1 Automi — 10 Points

- Mostrare che la seguente grammatica è ambigua:

$S \to TbT$
$T \to aTbT \mid bTaT \mid \varepsilon$.

- Dimostrare che se un linguaggio è regolare allora esiste una espressione regolare che lo descrive.

La grammatica e' ambigua perche' due derivazioni distinte generano la stessa stringa:

$S \to \underline{T}bT \to aTb\underline{T}bT \to a\underline{T}bbTaTbT \to ababb\underline{T}a\underline{T}b\underline{T}$ (bTaT, ba) $\to T \cdot \varepsilon = abababab$

$S \to \underline{T}bT \to aTb\underline{T}bT \to aTbaTbTb\underline{T} \to a\underline{T}ba\underline{T}b\underline{T}bab$ (aTbT, ab)

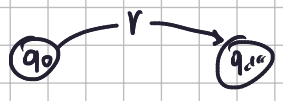
Sia $L \in REG$ e N l'NFA che decide L, trasformo N in un GNFA

- Un solo stato accettante
- Fra ogni coppia di stati c'e' un arco
- q_0 ha solo archi uscenti
- q_{acc} solo entranti

Gli archi sono etichettati con espressioni regolari

G e' il GNFA, Convert(G) e' un operazione che modifica G rimuovendo uno stato, fino a ritornare una regex

Convert(G):

- Se ci sono 2 stati ritorna la regex 

- Seleziona $q_{rip} \in Q \setminus \{q_0, q_{acc}\}$
- Definisci $G' = G$ ma rimuovi q_{rip}
- modifica la δ':

$$G' \to \delta'(q_i, q_j) = \left(\delta(q_i, q_{rip}) \delta(q_{rip}, q_{rip})^{*} \delta(q_{rip}, q_j)\right) \mid \delta(q_i, q_j) \quad (G)$$

- Ritorna G'

Al termine si avra' una REgex che genera L perche':

$L(G) = L(Convert(G))$

- w su G scaturisce la computazione $C = \{q_1, q_2 \dots q_k\}$
- Se $q_{rip} \notin C$ allora anche in G' w scaturisce C
- Se $q_{rip} \in C$ allora le etichette dei nodi adiacenti a q_{rip} sono state aggiornate considerando ogni stringa per andare da q_i a q_j passando per q_{rip} quindi w in G' porta sempre da q_1 a q_k. ■

2 Calcolabilità **10 Points**

- Sia $ALL_{NFA} = \{\langle N \rangle : N \text{ è un NFA e } L(N) = \Sigma^*\}$. Mostrare che ALL_{NFA} è decidibile.
- Dimostrare che per ogni macchina di Turing non-deterministica ne esiste una deterministica equivalente.

La seguente TM decide ALL_{DFA}:

- ha in input $\langle N \rangle$
- converte N in un DFA D (hanno lo stesso linguaggio)
- Definisce D' come segue: $D' = (\{q_0\}, \Sigma, \{\delta(q_0, a) = q_0 \ \forall a\}, q_0, \{q_0\})$
- $L(D') = \Sigma^*$
- Considera D'' tale che $L(D'') = L(D) \Delta L(D')$ con la differenza simmetrica
- D'' si può costruire facilmente e REG è chiusa per $\neg, \cup, \cap$ (coinvolti in Δ)
- esegue l'algoritmo per decidere E_{DFA}.

una TM deterministica può simulare una NTM semplicemente considerando ogni ramo di computazione in larghezza. Usa 3 nastri

nel primo mantiene l'input
nel secondo esegue il calcolo effettivo per ogni ramo
nel terzo mantiene l'indice del ramo per poter effettuare il calcolo

La TM quindi esplorerà in larghezza (BFS) i rami di computazione della NTM, ed accetterà se troverà un ramo che accetta, o rifiuterà se troverà un ramo che rifiuta.

se la NTM accetta, allora ha almeno un ramo accettante quindi la TM accetta
se la NTM rifiuta, allora ha almeno un ramo che rifiuta quindi la TM rifiuta
se la NTM va in loop, ogni ramo va in loop, e la TM di conseguenza non terminerà

10 Points

3 Complessità

- Sia L un linguaggio *PSPACE*-completo. Mostrare che se $L \in NP$, allora $NP = PSPACE$.
- Definire la classe di complessità *coNP* ed il problema UNSAT. Dimostrare che *UNSAT* è *coNP*-completo.

Se L e' PSPACE-completo, $\forall A \in PSPACE$, $A \leq_m^P L$

Se L e' NP, $\underset{\uparrow NTM}{N}$ decide L in $O(n^k)$

ogni $A \in PSPACE$ si riduce polinomialmente a L

Sia $x \in A$

- Su input x
- considero $R(x)$ // $pol_1(n)$
- eseguo $N(R(x))$ // $O(n^k)$

$\Rightarrow$ tale NTM decide A in tempo polinomiale $\Rightarrow A \in NP \Rightarrow PSPACE \subseteq NP$

Essendo SAT NP-completo e in PSPACE $\Rightarrow NP \subseteq PSPACE \Rightarrow NP = PSPACE$

$coNP = \{L \text{ t.c. } \bar{L} \in NP\}$

$UNSAT = \{\phi \text{ t.c. } \phi \notin SAT\} \Rightarrow UNSAT = \overline{SAT}$ e' in coNP

Se $A \in NP \Rightarrow A \leq_m^P SAT \Rightarrow \bar{A} \leq_m^P \overline{SAT} = UNSAT$ ma $\bar{A} \in coNP$ quindi ogni problema coNP si riduce ad UNSAT.