Algoritmi di decidibilità e dimostrazione di non regolarità di un linguaggio

Created	@May 5, 2024 9:53 PM
Class	Fondamenti di informatica: linguaggi formali

Un algoritmo di decidibilità prende in input un automa, e dà in output se l'automa soddisfa una determinata condizione. Uno tra gli algoritmi di decidibilità è proprio quello che riconosce la non regolarità di un linguaggio.

Linguaggi regolari e accettazione

$L = \{a^n b^m \mid mn \geq 1\}$ è un linguaggio regolare.

Automa che accetta questo linguaggio -

$L = \{a^n b^n \mid 1 \leq n \leq 3\}$ è un linguaggio regolare.

Automa che accetta questo linguaggio -

$L = \{a^n b^n \mid n \geq 0\}$ è un linguaggio non regolare, dal momento che nessun automa riesce a confrontare il numero di stati di a e b, un possibile automa che riconosce questo linguaggio avrà un numero infinito di stati.

Dimostrazione non regolarità di un linguaggio

💡 (Ricordanza corollario Pumping Lemma):

- $\forall\ L\ regolare\ \exists\ > 0$
- $\forall z \in L,\ |z| \geq n,\ \exists u, v, w:\ |u, v \leq n|\ v \neq \epsilon,\ z = uvw$
- $\forall i \geq 0,\ uv^i w \in L$

Per dimostrare la non regolarità di un linguaggio procediamo a negare il corollario del Pumping Lemma:

- $\forall n > 0$
- $\exists\ z \in L,\ |z| \geq n$
- $\forall u, v, w : z = uvw,\ |uv| \leq n,\ v \neq \epsilon$
- $\exists\ i \geq 0,\ uv^i w \notin L$

$L = \{a^k b^k\ |k \geq 0\}$ è un tipico linguaggio non regolare.

Quindi, per dimostrare che un linguaggio non è regolare, basta scegliere un qualunque n e dimostrare che $\exists\, n = a^n b^n,\ |Z_n| \geq n$

Comunque divido z in tre parti (u, v, w) riesco a trovare una $i : uv^i w \notin L$.

Si ha quindi che per ogni modo di dividere la stringa, avremo in uv solamente delle a, dal momento che $|uv| \leq n,\ quindi\ uv \in \{a*\}$.
$Siccome\ v \neq \epsilon,\ v \in \{a+\}$.