Immagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamente

Parola palindroma: 🡪 Siamo partiti da parola NON palindroma

Qui otterremmo una parola palindroma rispettando le condizioni del linguaggio 🡪 L non regolare

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamente

Esempio di parola “falla” 🡪

Immagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamenteN. di 0 maggiore rispetto al n. di 1 🡪 L non regolare.

Costruiremo una grammatica G' che genera dehash(L) basandosi sulla grammatica originale G che genera L.

Dato:

* L è un linguaggio privo di contesto
* dehash(L) = {dehash(w) | w ∈ L}
* dehash(w) è la stringa ottenuta rimuovendo tutti i simboli '#' da w

Prova:

Poiché L è libera dal contesto, esiste una grammatica libera dal contesto G in Forma Normale di Chomsky che genera L.

Costruiamo una nuova grammatica G' che genera dehash(L) come segue:

a) G' contiene tutte le variabili di G.

b) Il simbolo di inizio di G' è uguale al simbolo di inizio di G.

c) Per ogni regola di G, creiamo le regole corrispondenti in G':

Se la regola è della forma A → BC, aggiungiamo A → BC a G'.

Se la regola è della forma A → a (dove a è un terminale), aggiungiamo:

A → a se a ≠ '#'

A → ε (epsilon) se a = '#'

Aggiungiamo anche A → A per ogni variabile A per permettere di saltare '#' nelle derivazioni più lunghe.

Prova che G' genera dehash(L):

* Le regole copiate da G permettono a G' di generare la struttura delle parole in L.
* La modifica delle regole terminali rimuove tutti i simboli '#'.
* Le regole A → A permettono a G' di “saltare” un numero qualsiasi di simboli '#' nella derivazione originale.

Correttezza:

* Qualsiasi parola generata da G' corrisponde a una parola in L con tutti i simboli '#' rimossi.
* Qualsiasi parola in dehash(L) può essere generata da G' seguendo una derivazione simile a quella di G, ma omettendo i passaggi che produrrebbero '#'.

Conclusione:

* Poiché abbiamo costruito una grammatica libera da contesto G' che genera dehash(L), abbiamo dimostrato che dehash(L) è libera da contesto quando L è libera da contesto.
* Immagine che contiene testo, schermata, Carattere, algebra

  Descrizione generata automaticamenteQuesta costruzione garantisce che G' generi esattamente il linguaggio dehash(L), preservando la natura context-free del linguaggio originale L e rimuovendo tutti i simboli '#'.

Usiamo come sottoinput di 🡪 Se A è indecidibile allora B è indecidibile

Vale anche l’opposto! Quindi, se B (sottoinput) è decidibile, A è decidibile (superinput)

Immagine che contiene testo, Carattere, schermata, algebra

Descrizione generata automaticamenteLa funzione di riduzione prende sempre in input (TM) e (stringa di input). Esempio classico:

= su input , dove è una TM, e è una stringa:

* Costruiamo (risolve )
  + = su input
    - Copia l’input su tutto il nastro
    - Se trova la cella
    - Esegui (che permette di risolvere usando = sottoinput)
    - Se accetta, allora
    - Se rifiuta, rifiuta
  + Ritorna

Se appartiene a 2023, la TM si ferma accettando la stringa qualora la cella 2023 compaia, così accetta.

Se non appartiene, non abbiamo trovato la cella 2023 e la macchina rifiuta