1 Automi

- Utilizzare il *pumping lemma* per mostrare che il seguente linguaggio su alfabeto $\Sigma = \{a, b, c\}$ non è regolare:
$$L = \{c^4 a^n b^m : n \geq m\}.$$
- Dimostrare che un linguaggio è regolare se e solo se esiste una espressione regolare che lo descrive.

Sia $w = c^4 a^{p-4} b^{p-4}$, $w = xyz$ il pumping e' p

$|xy| \leq p \Rightarrow xy = c^4 a^{p-4}$

$|y| > 0 \Rightarrow y = a^{p-4}$ y non puo' contenere c perche' le occorrenze di c devono rimanere 4

$\Rightarrow c^4 a^{p-4 \cdot 0} b^{p-4} = c^4 b^{p-4} \notin L$

Se $W = c^4 a^{p-4-k} a^k b^{p-4}$

Se $y = a^{p-4-k} \Rightarrow xy^0z = c^4 a^k b^{p-4}$ e $k < p-4 \Rightarrow xy^0z \notin L$

Se $y = a^k \Rightarrow xy^0z = c^4 a^{p-4-k} b^{p-4}$ e $p-4 > p-4-k \Rightarrow xy^0z \notin L$

Inoltre la suddivisione $\underbrace{c^4 a}_{x} \underbrace{a^{p-5} b}_{y} b^{p-5}$ non e' valida perche' $|xy| = 5+p-5+1 = p+1$

Sia D un NFA e sia $L = L(D)$, definisco un GNFA... finire

2 Calcolabilità

- Sia $B = \{\langle M \rangle : M$ è una TM e $L(M) = (01)^*\}$. Mostrare che $A_{TM} \leq_m B$. Cosa si può concludere sulla decidibilità di B?
- Dimostrare che esistono linguaggi che non sono Turing-riconoscibili. Fornire un esempio concreto di linguaggio che abbia questa proprietà. $(\overline{A_{TM}})$

Definisco la riduzione R come segue

- Su input $\langle M, w \rangle$
- Definisce M' con il seguente comportamento
  - Su input x
  - Se $M(w)$ accetta, M' accetta sse $x \in (01)^*$
  - Altrimenti, M' rifiuta sempre
- In output M'

$$\left.\begin{array}{l} \text{Se } \langle M, w \rangle \in A_{TM} \Rightarrow R(\langle M, w \rangle) = M' \text{ t.c. } L(M') = (01)^* \\ \text{Se } \langle M, w \rangle \notin A_{TM} \Rightarrow R(\langle M, w \rangle) = M' \text{ t.c. } L(M') = \emptyset \end{array}\right\} \Rightarrow B \text{ non e' decidibile.}$$

Supponiamo che ogni linguaggio sia riconoscibile, allora A_{TM} e $\overline{A_{TM}}$ sono riconoscibili, M_1 riconosce A_{TM} e M_2 riconosce $\overline{A_{TM}}$. Definisco M:

- Su input $\langle T, w \rangle$
- Esegue parallelamente $M_1(\langle T, w \rangle)$ e $M_2(\langle T, w \rangle)$
- Uno dei due (per ipotesi) necessariamente accetta
- Se M_1 accetta, M accetta $\langle T, w \rangle$
- Se M_2 accetta, M rifiuta $\langle T, w \rangle$

$\Rightarrow M$ e' un decisore per A_{TM}, ma cio' e' impossibile, essendo A_{TM} riconoscibile, se ne conclude che $\overline{A_{TM}}$ non e' riconoscibile. $\blacksquare$

3 Complessità

- Dimostrare che se $P = NP$, allora $EXP = NEXP$.
- Definire la complessità di spazio. Enunciare e dimostrare il teorema di Savitch.

Sappiamo che $EXP \subseteq NEXP$.

Sia $L \in NEXP$, N decide $x \in L$ in $O(2^{n^k})$

Definisco $L' = \{ \langle x, 1^{2^{n^k}} \rangle \mid x \in L \}$

La seguente TM decide L'

- Su input, controlla che e' della forma $\langle x, 1^{2^{n^k}} \rangle$
- estrae x ed esegue $N(x)$

tale TM opera in tempo polinomiale nel suo input

$L' \in NP = P \Rightarrow L' \in P$. Sia N' che decide L' in $O(n^k)$

Definisco N'' che

- Su input $x \in L$
- Crea $y = \langle x, 1^{2^{n^k}} \rangle$ (impiega $O(2^{n^k})$)
- esegue $N'(y)$

$\Rightarrow N''$ e' deterministica e decide L in $O(2^{n^k}) \Rightarrow L \in EXP$.

Con complessità di spazio si intende una funzione relativa ad una TM, che associa ad n il numero massimo di celle (distinte) utilizzate dalla TM per decidere una stringa lunga n

<u>Teo</u>(SAVITCH): $NLOG \subseteq P \cap space(\log^2(n))$

<u>Dimo</u>: Sia $A \in NLOG$, N decide A in $O(\log(n))$, N su input x può assumere al più $2^{O(\log n)} = O(n)$ differenti configurazioni. Si definisce un grafo G:

- i vertici sono le possibili configurazioni
- $(C_i, C_j) \in E$ se C_j segue da C_i in accordo con la δ
- Si può assumere (WLOG) che ci sia una sola configurazione accettante.

$\Rightarrow N(x) = 1 \Leftrightarrow \exists$ cammino da C_{init} a C_{acc}.

A si riduce a PATH e $PATH \in P \cap space(\log^2(n))$. ∎