There is a specific problem that is algorithmically unsolvable. Computers appear to be so powerful that you may believe that all problems will eventually yield to them. However, you will be disappointed. The general problem of software verification is not solvable by computer.

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, diagramma, design

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, documento

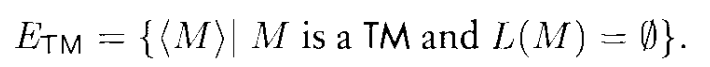
Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere

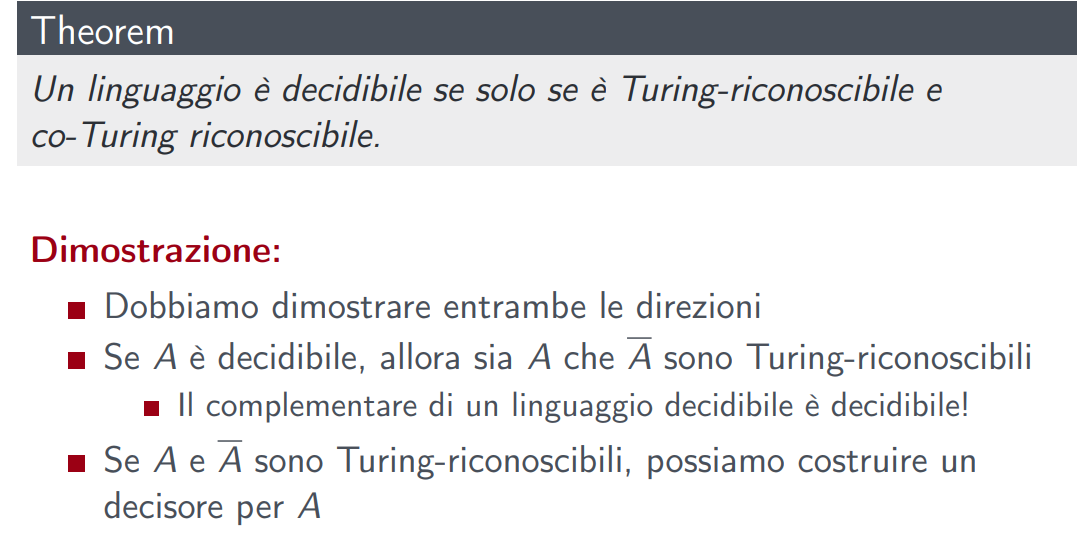
Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamente

Immagine che contiene testo, diagramma, Carattere, linea

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, linea, schermata

Descrizione generata automaticamenteImmagine che contiene testo, diagramma, linea, schermata

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata

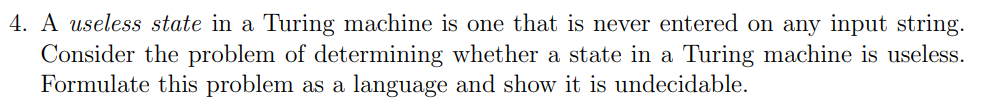
Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata

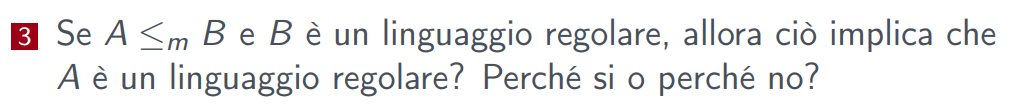
Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, algebra

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, algebra

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, linea, schermata

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, numero

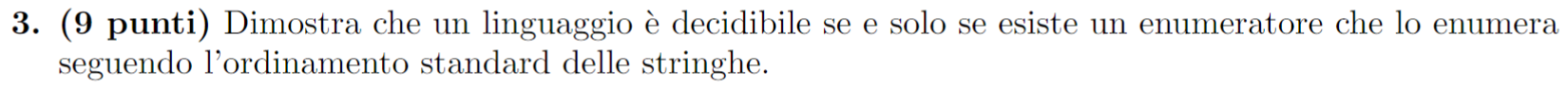
Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, carta

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente



Un linguaggio L è decidibile se e solo se esiste una macchina di Turing che accetta L e si ferma per ogni input. Questo significa che per ogni stringa nel linguaggio, la macchina di Turing decide se la stringa appartiene o meno al linguaggio in un numero finito di passaggi.

Un enumeratore per un linguaggio L è una macchina di Turing che elenca tutte le stringhe in L in ordine lessicografico. Questo significa che l’enumeratore genera tutte le stringhe in L una dopo l’altra, in ordine crescente secondo l’ordinamento standard delle stringhe.

Ora, dimostriamo che un linguaggio L è decidibile se e solo se esiste un enumeratore che lo enumera seguendo l’ordinamento standard delle stringhe.

1. **(⇒) Se L è decidibile, allora esiste un enumeratore che enumera L in ordine standard:**

Se L è decidibile, allora esiste una macchina di Turing M che decide L. Possiamo costruire un enumeratore E per L come segue: E genera tutte le stringhe in ordine lessicografico e per ogni stringa generata, esegue M su di essa. Se M accetta la stringa, allora E la stampa. In questo modo, E enumera tutte le stringhe in L in ordine lessicografico.

1. **(⇐) Se esiste un enumeratore che enumera L in ordine standard, allora L è decidibile:**

Se esiste un enumeratore E che enumera L in ordine lessicografico, allora possiamo costruire una macchina di Turing M che decide L come segue: dato un input x, M esegue E e confronta ogni stringa stampata da E con x. Se E stampa una stringa che è uguale a x, allora M accetta x. Se E stampa una stringa che è maggiore di x nell’ordinamento lessicografico, allora M rifiuta x. In questo modo, M decide se una data stringa appartiene o meno a L.

Quindi, un linguaggio L è decidibile se e solo se esiste un enumeratore che enumera L seguendo l’ordinamento standard delle stringhe. Questa è una dimostrazione ad alto livello e l’implementazione effettiva richiederebbe una gestione accurata delle transizioni di stato e dei movimenti del nastro. Ma dal punto di vista teorico, questa dimostrazione mostra l’equivalenza tra la decidibilità di un linguaggio e l’esistenza di un enumeratore per il linguaggio.