

程序代写代做 CS编程辅导

FIT2014 Theory of Computation



Lecture 22

WeChat: cstutorcs

Assignment Project Exam Help

slides by Graham Farr
Email: tutorcs@163.com

QQ: 749389476

COMMONWEALTH OF AUSTRALIA

Copyright Regulations 1969

<https://tutorcs.com>

This material has been reproduced and communicated to you by or on behalf of Monash University
in accordance with s113P of the Copyright Act 1968 (the Act).

The material in this communication may be subject to copyright under the Act.

Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

Do not remove this notice.

Overview

程序代写代做 CS编程辅导



- ▶ Halting Problem (or Entscheidungsproblem)
- ▶ Proof of its undecidability
- ▶ Using mapping reductions to prove undecidability
- ▶ Other undecidable problems

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Undecidable languages exist

程序代写代做 CS编程辅导

The set of all deciders is countable.

- ▶ $\{\text{CWL-encodings of deciders}\} \subseteq \Sigma^*$
- ▶ and Σ^* is countable. (Lecture 5)

WeChat: cstutorcs

The set of all decidable languages is countable.

Assignment Project Exam Help

The set of *all* languages is uncountable. (Lecture 5)

Email: tutorcs@163.com

Therefore undecidable languages exist.

QQ: 749389476

<https://tutorcs.com>



Halting Problem: Definition

Halting Problem

程序代写代做 CS编程辅导

INPUT: Turing machine P , input x

QUESTION: If P is run with  does it eventually halt?

As a language:

HaltingProblem $:= \{ \langle P, x \rangle : \text{when } P \text{ is run with input } x, \text{ it eventually halts.} \}$

WeChat: cstutorcs

Assignment Project Exam Help

Theorem.

The Halting Problem is undecidable.

Email: tutorcs@163.com

QQ: 749389476

Proved by:

<https://tutorcs.com>

- ▶ Alonzo Church (1936): lambda calculus
- ▶ Alan Turing (1936-37): Turing machines

Halting Problem

程序代写代做 CS编程辅导



Theorem.

The Halting Problem is undecidable.

Proof ingredients:

- ▶ contradiction
- ▶ diagonalisation
- ▶ a version of the Liar Paradox

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476 "This sentence is false."

<https://tutorcs.com>

Consider what happens when we run Turing machines (encoded as strings) on input strings.

✓ = Halts; ✗ = Doesn't halt.

Turing machines

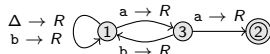
ϵ

a

b

aa

⋮



ababababbababbbbbbabaaabaaaab-
aaababababbbaabaabaaaab

⋮

程序代写代做 CS编程辅导 inputs to TMs

ϵ	a	b	aa	ab	ba	bb	aaa	aab	...
	✗	✗	✓	✗	✓	✓	✓	✗	...
	✗	✓	✓	✗	✗	✗	✓	✓	...
WeChat: cstutorcs	✓	✗	✗	✗	✓	✗	✗	✓	...
Assignment Project Exam Help	✓	✗	✓	✗	✗	✓	✓	✗	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	...
Email: tutorcs@163.com	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	...
QQ: 749389476	✗	✓	✓	✗	✓	✗	✓	✓	...
https://tutorcs.com	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	...

E:



...

Halting Problem is Undecidable

程序代写代做 CS编程辅导

Proof. (by contradiction)



Assume there is a Decider, D , for the Halting Problem.

WeChat: cstutorcs

So it can tell, for any P and x , whether or not P eventually halts after being given input x .

Assignment Project Exam Help

So it can tell, for any P , whether or not P eventually halts after being given input P !

Email: tutorcs@163.com

Construct another program (Turing machine) E as follows ...

QQ: 749389476

<https://tutorcs.com>

Halting Problem is Undecidable (cont'd)

E

程序代写代做 CS编程辅导

Input: P

Use D to determine whether P halts if P runs on itself.

If D says, " P halts, with output x " : loop forever.

If D says, " P loops forever" : input P : Halt.



WeChat: cstutorcs

What happens when E is given itself as input?

Assignment Project Exam Help

If E halts, for input E : then E loops forever, for input E .

Email: tutorcs@163.com

If E loops forever, for input E : then E halts, for input E .

QQ: 749389476

Contradiction!

<https://tutorcs.com>



YouTube film of proof: <https://www.youtube.com/watch?v=92WHN-pAFCs>

Other Undecidable Problems

DIAGONAL HALTING PROBLEM

程序代写代做 CS编程辅导

INPUT: Turing machine P

QUESTION: Does P eventually halt for input P ?



Above proof already shows this

WeChat: cstutorcs

HALT FOR INPUT ZERO

INPUT: Turing machine P

Assignment Project Exam Help

QUESTION: Does P eventually halt, for input 0?

Email: tutorcs@163.com

Theorem.

HALT FOR INPUT ZERO is undecidable

QQ: 749389476

<http://tutorcs.com>

We'll prove this by mapping reduction from the Diagonal Halting Problem.

Using mapping reductions

程序代写代做 CS编程辅导



Recall:

If there is a mapping reduction f from K to L , then:

If L is decidable, then K is decidable.

If K is undecidable, then L is undecidable.

Email: tutorcs@163.com

QQ: 749389476

<https://tutorcs.com>

Proof. ... that **HALT FOR INPUT ZERO** is undecidable:

Let M be any program, which we regard as an input to the Diagonal Halting Problem.
Define M' as follows:

M'

Input: x
Run M on input M .



WeChat: cstutorcs

Observe:

Assignment Project Exam Help

- ▶ The construction $M \mapsto M'$ is computable.
- ▶ M halts on input M if and only if M' halts on input 0.

Email: tutores@163.com

QQ: 749389476

So, the function that sends $M \mapsto M'$ is a mapping reduction from
DIAGONAL HALTING PROBLEM to **HALT FOR INPUT ZERO**.

https://tutorcs.com

Therefore **HALT FOR INPUT ZERO** is undecidable.



Other Undecidable Problems

程序代写代做 CS编程辅导

There's nothing special about z
So we get a whole lot of undecidable results.



For example:

WeChat: cstutorcs

HALT FOR INPUT 42

INPUT: Turing machine P

Assignment Project Exam Help

QUESTION: Does P eventually halt, for input 42?

Email: tutorcs@163.com

Proof of undecidability is virtually identical to the previous one ...

Use a mapping reduction, with 42 instead of 0.

QQ: 749389476

<https://tutorcs.com>

Other Undecidable Problems

程序代写代做 CS编程辅导



ALWAYS HALTS

INPUT: Turing machine P

QUESTION: Does P always halt eventually, for any input?

WeChat: cstutorcs

Theorem.

ALWAYS HALTS is undecidable.

Assignment Project Exam Help

Email: tutorcs@163.com

Proof is virtually identical to the previous one . . .

QQ: 749389476

<https://tutorcs.com>

Proof. ... that **ALWAYS HALTS** is undecidable:

Let M be any program, which we regard as an input to the Diagonal Halting Problem.
Define M' as follows:

M'

Input: x
Run M on input M .



WeChat: cstutorcs

Observe:

Assignment Project Exam Help

- ▶ The construction $M \mapsto M'$ is computable.
- ▶ M halts on input M if and only if M' always halts.

Email: tutores@163.com

QQ: 749389476

So, the function that sends $M \mapsto M'$ is a mapping reduction from
DIAGONAL HALTING PROBLEM to **ALWAYS HALTS**.

https://tutorcs.com

Therefore **ALWAYS HALTS** is undecidable.

Other Undecidable Problems

程序代写代做 CS编程辅导

SOMETIMES HALTS

INPUT: Turing machine P



QUESTION: Is there some input for which P eventually halts?

WeChat: cstutorcs

Theorem.

SOMETIMES HALTS is undecidable.

Assignment Project Exam Help

Email: tutorcs@163.com

Proof is virtually identical to the previous one . . .

QQ: 749389476

<https://tutorcs.com>

Proof. ... that **SOMETIMES HALTS** is undecidable:

Let M be any program, which we regard as an input to the Diagonal Halting Problem.
Define M' as follows:

M'

Input: x
Run M on input M .



WeChat: cstutorcs

Observe:

Assignment Project Exam Help

- ▶ The construction $M \mapsto M'$ is computable.
- ▶ M halts on input M if and only if M' halts for some input.

Email: tutores@163.com

QQ: 749389476

So, the function that sends $M \mapsto M'$ is a mapping reduction from
DIAGONAL HALTING PROBLEM to **SOMETIMES HALTS**.

https://tutorcs.com

Therefore **SOMETIMES HALTS** is undecidable.

Other Undecidable Problems

NEVER HALTS

INPUT: Turing machine P

QUESTION: Does P always loop, for any input?



程序代写代做 CS编程辅导

Theorem.

NEVER HALTS is undecidable. WeChat: cstutorcs

Proof. by a more general type of reduction, from SOMETIMES HALTS. Assignment Project Exam Help

If D is a decider for NEVER HALTS, then switching Accept and Reject gives a decider for SOMETIMES HALTS. Email: tutorcs@163.com
QQ: 749389476

But we now know that SOMETIMES HALTS is undecidable.

Contradiction.

So NEVER HALTS is undecidable too.



Other Undecidable Problems

程序代写代做 CS编程辅导



INPUT: Turing machine P and

QUESTION: Do P and Q always halt, or both loop?
i.e., is it the case that:

WeChat: cstutorcs

$\forall x : P \text{ halts on input } x \iff Q \text{ halts on input } x \quad \dots?$

Assignment Project Exam Help

Email: tutorcs@163.com

INPUT: Turing machine P

QUESTION: If P is run on the input "42", does it output "42"?

<https://tutorcs.com>

Decidable or Undecidable?

程序代写代做 CS编程辅导

INPUT: Turing machine P , input x

QUESTION: Does P accept x ?



INPUT: Turing machine P , input x , positive integer t

QUESTION: When P is run on x , does it halt in $\leq t$ steps?

Assignment Project Exam Help

INPUT: Turing machine P , positive integer s .

QUESTION: Does P have $\leq s$ states?

QQ: 749389476

INPUT: Turing machine P , positive integer k .

QUESTION: Does P halt for some input of length $\leq k$.

<https://tutorcs.com>

Other Undecidable Problems

程序代写代做 CS编程辅导

INPUT: a Turing machine P

QUESTION: Is $\text{Accept}(P)$ regular?
i.e., is P equivalent to a Finite Automaton?



INPUT: a CFG

QUESTION: is the language it generates regular?

WeChat: cstutorcs

Assignment Project Exam Help

INPUT: a CFG

QUESTION: is there any string that it doesn't generate? (over same alphabet)

Email: tutorcs@163.com

QQ: 749389476

INPUT: two CFGs.

QUESTION: Do they define the same language?

<https://tutorcs.com>

Other Undecidable Problems

程序代写代做 CS编程辅导

INPUT: a polynomial (in several variables)

QUESTION: Does it have an integer root?



(Y. Matiyasevich, 1970)

WeChat: cstutorcs

Assignment Project Exam Help

Post Correspondence Problem Email: tutorcs@163.com

(a problem about string matching;
see Sipser, Section 5.2)

QQ: 749389476

<https://tutorcs.com>



<https://mathshistory.st-andrews.ac.uk/Biographies/Matiyasevich/>

Yuri Matiyasevich (b. 1947)



<https://mathshistory.st-andrews.ac.uk/Biographies/Post/>

Emil Post (1897–1954)

Language classes

程序代写代做 CS编程辅导



ble

?

WeChat: cstutorcs

Assignment Project Exam Help

Email: tutorcs@163.com

QQ: 749389476 CFLs

<https://tutorcs.com>

regular
languages

Revision

程序代写代做 CS编程辅导



- ▶ Know and understand the problem
- ▶ Prove its undecidability
- ▶ Be able to use mapping reductions to prove undecidability
- ▶ Know examples of undecidable problems.

WeChat: cstutorcs

Assignment Project Exam Help

Reading: Sipser, pp. 201–209, 215–220, 234–236.

Email: tutorcs@163.com

Preparation: Sipser, pp. 170, 209–210.

QQ: 749389476

<https://tutorcs.com>