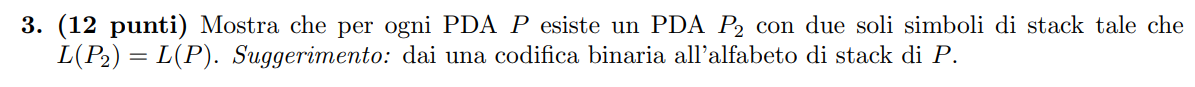
Immagine che contiene testo, Carattere, schermata, algebra

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

Descrizione generata automaticamente



Per costruire un PDA P2 con due soli simboli di stack equivalente a P, usiamo una codifica binaria dei simboli di stack di P.

Sia Γ = {γ1, ..., γn} l'alfabeto di stack di P. Codifichiamo ogni γi come una stringa binaria bi di lunghezza ⌈log2 n⌉.

P2 simula P usando due simboli di stack 0 e 1:

* Quando P fa push di γi, P2 fa push della codifica binaria bi
* Quando P fa pop di γi, P2 fa pop di ⌈log2 n⌉ simboli e verifica che corrispondano a bi
* P2 mantiene gli stessi stati di P

P2 accetta se e solo se P accetta, quindi L(P2) = L(P).

Questa costruzione mostra che è possibile simulare qualsiasi PDA usando solo due simboli di stack, mantenendo lo stesso linguaggio riconosciuto.

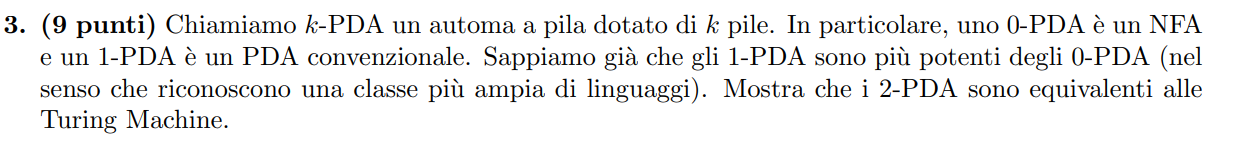
Immagine che contiene testo, Carattere, bianco

Descrizione generata automaticamente

Mostreremo che esiste un linguaggio riconoscibile da un 2-PDA ma non da un 1-PDA.

Consideriamo il linguaggio L = {a^n b^n c^n | n ≥ 0}.

1. L è riconoscibile da un 2-PDA: Un 2-PDA può riconoscere L come segue:
   * Usa la prima pila per contare le 'a'
   * Usa la seconda pila per contare le 'b'
   * Confronta le 'c' con entrambe le pile
2. L non è riconoscibile da un 1-PDA: Supponiamo per assurdo che esista un 1-PDA P che riconosce L. Sia k il numero di stati di P. Consideriamo la stringa w = a^m b^m c^m con m > k. Durante la lettura di a^m b^m, P deve memorizzare informazioni su m nella sua pila. Ma dopo aver letto b^m, P non può conservare abbastanza informazioni per verificare c^m. Usando il pumping lemma per linguaggi context-free, possiamo dimostrare che se P accetta L, allora accetterebbe anche stringhe non in L, come a^m b^m c^(m+1).

Quindi, L è riconoscibile da un 2-PDA ma non da un 1-PDA, dimostrando che i 2-PDA sono più potenti.

- 2-PDA ⊆ TM: un 2-PDA può essere simulato da una TM che usa

una traccia del nastro per ogni pila e lo stato finito.

Quindi i 2-PDA sono al più potenti quanto le TM.

- TM ⊆ 2-PDA: Una TM può essere simulata da un 2-PDA che usa:

- Una pila per simulare il nastro sinistro della TM

- Una pila per simulare il nastro destro

- Uno stato finito per tenere traccia dello stato e del simbolo sotto la testina

Il 2-PDA può simulare ogni mossa della TM con un numero finito di mosse.

Quindi i 2-PDA sono almeno potenti quanto le TM.

Combinando i due punti, i 2-PDA e le TM sono equivalenti.