CS310 Automata Theory – 2016-2017

Nutan Limaye

Indian Institute of Technology, Bombay
nutan@cse.iitb.ac.in

Lecture 29: Turing machines, computability
March 28, 2017

At the end of last class

Introduction to Turing machines

- Undecidability of the following languages:
  - $A_{TM} = \{(M, w) \mid M \text{ accepts } w\}$.

At the end of last class

Introduction to Turing machines

Undecidability of the following languages:

$A_{TM} = \{(M, w) \mid M \text{ accepts } w\}$.

$\text{Halt} = \{(M, w) \mid M \text{ hants on } w\}$.

At the end of last class

Introduction to Turing machines

Undecidability of the following languages:

$A_{TM} = \{(M, w) \mid M \text{ accepts } w\}$.

$\text{Halt} = \{(M, w) \mid M \text{ hants on } w\}$.

$E_{TM} = \{\langle M \rangle \mid L(M) = \varnothing\}$.

At the end of last class

Introduction to Turing machines

Undecidability of the following languages:

$A_{TM} = \{(M, w) \mid M \text{ accepts } w\}$.

$\text{Halt} = \{(M, w) \mid M \text{ hants on } w\}$.

$E_{TM} = \{\langle M \rangle \mid L(M) = \varnothing\}$.

$EQ_{TM} = \{(M_1, M_2) \mid L(M_1) = L(M_2)\}$.

At the end of last class

Introduction to Turing machines

- Undecidability of the following languages:
  - $A_{TM} = \{(M, w) \mid M \text{ accepts } w\}$.
  - $\text{Halt} = \{(M, w) \mid M \text{ hants on } w\}$.
  - $E_{TM} = \{\langle M \rangle \mid L(M) = \varnothing\}$.
  - $EQ_{TM} = \{(M_1, M_2) \mid L(M_1) = L(M_2)\}$.
  - $REG_{TM} = \{\langle M \rangle \mid L(M) \text{ is regular}\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

$\{M \mid L(M) \text{ is context-free}\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

$\{M \mid L(M) \text{ is context-free}\}$.

$\{M \mid L(M) = \varnothing\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

$\{M \mid L(M) \text{ is context-free}\}$.

$\{M \mid L(M) = \varnothing\}$.

The following languages are decidable.

$\{M \mid M \text{ has more than 10 states}\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

$\{M \mid L(M) \text{ is context-free}\}$.

$\{M \mid L(M) = \varnothing\}$.

The following languages are decidable.

$\{M \mid M \text{ has more than 10 states}\}$.

$\{M \mid M \text{ never moves left on any input}\}$.

Rice's theorem

The following languages are undecidable.

$\{M \mid L(M) \text{ is regular}\}$.

$\{M \mid L(M) \text{ is context-free}\}$.

$\{M \mid L(M) = \varnothing\}$.

The following languages are decidable.

$\{M \mid M \text{ has more than 10 states}\}$.

$\{M \mid M \text{ never moves left on any input}\}$.

Rice's theorem: A systematic way of proving undecidability of languages.

Property P

Definition

A property P is simply a subset of Turing recognizable languages.

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages.

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Set of context-free languages.

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Set of context-free languages. ✓

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Set of context-free languages. ✓

$\{\varnothing\}$.

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Set of context-free languages. ✓

$\{\varnothing\}$. ✓

Property P

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Examples

Set of regular languages. ✓

Set of context-free languages. ✓

$\{\varnothing\}$. ✓

Property P and $\mathcal{L}_P$

Definition

A property P is simply a subset of Turing recognizable languages.

Property P and $\mathcal{L}_P$

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

Property P and $\mathcal{L}_P$

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

For any property P, let $\mathcal{L}_P = \{M \mid L(M) \in P\}$

Property P and $\mathcal{L}_P$

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

For any property P, let $\mathcal{L}_P = \{M \mid L(M) \in P\}$, i.e. the set of all Turing machine such that $L(M) \in P$.

Property P and $\mathcal{L}_P$

Definition

A property P is simply a subset of Turing recognizable languages. We say that a language L satisfies a property P, if $L \in P$.

For any property P, let $\mathcal{L}_P = \{M \mid L(M) \in P\}$, i.e. the set of all Turing machine such that $L(M) \in P$.

We say that a property P is trivial if either $\mathcal{L}_P = \varnothing$ or $\mathcal{L}_P$ is the set of all the Turing recognizable languages.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$?

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM accepting a^*b^* is in $\mathcal{L}_P$.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM accepting $a^* b^*$ is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$?

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM accepting $a^* b^*$ is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$? No.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM accepting a^*b^* is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$? No. For example, a TM accepting $\{a^n b^n \mid n \geq 0\}$ is not in $\mathcal{L}_P$.

Examples of properties

Examples

1. $\mathcal{L}_P = \{M \mid L(M) \text{ is regular}\}$.

   $\mathcal{L}_P$ is collection of TMs M such that $L(M)$ is regular.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM accepting a^*b^* is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$? No. For example, a TM accepting $\{a^nb^n \mid n \geq 0\}$ is not in $\mathcal{L}_P$.

   Therefore, P is not trivial.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

Is $\mathcal{L}_P = \varnothing$?

Examples of properties

Examples

2. $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

   Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

   Is $\mathcal{L}_P = \varnothing$? No.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

Is $\mathcal{L}_P = \varnothing$? No. For example, a TM M that rejects any string is in $\mathcal{L}_P$.

Examples of properties

Examples

2. $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

   Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM M that rejects any string is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$?

Examples of properties

Examples

2. $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

   Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

   Is $\mathcal{L}_P = \varnothing$? No. For example, a TM M that rejects any string is in $\mathcal{L}_P$.

   Is $\mathcal{L}_P = \Sigma^*$? No.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

Is $\mathcal{L}_P = \varnothing$? No. For example, a TM M that rejects any string is in $\mathcal{L}_P$.

Is $\mathcal{L}_P = \Sigma^*$? No. For example, a TM M that accepts a single string $\{a\}$ is not in $\mathcal{L}_P$.

Examples of properties

Examples

2 $\mathcal{L}_P = \{M \mid L(M) = \varnothing\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that $L(M) = \varnothing$.

Is $\mathcal{L}_P = \varnothing$? No. For example, a TM M that rejects any string is in $\mathcal{L}_P$.

Is $\mathcal{L}_P = \Sigma^*$? No. For example, a TM M that accepts a single string $\{a\}$ is not in $\mathcal{L}_P$.

Example of a trivial property

Examples

3. $\mathcal{L}_P = \left\{ M \;\middle|\; \begin{array}{c} M \text{ is a TM and } L(M) \text{ is accepted by} \\ \text{a TM that has even number of states} \end{array} \right\}.$

Example of a trivial property

Examples

3. $\mathcal{L}_P = \left\{ M \;\middle|\; \begin{array}{c} M \text{ is a TM and } L(M) \text{ is accepted by} \\ \text{a TM that has even number of states} \end{array} \right\}.$

Here P is a property of Turing recognizable languages.

Example of a trivial property

Examples

3. $\mathcal{L}_P = \left\{ M \;\middle|\; \begin{array}{c} M \text{ is a TM and } L(M) \text{ is accepted by} \\ \text{a TM that has even number of states} \end{array} \right\}.$

Here P is a property of Turing recognizable languages.

But any TM can be converted into another one that has even number of states.

Example of a trivial property

Examples

3. $\mathcal{L}_P = \left\{ M \;\middle|\; \begin{array}{c} M \text{ is a TM and } L(M) \text{ is accepted by} \\ \text{a TM that has even number of states} \end{array} \right\}.$

Here P is a property of Turing recognizable languages.

But any TM can be converted into another one that has even number of states.

Therefore, any Turing recognizable language has property P.

Example of a trivial property

Examples

3. $\mathcal{L}_P = \left\{ M \;\middle|\; \begin{array}{c} M \text{ is a TM and } L(M) \text{ is accepted by} \\ \text{a TM that has even number of states} \end{array} \right\}.$

Here P is a property of Turing recognizable languages.

But any TM can be converted into another one that has even number of states.

Therefore, any Turing recognizable language has property P.

Therefore, P is in fact all Turing recognizable languages.

Examples of properties

Examples

$$P = \{M \mid M \text{ has more than 10 states}\}.$$

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}.$

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

This is not a property of $L(M)$, but the property of M itself.

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

This is not a property of $L(M)$, but the property of M itself.

Consider a language L which is accepted by a TM with less than 10 states.

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

This is not a property of $L(M)$, but the property of M itself.

Consider a language L which is accepted by a TM with less than 10 states.

For L we can create another TM M' such that $L(M') = L$, but M' has more than 10 states.

Examples of properties

Examples

$P = \{M \mid M \text{ has more than 10 states}\}$.

Here $\mathcal{L}_P$ is a collection of TMs M such that M has more than 10 states.

This is not a property of $L(M)$, but the property of M itself.

Consider a language L which is accepted by a TM with less than 10 states.

For L we can create another TM M' such that $L(M') = L$, but M' has more than 10 states.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$$\{\langle M\rangle \mid M \text{ has at least ten states}\}.$$

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$\{\langle M \rangle \mid M \text{ has at least ten states}\}$.

$\{\langle M \rangle \mid M \text{ never moves left on any input string } \}$.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$\{\langle M\rangle \mid M \text{ has at least ten states}\}$.

$\{\langle M\rangle \mid M \text{ never moves left on any input string }\}$.

$\{\langle M\rangle \mid M \text{ has no useless state }\}$.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$\{\langle M \rangle \mid M$ has at least ten states$\}$.

$\{\langle M \rangle \mid M$ never moves left on any input string $\}$.

$\{\langle M \rangle \mid M$ has no useless state $\}$.

To prove non-recognizability of a property of languages.

Rice's theorem cannot be used to prove non-recognizability of languages.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$\{\langle M\rangle \mid M$ has at least ten states$\}$.

$\{\langle M\rangle \mid M$ never moves left on any input string $\}$.

$\{\langle M\rangle \mid M$ has no useless state $\}$.

To prove non-recognizability of a property of languages.

Rice's theorem cannot be used to prove non-recognizability of languages.

It is only used to prove undecidability.

Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

When is the theorem NOT applicable?

$\{\langle M\rangle \mid M$ has at least ten states$\}$.

$\{\langle M\rangle \mid M$ never moves left on any input string $\}$.

$\{\langle M\rangle \mid M$ has no useless state $\}$.

To prove non-recognizability of a property of languages.

Rice's theorem cannot be used to prove non-recognizability of languages.

It is only used to prove undecidability.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid M \text{ runs for atmost 10 steps on } aab\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid M$ runs for atmost 10 steps on $aab\}$.

Not applicable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M\rangle \mid M$ runs for atmost 10 steps on $aab\}$.

Not applicable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with at least 10 states}\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M\rangle \mid L(M)$ is recognized a TM with at least 10 states$\}$.

Applicable, but property is trivial.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M \rangle \mid L(M) \text{ is recognized a TM with at least 10 states}\}.$$

Applicable, but property is trivial.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M \rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applicable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid L(M)$ is recognized a TM with atmost 10 states$\}$.

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid L(M)$ is recognized a TM with atmost 10 states$\}$.

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid L(M)$ is recognized a TM with atmost 10 states$\}$.

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

In that case, the property is trivial.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

In that case, the property is trivial.

Textbooks usually consider this property to be not trivial.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

In that case, the property is trivial.

Textbooks usually consider this property to be not trivial.

This is because the usual assumption is that you always fix the tape alphabet.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

In that case, the property is trivial.

Textbooks usually consider this property to be not trivial.

This is because the usual assumption is that you always fix the tape alphabet.
In that case, Rice's theorem is applicable and the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid L(M) \text{ is recognized a TM with atmost 10 states}\}.$$

Applicable.
If we can simulate any TM with another with less than 10 states, then the property will be trivial.

This is doable if we allow for the tape alphabet size to grow.

In that case, the property is trivial.

Textbooks usually consider this property to be not trivial.

This is because the usual assumption is that you always fix the tape alphabet.
In that case, Rice's theorem is applicable and the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{\langle M\rangle \mid M \text{ has at most 10 states}\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid M$ has at most 10 states$\}$.

Not applicable, but the language is decidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{\langle M \rangle \mid M$ has at most 10 states$\}$.

Not applicable, but the language is decidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid L(M) \text{ contains } \langle M \rangle\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M \rangle\}$.

Applicable, the property is not trivial

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M \rangle\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M \rangle\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid L(M) \text{ contains } \langle M \rangle\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M \rangle\}$.

Applicable, the property is not trivial

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M \rangle\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ contains } \langle M\rangle\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ is finite }\}$.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ is finite }\}$.

Applicable, the property is not trivial

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ is finite }\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) \text{ is finite }\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid L(M) = \Sigma^*\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid L(M) = \Sigma^*\}.$$

Applicable, the property is not trivial

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid L(M) = \Sigma^*\}.$$

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid L(M) = \Sigma^*\}$.

Applicable, the property is not trivial, therefore undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{M \mid M \text{ has a useless state }\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid M$ has a useless state $\}$.

Not applicable, but the language is in fact undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{M \mid M$ has a useless state $\}$.

Not applicable, but the language is in fact undecidable.

Rice's theorem cannot be used to prove the undecidability of this language!

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{(M, w) \mid M \text{ writes a symbol a on the tape on input } w\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\{(M, w) \mid M \text{ writes a symbol a on the tape on input } w\}.$$

Not applicable, but the language is in fact undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$\{(M, w) \mid M$ writes a symbol a on the tape on input $w\}$.

Not applicable, but the language is in fact undecidable.

Rice's theorem cannot be used to prove the undecidability of this language!

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\left\{ M \;\middle|\; \begin{array}{l} M \text{ tries to write on the left of the cell when it} \\ \text{is at the leftmost bit of the input} \end{array} \right\}.$$

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\left\{ M \;\middle|\; \begin{array}{l} M \text{ tries to write on the left of the cell when it} \\ \text{is at the leftmost bit of the input} \end{array} \right\}.$$

Not applicable, but the language is in fact undecidable.

Applications of Rice's theorem

We now learn how to apply Rice's theorem

$$\left\{ M \,\middle|\, \begin{array}{l} M \text{ tries to write on the left of the cell when it} \\ \text{is at the leftmost bit of the input} \end{array} \right\}.$$

Not applicable, but the language is in fact undecidable.

Rice's theorem cannot be used to prove the undecidability of this language!

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that
$\mathcal{L}_P = \{M \mid L(M) \in P\}$.

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Assume that $\mathcal{L}_P$ is decidable.

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Assume that $\mathcal{L}_P$ is decidable.

Using this assumption prove that A_{TM} is decidable.

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Assume that $\mathcal{L}_P$ is decidable.

Using this assumption prove that A_{TM} is decidable.

More specifically:

$$(M, w) \qquad \longrightarrow \quad N$$

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Assume that $\mathcal{L}_P$ is decidable.

Using this assumption prove that A_{TM} is decidable.

More specifically:

$$
\begin{array}{ccc}
(M, w) & \longrightarrow & N \\
\text{if } w \in L(M) & \longrightarrow & \langle N \rangle \in \mathcal{L}_P
\end{array}
$$

Proof of Rice's theorem

Theorem

Let P be a property such that it is not trivial. Recall that $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof Idea:

Let P be a non-trivial property.

Assume that $\mathcal{L}_P$ is decidable.

Using this assumption prove that A_{TM} is decidable.

More specifically:

$$(M, w) \longrightarrow N$$

$$\begin{array}{lcl} \text{if } w \in L(M) & \longrightarrow & \langle N \rangle \in \mathcal{L}_P \\ \text{if } w \notin L(M) & \longrightarrow & \langle N \rangle \notin \mathcal{L}_P \end{array}$$

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$.

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \emptyset$.

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \varnothing$.

we assume that $\varnothing$ does not have property P

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \emptyset$.

we assume that $\emptyset$ does not have property P[1]

[1] We will remove this assumption later.

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \varnothing$.

we assume that $\varnothing$ does not have property P

on input x
{
if M accepts w
then if M_1 accepts x
then accept
}

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \varnothing$.

we assume that $\varnothing$ does not have property P

on input x
{
 if M accepts w
 then if M_1 accepts x
  then accept
}

Claim: $w \in L(M)$ if and only if $\langle N \rangle \in \mathcal{L}_P$

Proof of Rice's theorem

Theorem

Let P be a non-trivial property of Turing recognizable languages. Let $\mathcal{L}_P = \{M \mid L(M) \in P\}$. Then $\mathcal{L}_P$ is undecidable.

Design of N

Let M_1 be the TM s.t. $L(M_1)$ has Property P.

Let $L(M_2)$ be the TM s.t. $L(M_2) = \varnothing$.

we assume that $\varnothing$ does not have property P

on input x
{
if M accepts w
then if M_1 accepts x
then accept
}

Claim: $w \in L(M)$ if and only if $\langle N \rangle \in \mathcal{L}_P$