \text{ halts on all inputs}\}$

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance:	TM M
question:	is $L(M)$ finite ?

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance:	TM M
question:	is $L(M)$ finite ?

Proof.

$\mathrm{A} \leq_{\mathrm{m}} \mathrm{B}$ and B is r.e $\implies$ A is r.e.

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance:	TM M
question:	is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides $\#14 - 15$)
- $A := \sim \mathrm{HP}$
  $B := \mathrm{FIN} = \{M \mid L(M) \text{ is finite}\}$

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides $\#14-15$)
- A := $\sim$ HP
  B := FIN $= \{M \mid L(M)$ is finite$\}$
- $\sim$ HP $\leq_m$ FIN and $\sim$ HP is not r.e. $\implies$ FIN is not r.e.

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- A := $\sim$ HP
  B := FIN $= \{M \mid L(M) \text{ is finite}\}$
- $\sim$ HP $\leq_m$ FIN and $\sim$ HP is not r.e. $\implies$ FIN is not r.e.

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- A := $\sim$ HP
  B := FIN $= \{M \mid L(M) \text{ is finite}\}$
- $\sim$ HP $\leq_m$ FIN and True $\implies$ FIN is not r.e.

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- A := $\sim$ HP
  B := FIN $= \{M \mid L(M) \text{ is finite}\}$
- $\sim$ HP $\leq_m$ FIN $\implies$ FIN is not r.e.

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance:	TM M
question:	is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \mathrm{HP}$
  $B := \mathrm{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim$ HP $\leq_m$ FIN $\implies$ FIN is not r.e.

reduction σ: $\sim$ HP $\to_m$ FIN

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \mathrm{HP}$
  $B := \mathrm{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim \mathrm{HP} \leq_m \mathrm{FIN} \implies$ FIN is not r.e.

reduction $\sigma\colon \sim \mathrm{HP} \to_m \mathrm{FIN}$

- σ transforms $M \# x$ into TM N

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \text{HP}$
  $B := \text{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim \text{HP} \leq_m \text{FIN} \implies \text{FIN}$ is not r.e.

reduction $\sigma\colon \sim \text{HP} \to_m \text{FIN}$

- σ transforms $M\#x$ into TM N that
  1. erases its input

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \mathrm{HP}$
  $B := \mathrm{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim$ HP $\leq_m$ FIN $\implies$ FIN is not r.e.

reduction σ: $\sim$ HP $\rightarrow_m$ FIN

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim\text{HP}$
  $B := \text{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim\text{HP} \leq_m \text{FIN} \implies$ FIN is not r.e.

reduction $\sigma\colon \sim\text{HP} \to_m \text{FIN}$

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape
  3. runs M on input x

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \text{HP}$
  $B := \text{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim \text{HP} \leq_m \text{FIN} \implies \text{FIN}$ is not r.e.

reduction $\sigma : \sim \text{HP} \to_m \text{FIN}$

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape
  3. runs M on input x
  4. accepts if M halts on x

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- A := $\sim$ HP
  B := FIN $= \{M \mid L(M)$ is finite$\}$
- $\sim$ HP $\leq_m$ FIN $\implies$ FIN is not r.e.

reduction σ: $\sim$ HP $\rightarrow_m$ FIN

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape
  3. runs M on input x
  4. accepts if M halts on x
- $M \# x \in \sim$ HP $\iff$ M does not halt on x

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \text{HP}$
  $B := \text{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim \text{HP} \leq_m \text{FIN} \implies$ FIN is not r.e.

reduction $\sigma : \sim \text{HP} \to_m \text{FIN}$

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape
  3. runs M on input x
  4. accepts if M halts on x
- $M \# x \in \sim \text{HP} \iff M$ does not halt on $x \iff L(N) = \varnothing$

Theorem

finiteness problem for TMs is not semi-decidable (is not r.e.):

instance: TM M
question: is $L(M)$ finite ?

Proof.

A $\leq_m$ B and B is r.e $\implies$ A is r.e.

- A $\leq_m$ B and A is not r.e $\implies$ B is not r.e. (by the second theorem on slides #14 – 15)
- $A := \sim \mathrm{HP}$
  $B := \mathrm{FIN} = \{M \mid L(M) \text{ is finite}\}$
- $\sim \mathrm{HP} \leq_m \mathrm{FIN} \implies \mathrm{FIN}$ is not r.e.

reduction $\sigma\colon \sim \mathrm{HP} \to_m \mathrm{FIN}$

- σ transforms $M \# x$ into TM N that
  1. erases its input
  2. writes x on its tape
  3. runs M on input x
  4. accepts if M halts on x
- $M \# x \in \sim \mathrm{HP} \iff M$ does not halt on $x \iff L(N) = \varnothing \iff N \in \mathrm{FIN}$

Thanks! & Questions?