

Was ist Hardware?

- : r1 Computerhardware.
- : r2 Computer-Software
- : r3 Informationseinheit
- : r4 Der Teil des Computers, den ich nicht erreichen kann.

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Was ist Software?

- : r1 Computerhardware.
- : r2 Computer-Software
- : r3 Informationseinheit
- : r4 Der Teil des Computers, den ich berühren kann.

:r1 0

:r2 2 ok

:r3 0

:r4 0

--

Was ist ein Positionssystem?

- : r1 Der Wert der Zahl ergibt sich aus der Summe aller Ziffern
- : r2 Ein numerisches System, in dem nicht alle mathematischen Operationen ausgeführt werden können.
- : r3 Der Wert jeder Ziffer, angegeben durch ihre Position in der Symbolfolge
- : r4 Eine Methode zur Darstellung von Zahlen, bei der der Wert einer Ziffer nicht durch ihre Position

in der angegebenen Ziffernfolge angegeben wird

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Was ist ein nicht positionelles Zahlensystem?

- : r1 Der Wert der Zahl ergibt sich aus der Summe aller Ziffern
- : r2 Ein numerisches System, in dem nicht alle mathematischen Operationen ausgeführt werden

können.

: r3 Der Wert jeder Ziffer, angegeben durch ihre Position in der Symbolfolge

: r4 Eine Methode zur Darstellung von Zahlen, bei der der Wert einer Ziffer nicht durch ihre Position in der angegebenen Ziffernfolge angegeben wird

: r1 0

: r2 0

: r3 0

: r4 2 ok

Was ist Logik?

: r1 Die Wissenschaft des guten Denkens und die Kunst des Denkens

: r2 Vorher festgelegtes Verfahren zur Lösung des Problems

: r3 Informationen zuerst erhalten

: r4 Das Verfahren, mit dem wir aus einer bestimmten Beobachtung allgemeine Schlüsse ziehen

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Was ist Induktion?

: r1 Verallgemeinerung; der Prozess, allgemeine Schlussfolgerungen aus einer bestimmten Beobachtung zu ziehen

: r2 Die Wissenschaft des guten Denkens

: r3 Wir erstellen Hypothesen für beobachtete Phänomene, Diagnostik von "Fehlern"

: r4 Die Schlussfolgerung folgt logischerweise aus Annahmen, wenn unter keinen Umständen der Fall sein kann, dass die Annahmen wahr und die Schlussfolgerung falsch wären.

: r1 2 ok

: r2 0

: r3 0

: r4 0

--

Was ist Abzug?

: r1 Verallgemeinerung; der Prozess, allgemeine Schlussfolgerungen aus einer bestimmten Beobachtung zu ziehen

: r2 Die Wissenschaft des guten Denkens

: r3 Wir erstellen Hypothesen für beobachtete Phänomene, Diagnostik von "Fehlern"

: r4 Die Schlussfolgerung folgt logischerweise aus Annahmen, wenn unter keinen Umständen der Fall sein kann, dass die Annahmen wahr und die Schlussfolgerung falsch wären.

: r1 0

: r2 0

: r3 0

: r4 2 ok

-

Was ist Entführung?

: r1 Verallgemeinerung; der Prozess, allgemeine Schlussfolgerungen aus einer bestimmten Beobachtung zu ziehen

: r2 Die Wissenschaft des guten Denkens

: r3 Wir erstellen Hypothesen für beobachtete Phänomene, Diagnostik von "Fehlern"

: r4 Die Schlussfolgerung folgt logischerweise aus Annahmen, wenn unter keinen Umständen der Fall sein kann, dass die Annahmen wahr und die Schlussfolgerung falsch wären.

: r1 0

: r2 0

: r3 2 ok

: r4 0

--

Was ist ein Set?

: r1 Zusammenfassung der Informationen zum Objekt

: r2 Ein anderer Name für Sitzungen

: r3 Objekte, die mathematisch nicht beschrieben werden können

: r4 Eine Sammlung von Objekten, die klar identifiziert und unterscheidbar sind und einen Teil der Welt unserer Ideen und Gedanken bilden

: r1 0

: r2 0

: r3 0

: r4 2 ok

-

Was ist eine Sitzung?

: r1 Zusammenfassung der Informationen zum Objekt

: r2 n-Beziehung zwischen Mengen jede Untermenge von kartesischen Produkt-n-Mengen
: r3 Objekte, die mathematisch nicht beschrieben werden können
: r4 Eine Sammlung von Objekten, die klar identifiziert und unterscheidbar sind und einen Teil der Welt unserer Ideen und Gedanken bilden

: r1 0

: r2 2 ok

: r3 0

: r4 0

--

Was ist ein Potenzsatz?

: r1 Eine Menge, die kein Element enthält

: r2 Menge aller Teilmengen

: r3 Menge aller Elemente

: r4 Menge aller Sitzungen

: r1 0

: r2 2 ok

: r3 0

: r4 0

-

Was ist das kartesische Produkt $X \times Y$?

: r1 Ein diskretes Produkt, ein Satz, der ungeordnete Paare enthält, wobei eines oder beide Elemente sowohl X als auch Y sein können

: r2 Setze $X \times Y$, das alle geordneten Paare enthält, wobei das erste Element von X und das zweite Element von Y ist

: r3 Eine Menge solcher Paare, bei denen das Element von X gleich dem Element von Y ist

: r4 Menge aller Elemente von X und Y.

: r1 0

: r2 2 ok

-

Was ist ein orientiertes Diagramm?

: r1 Ein Paar (V, E) , wobei E eine Teilmenge des kartesischen Produkts $V \times V$ ist. E ist ein geordnetes Paar von Eckpunkten

: r2 Liste der Eckpunkte und Kanten, an denen die Kanten durcheinandergewürfelte Paare bilden

- : r3 Eine Menge, in der jedes nicht leere Teil das kleinste Element hat.
- : r4 Gezeichnete Prozedur zur Ausführung des angegebenen Algorithmus

:r1 2 ok

:r2 0

:r3 0

:r4 0

--

Was ist ein bestellter Satz?

- : r1 Eine Menge, bei der zwei Elemente nicht verglichen werden können
- : r2 Die Menge, auf der die Beziehung R definiert ist, die transitiv, schwach antisymmetrisch und trichotomisch ist.
- : r3 Ein anderer Name für eine leere Menge
- : r4 Menge aller geordneten Paare (x, y) , für die $x < y$ ist

: r1 0

: r2 2 ok

: r3 0

: r4 0

-

Was ist eine scharfe Anordnung?

- : r1 Eine Menge ist scharf geordnet, wenn alle ihre Elemente gleich sind
- : r2 Ein anderer Name für eine leere Menge
- : r3 Für jeweils zwei Elemente der Menge gilt $x < y$.
- : r4 Für jeweils zwei Elemente der Menge gilt $x \geq y$.

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Was ist ein verschwommenes Arrangement?

- : r1 Eine Menge, bei der zwei Elemente nicht verglichen werden können
- : r2 Ein anderer Name für eine leere Menge

: r3 Für jeweils zwei Elemente der Menge gilt $x < y$.

: r4 Für jeweils zwei Elemente der Menge gilt $x \neq y$.

:r1 0

:r2 0

:r3 0

:r4 2 ok

--

Was ist Injektion (injizierbare Bildgebung)?

: r1 $f: A \rightarrow B$, die ganze Menge B ist ein Wertebereich

: r2 klar zueinander

: r3 $f: A \rightarrow B$ und $g: B \rightarrow C$, $h: A \rightarrow C$, $h = g \circ f$,

: r4 $f: A \rightarrow B$, $\exists b \in B: \exists! a \in A: f(a) = b$

: r1 0

: r2 0

: r3 0

: r4 2 ok

-

Was ist Bijektion?

: r1 $f: A \rightarrow B$, die ganze Menge B ist ein Wertebereich

: r2 klar zueinander

: r3 $f: A \rightarrow B$ und $g: B \rightarrow C$, $h: A \rightarrow C$, $h = g \circ f$,

: r4 $f: A \rightarrow B$, $\exists b \in B: \exists! a \in A: f(a) = b$

: r1 0

: r2 2 ok

: r3 0

-

Was ist ein Widerspruch?

: r1 $f: A \rightarrow B$, die ganze Menge B ist ein Wertebereich

: r2 klar zueinander

: r3 $f: A \rightarrow B$ und $g: B \rightarrow C$, $h: A \rightarrow C$, $h = g \circ f$,

: r4 $f: A \rightarrow B$, $\exists b \in B: \exists! a \in A: f(a) = b$

:r1 2 ok

:r2 0

:r3 0

:r4 0

--

Was ist das Komponieren von Impressionen?

: r1 $f: A \rightarrow B$, die ganze Menge B ist ein Wertebereich

: r2 klar zueinander

: r3 $f: A \rightarrow B$ und $g: B \rightarrow C$, $h: A \rightarrow C$, $h = g \circ f$

: r4 $f: A \rightarrow B$, $\exists b \in B: \exists! a \in A: f(a) = b$

: r1 0

: r2 0

: r3 2 ok

: r4 0

Booleas Algebra - Wählen Sie die richtige Behauptung

: r1 $(A-B) = (\text{NICHT } A) - B$

: r2 $(AB) = (AB)$

: r3 $(\neg B) = (\neg \text{NOT } (B))$

: r4 $(A \neg B) = (\text{NICHT } A) \neg \text{NOT } (B)$

: r1 2 ok

: r2 0

: r3 0

: r4 0

--

Booleas Algebra - Wählen Sie die richtige Behauptung

: r1 $(A-B) = (\text{NICHT } A) - B$

: r2 $(AB) = (\text{NICHT } (A \text{ XOR } B))$

: r3 $(A \neg B) = (\text{NICHT } A) \neg \text{NOT } (B)$

: r4 $(A-B) = (\text{NICHT } A) - B$

:r1 0

:r2 2 ok

:r3 0

:r4 0

--

Booleaova algebra – vyber správné tvrzení

:r1 $(A \text{ XOR } B) = (\text{NOT}(A) \text{ AND } \text{NOT}(B))$

:r2 $(A \text{ XOR } B) = (A \text{ AND } \text{NOT}(B))$

:r3 $(A \text{ XOR } B) = ((B \text{ AND } \text{NOT}(A)) \text{ AND } (A \text{ AND } \text{NOT}(B)))$

:r4 $(A \text{ XOR } B) = (\text{NOT}(A) \text{ AND } B)$

:r1 0

:r2 0

:r3 2 ok

:r4 0

--

Was ist Determinismus?

: r1 Eine Reihe von Konzepten, Regeln und Verfahren, mit deren Hilfe wir numerische Informationen organisieren können

: r2 Determinismus beschreibt das Verhalten einer Menge

: r3 Es beschreibt eine Situation, in der es nur eine mögliche Zukunft gibt, und dies ist theoretisch durchaus vorhersehbar.

: r4 Beschreibt eine Situation, in der es mehr mögliche Zukünfte gibt, die zufällig ausgewählt werden.

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Was bedeutet Stochastizität?

: r1 Eine Reihe von Konzepten, Regeln und Verfahren, mit deren Hilfe wir numerische Informationen organisieren können

: r2 Stochastizität beschreibt das Verhalten von Sitzungen

: r3 Es beschreibt eine Situation, in der es nur eine mögliche Zukunft gibt, und dies ist theoretisch durchaus vorhersehbar.

: r4 Beschreibt eine Situation, in der es mehr mögliche Zukünfte gibt, die zufällig ausgewählt werden.

: r1 0

: r2 0

: r3 0

: r4 2 ok

-

Was ist Modus?

: r1 Durchschnittswert

: r2 Häufigster Wert

: r3 Wert genau in der Mitte der Verteilung

: r4 Die Summe der kleinsten und größten Werte geteilt durch zwei

:r1 0

:r2 2 ok

:r3 0

:r4 0

--

Was ist der Median?

: r1 Durchschnittswert

: r2 Häufigster Wert

: r3 Wert genau in der Mitte der Verteilung

: r4 Die Summe der kleinsten und größten Werte geteilt durch zwei

: r1 0

: r2 0

: r3 2 ok

: r4 0

--

Rauschkanal beschreibt

- : r1 Übertragung von Informationen über einen verrauschten Kanal
- : r2 Informationen mit Rauschen übertragen
- : r3 Übertragung von Informationen über den Rauschkanal
- : r4 Wie der Kanal Rauschen überträgt, ohne eine Nachricht zu senden

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Syntax

- : r1 Beschreibt den qualitativen Aspekt der Information
- : r2 Gibt die Wichtigkeit der Nachricht an
- : r3 Ist die Bedeutung von Informationen
- : r4 Bezieht sich auf die Anordnung von Zeichen als Informationsträger

:r1 0

:r2 0

:r3 0

:r4 2 ok

--

Semantik

- : r1 Beschreibt den quantitativen Aspekt der Information
- : r2 Gibt die Wichtigkeit der Nachricht an
- : r3 Ist die Bedeutung von Informationen
- : r4 Bezieht sich auf die Anordnung von Zeichen als Informationsträger

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Pragmatischer Inhalt

- : r1 Beschreibt den qualitativen Aspekt der Information
- : r2 Beschreibt den quantitativen Aspekt der Information
- : r3 Ist die Bedeutung von Informationen
- : r4 Bezieht sich auf die Anordnung von Zeichen als Informationsträger

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Paritätssicherheit

- : r1 Gibt an, dass ein neuer Datenbereich erstellt wird, der die Anzahl der Nullen und Einsen angibt
- : r2 Gibt an, dass an jedes Segment ein zusätzliches Bit angehängt wird, um die Anzahl der binären Einsen zu einer geraden (ungerade Parität) oder ungeraden (ungerade Parität) Zahl hinzuzufügen
- : r3 Die Daten werden in Abschnitte der gewünschten Länge (8, 16, 32 Bit) unterteilt und diese Abschnitte werden ohne Übertragung bitweise addiert. Das resultierende Datensegment wird mit den übertragenen Daten verbunden.
- : r4 Das neue Daten-Slice wird an die Datennachricht angehängt oder mit dieser logisch verknüpft, sodass die Identität der Person in Bezug auf die Datennachricht überprüft werden kann.

:r1 0

:r2 2 ok

:r3 0

:r4 0

--

CRC-Sicherheit

- : r1 Gibt an, dass ein neues Datensegment erstellt wird, das die Anzahl der Nullen und Einsen in den Daten angibt
- : r2 Gibt an, dass an jedes Segment ein zusätzliches Bit angehängt wird, um die Anzahl der binären Einsen zu einer geraden (ungerade Parität) oder ungeraden (ungerade Parität) Zahl hinzuzufügen
- : r3 Die Daten werden in Abschnitte der gewünschten Länge (8, 16, 32 Bit) unterteilt und diese Abschnitte werden ohne Übertragung bitweise addiert. Das resultierende Datensegment wird mit

den übertragenen Daten verbunden.

: r4 Das neue Daten-Slice wird an die Datennachricht angehängt oder mit dieser logisch verknüpft, sodass die Identität der Person in Bezug auf die Datennachricht überprüft werden kann.

: r1 0

: r2 0

: r3 2 ok

: r4 0

-

Hamming-Code

: In der Telekommunikation verwendeter linearer Code r1 zum Erkennen von bis zu zwei fehlerhaften Bits oder zum Korrigieren eines fehlerhaften Bits

: r2 Gibt an, dass ein neues Datensegment erstellt wird, das die Anzahl der Nullen und Einsen in den Daten angibt

: r3 Ist eine Möglichkeit, den Datenübertragungsalgorithmus auszudrücken

: r4 Die Daten werden in Abschnitte der gewünschten Länge (8, 16, 32 Bit) unterteilt und diese Abschnitte werden ohne Übertragung bitweise addiert. Das resultierende Datensegment wird mit den übertragenen Daten verbunden

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Elektronische Unterschrift

: r1 Gibt an, dass ein neues Datensegment erstellt wird, das die Anzahl der Nullen und Einsen in den Daten angibt

: r2 Gibt an, dass an jedes Segment ein zusätzliches Bit angehängt wird, um die Anzahl der binären Einsen zu einer geraden (ungerade Parität) oder ungeraden (ungerade Parität) Zahl hinzuzufügen

: r3 Die Daten werden in Abschnitte der gewünschten Länge (8, 16, 32 Bit) unterteilt und diese Abschnitte werden ohne Übertragung bitweise addiert. Das resultierende Datensegment wird mit den übertragenen Daten verbunden.

: r4 Wird an eine Datennachricht angehängt oder ist mit dieser logisch verknüpft. Dadurch kann die Identität des Unterzeichners in Bezug auf die Datennachricht überprüft werden.

:r1 0

:r2 0

:r3 0

:r4 2 ok

--

Komplexitätstheorie

: r1 Beschreibt, wie schwierig es ist, einen bestimmten Algorithmus zu implementieren

: r2 Es konzentriert sich auf die Klassifizierung von Rechenproblemen anhand ihrer eigenen Komplexität und Klassenbeziehungen.

: r3 Er sagt, wenn das Problem komplex ist, kann es nicht algorithmisch gelöst werden

: r4 Beschreibt die Beschreibung des Algorithmus und dessen Vereinfachung

: r1 0

: r2 2 ok

: r3 0

: r4 0

-

Analyse von Algorithmen

: r1 Es behandelt die Menge an Ressourcen, die von einem bestimmten Algorithmus benötigt werden

: r2 Sagt, wenn das Problem komplex ist, kann es nicht algorithmisch gelöst werden

: r3 Beschreibt die Beschreibung des Algorithmus und dessen Vereinfachung

: r4 Der Schwerpunkt liegt auf der Klassifizierung von Rechenproblemen nach ihrer eigenen Komplexität und der Bestimmung von Klassenbeziehungen.

: r1 2 ok

: r2 0

: r3 0

: r4 0

--

Entscheidungsproblem

- : r1 Es gibt ein Problem, bei dem wir uns zwischen mindestens drei Ergebnissen entscheiden müssen
- : r2 Es gibt ein Problem, bei dem das Programm immer stoppt.
- : r3 Grundtyp der Komplexitätstheorie Probleme, es hat nur zwei Ausgänge Ja / Nein
- : r4 beschreibt eine Situation, in der jedes Problem maximal in linearer Zeit (n-fache Eingabegröße) entschieden werden kann.

:r1 0

:r2 0

:r3 2 ok

:r4 0

--

NP-vollständige Probleme

r1 sind nichtdeterministische Polynomprobleme, auf die alle anderen NP-Probleme polynomisch reduzierbar sind

- : r2 ein Problem, das in Turing vollständig beschrieben ist
- : r3 sind Probleme, deren Lösung durch eine deterministische Turingmaschine in linearer Zeit gefunden werden kann.
- : r4 Sie können niemals gelöst werden - völlig unlösbare Probleme

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Sprache

- : r1 Eine Menge von Wörtern, die nicht durch einen regulären Ausdruck beschrieben werden können
- : r2 Eine Struktur, die die Grammatik beschreibt
- : r3 Eine Gruppe von Wörtern, die von keinem Automaten akzeptiert werden
- : r4 Wird durch eine nicht leere Menge V definiert, die ein Alphabet ist. Ihre Elemente sind Zeichen oder Symbole

: r1 0

: r2 0

: r3 0

: r4 2 ok

-

Konečný automat

: r1 Beschreibt einen sehr einfachen Computer, der sich in einem von mehreren Zuständen befinden kann und basierend auf den Symbolen, die er von der Eingabe liest, zwischen diesen umschaltet

: r2 Es ist ein Automat, dessen Satz von Zuständen nicht endlich ist. Der endliche Automat hat ein endliches Gedächtnis, das mindestens zwei Schritte zurückliegt (da ein binäres System verwendet wird).

: r3 Eine Maschine, die nur zwei mögliche Zustände hat und deren Lauf immer nach dem ersten Schritt endet

: r4 Eine Maschine, deren Operation niemals endet, sondern auf einer endlichen Menge von Zuständen mit einem endlichen Speicher stattfindet, der der Anzahl von Zuständen pro Sekunde entspricht

:r1 2 ok

:r2 0

:r3 0

:r4 0

--

Deterministischer endlicher Automat

: r1 Es hat genau einen Zielzustand an jedem Punkt in der Übergangstabelle

: r2 Ist ein Automat, dessen Satz von Zuständen nicht endlich ist, hat der endliche Automat ein Gedächtnis von mindestens zwei Schritten zurück (wegen der Verwendung des Binärsystems)

: r3 Es gibt nicht nur einen Zielzustand in jedem Punkt der Tabelle, sondern eine ganze Reihe von Zuständen

: r4 Eine endliche Zustandsmaschine, deren Übergangstabelle jeweils aus einer Zeile besteht

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Nichtdeterministischer endlicher Automat

: r1 Ist ein Automat, dessen Satz von Zuständen nicht endlich ist, der endliche Automat hat ein Gedächtnis von mindestens zwei Schritten zurück (wegen der Verwendung eines binären Systems)

: r2 Es gibt nicht nur einen Zielzustand in jedem Punkt der Tabelle, sondern eine ganze Reihe von Zuständen

: r3 Eine endliche Zustandsmaschine, deren Übergangstabelle jeweils nur aus einer Zeile besteht

: r4 Es hat genau einen Zielzustand an jedem Punkt in der Übergangstabelle

: r1 0

: r2 2 ok

: r3 0

: r4 0

-

Turingmaschine

: r1 Eine Zustandsmaschine, die ein rechtes Endlosband enthält, auf das Zustände geschrieben werden, und ein linkes Endlosband, von dem Anweisungen gelesen werden

: r2 Es besteht aus einer Verarbeitungseinheit, die aus einer endlichen Zustandsmaschine, einem Programm in Form von Regeln der Übergangsfunktion und einem rechten Endlosband zum Schreiben von Zwischenergebnissen besteht

: r3 Ist der einzige Maschinentyp (Algorithmus), der NP-vollständige Probleme in linearer Zeit lösen kann.

: r4 Ist der einzige Maschinentyp (Algorithmus), der NP-vollständige Probleme in konstanter Zeit lösen kann.

:r1 0

:r2 2 ok

:r3 0

:r4 0

--

N-Band Turingmaschine

- : r1 Es ist eine Modifikation der Turing-Grundmaschine, die "multiple choice" erlaubt, dh in der Übergangstabelle gibt es an jeder Stelle n-Statustabellen
- : r2 Es handelt sich um eine Modifikation der Turing-Grundmaschine, bei der mehrere Bänder gleichzeitig gelesen und beschrieben werden
- : r3 Es ist eine Modifikation der Turing-Grundmaschine, die jede Turing-Maschine simulieren kann
- : r4 Es ist eine Modifikation der Turing-Grundmaschine, bei der die Maschine Zustände auf das n-Band schreibt und das Lesen immer nur von einer Stelle aus erfolgt

-

Universal-Turingmaschine

- : r1 Es ist eine Modifikation der Turing-Grundmaschine, die "multiple choice" erlaubt, dh in der Übergangstabelle gibt es an jeder Stelle n-Statustabellen
- : r2 Es handelt sich um eine Modifikation der Turing-Grundmaschine, bei der mehrere Bänder gleichzeitig gelesen und beschrieben werden
- : r3 Es ist eine Modifikation der Turing-Grundmaschine, die jede Turing-Maschine simulieren kann
- : r4 Es ist eine Modifikation der Turing-Grundmaschine, bei der die Maschine Zustände auf das n-Band schreibt und das Lesen immer nur von einer Stelle aus erfolgt

-

Vollständigkeit prüfen

- : r1 Turing Kompletmaschine löst NP-Kompletprobleme in konstanter Zeit
- : r2 Turing Maschine komplette Maschine kann alle Probleme in der Endzeit lösen, einschließlich des Problems des Anhaltens
- : r3 Die Turing-Sprachen sind Programmiersprachen und Computer mit der gleichen Rechenleistung wie die Turing-Maschine
- : r4 Turing machine complete löst NP-complete-Probleme in linearer Zeit

: r1 0

Definitions of N-páskový Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 2 ok :r3 0 :r4 0 -- Univerzální Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 0 :r3 2 ok :r4 0 -- Turingovská úplnost :r1 Turingovsky úplný stroj řeší NP-úplné problémy v konstantním čase :r2 Turingovsky úplný stroj dokáže řešit libovolné problémy v konečném čase, včetně problému zastavení :r3 Turingovsky úplné jsou právě ty programovací jazyky a počítače, které mají stejnou výpočetní sílu jako Turingův stroj :r4 Turingovsky úplný stroj řeší NP-úplné problémy v lineárním čase :r1 0

Examples of N-páskový Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 2 ok :r3 0 :r4 0 -- Univerzální Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 0 :r3 2 ok :r4 0 -- Turingovská úplnost :r1 Turingovsky úplný stroj řeší NP-úplné problémy v konstantním čase :r2 Turingovsky úplný stroj dokáže řešit libovolné problémy v konečném čase, včetně problému zastavení :r3 Turingovsky úplné jsou právě ty programovací jazyky a počítače, které mají stejnou výpočetní sílu jako Turingův stroj :r4 Turingovsky úplný stroj řeší NP-úplné problémy v lineárním čase :r1 0

Synonyms of N-páskový Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 2 ok

:r3 0 :r4 0 -- Univerzální Turingův stroj :r1 Je modifikací základního Turingova stroje, který umožňuje „výběr z více možností“ tj. v přechodové tabulce má v každém místě tabulky n-stavů :r2 Je modifikací základního Turingova stroje, čte a zapisuje do více pásek najednou :r3 Je modifikací základního Turingova stroje, který dokáže simulovat libovolný Turingův stroj :r4 Je modifikací základního Turingova stroje, kdy stroj zapisuje stavy na n-pásek a čtení probíhá vždy pouze z jedné :r1 0 :r2 0 :r3 2 ok :r4 0 -- Turingovská úplnost :r1 Turingovsky úplný stroj řeší NP-úplné problémy v konstantním čase :r2 Turingovsky úplný stroj dokáže řešit libovolné problémy v konečném čase, včetně problému zastavení :r3 Turingovsky úplné jsou právě ty programovací jazyky a počítače, které mají stejnou výpočetní sílu jako Turingův stroj :r4 Turingovsky úplný stroj řeší NP-úplné problémy v lineárním čase :r1 0

See also

:r2 0

:r3 2 ok

:r4 0

--

Berechenbarkeitstheorie

: r1 Die Grundfragen der Berechenbarkeitstheorie lauten: Was ist alles und was ist nicht rechnerisch berechenbar? Zu welchen Problemen gibt es Algorithmen, die sie lösen?

: r2 Es wird versucht zu quantifizieren, wie viele Algorithmusschritte benötigt werden, um ein bestimmtes Problem zu lösen.

: r3 Die Grundfrage der Lösbarkeitstheorie lautet: Welcher Minimalalgorithmus löst das Problem?

: r4 Gibt an, wie viele Ressourcen benötigt werden, um ein bestimmtes Problem mit einer Turing-Maschine zu simulieren.

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Algorithmische Lösbarkeit

: r1 Wir sagen, dass ein Problem algorithmisch lösbar ist, wenn es einen Algorithmus gibt, der in der Lage ist, jede Instanz des Problems als Eingabe zu akzeptieren, und dessen Berechnung für eine solche Eingabe immer endet, wobei das Ergebnis das gewünschte Ergebnis ist.

: r2 Wir werden über ein gegebenes Ja / Nein-Problem sagen, dass es algorithmisch lösbar ist, wenn es einen Algorithmus gibt, der in der Lage ist, eine Instanz des gegebenen Problems als Eingabe zu akzeptieren, und dessen Berechnung für jede solche Eingabe immer endet, wobei die Ausgabe die gewünschte Antwort Ja ist. Wenn die richtige Antwort Nein lautet, endet der Algorithmus nicht.

: r3 Das Problem ist algorithmisch lösbar, wenn es möglich ist, einen solchen Algorithmus (Turing-Maschine) zu konstruieren, der das gewünschte Ergebnis in einer Zeit liefert, die linear von der Größe der Eingabe abhängt.

Das Problem soll algorithmisch lösbar sein, wenn es möglich ist, einen solchen deterministischen endlichen Automaten zu konstruieren, der unter allen Umständen das richtige Ergebnis liefert, unabhängig von der Größe und Art der Eingabe, zu einem Zeitpunkt, der maximal linear von der Größe der Eingabe abhängt.

: r1 2 ok

: r2 0

: r3 0

: r4 0

-

Algorithmische Entscheidbarkeit

: r1 Ein Problem ist algorithmisch entscheidbar, wenn es möglich ist, einen Algorithmus (Turing-Maschine) zu konstruieren, der das gewünschte Ergebnis in einer Zeit liefert, die linear von der Größe der Eingabe abhängt.

: r2 Wir sagen, dass ein Problem algorithmisch entscheidbar ist, wenn es einen Algorithmus gibt, der in der Lage ist, jede Instanz des Problems als Eingabe zu akzeptieren, und dessen Berechnung für eine solche Eingabe immer endet, wobei das Ergebnis das gewünschte Ergebnis ist.

: r3 Wir sagen über ein gegebenes Ja / Nein-Problem, das algorithmisch entscheidbar ist, wenn es einen Algorithmus gibt, der in der Lage ist, jede Instanz des Problems als Eingabe zu akzeptieren, und dessen Berechnung für eine solche Eingabe immer endet, wobei die Ausgabe die gewünschte Ja / Nein-Antwort ist.

: r4 Das Problem ist entscheidend, wenn wir den Algorithmus kennen, der stoppt und nur dann ein Ergebnis liefert, wenn die Antwort auf die Frage NEIN lautet. Wenn die Antwort JA lautet, werden wir es nie erfahren, da der Algorithmus niemals anhält

:r1 0

:r2 0

:r3 2 ok

:r4 0

--

Teilentscheidbarkeit

: r1 Das Problem ist teilweise entscheidbar, wenn wir den Algorithmus kennen, der stoppt und nur dann ein Ergebnis liefert, wenn die Antwort auf die Frage NEIN lautet. Wenn die Antwort JA lautet, werden wir es nie erfahren, da der Algorithmus niemals anhält.

: r2 Wir sagen, dass ein Problem algorithmisch teilweise lösbar ist, wenn es einen Algorithmus gibt, der in der Lage ist, jede Instanz des Problems als Eingabe zu akzeptieren, und dessen Berechnung für eine solche Eingabe immer endet, wobei das Ergebnis das gewünschte Ergebnis ist.

: r3 Wir sagen über ein gegebenes Ja / Nein-Problem, dass es algorithmisch teilweise entscheidbar ist, ob es einen Algorithmus gibt, der eine Instanz des Problems als Eingabe akzeptiert, und dessen Berechnung für eine solche Eingabe immer endet, wobei die Ausgabe die gewünschte Ja / Nein-Antwort ist.

: r4 Das Problem ist teilweise entscheidbar, wenn wir den Algorithmus kennen, der stoppt und nur dann ein Ergebnis liefert, wenn die Antwort auf die Frage Ja lautet. Wenn die Antwort Nein lautet, werden wir es nie erfahren, da der Algorithmus niemals anhält.

:r1 0

:r2 0

:r3 0

:r4 2 ok