1 Parte Prima — 10 Points

- Sia L_k il linguaggio di tutte le stringhe su alfabeto $\{0, 1, \dots, 9\}$ la somma delle cui cifre è un multiplo di k, per un fissato $k \in \mathbb{N}$. Descrivere formalmente un DFA che riconosca L_k per ogni $k \in \mathbb{N}$. Disegnare il diagramma di transizione degli stati del DFA per il caso $k = 4$.
- Dimostrare che un linguaggio è acontestuale se e solo se esiste un PDA che lo riconosce.

Dim PDA = CFG :

[⇒]: Sia $G = (V, \Gamma, R, S)$, definisco $P = (\{q_s, q_1, q_2\}, \Sigma, \Gamma, \delta, q_s, F)$ t.c.

$$\delta(q_s, \epsilon, \epsilon) = (q_1, S\$)$$
$$\delta(q_1, \epsilon, A) = \{(q_1, w) \text{ se } A \to w \in R\}$$
$$\delta(q_1, a, a) = (q_1, \epsilon)$$
$$\delta(q_1, \epsilon, \$) = (q_2, a)$$

⇒ Il PDA deriva ogni possibile $w \in L(G)$ dentro lo stack, e se una coincide con l'input accetta.

[⇐]: Sia $P = \{Q, \Sigma, \Gamma, \delta, q_s, F\}$, definisco $G = \{V, \Gamma, R, S\}$ in tal modo

$\forall p \in Q,\ A_{pp} \to \epsilon \in R$

$\forall p, r, q \in Q,\ A_{pq} \to A_{pr}A_{rq}$ $\quad \delta^*(p, x, \epsilon) = (r, \epsilon) \wedge \delta^*(r, y, \epsilon) = (q, \epsilon)$ Se esiste un cammino da p ad r e da r a q.

$\forall p, q, r, s,$ se $\delta(p, a, \epsilon) = (r, u) \wedge \delta(s, b, u) \to (q, \epsilon)$ allora $A_{pq} \to aA_{rs}b \in R$

Lemma: $A_{pq} \to x \iff x$ porta P da p a q con pila vuota

[⇒]
- Caso base: una produzione: $A_{pp} \to \epsilon$, ma ϵ porta da p a p con pila vuota.
- I.I. : $A_{pq} \to x$ allora x //
- Passo Ind: $A_{pq} \to x$, due casi:
  - $A_{pq} \to aA_{rs}b$, ma $\delta(p, a, \epsilon) = (r, u) \wedge \delta(s, b, u) \to (q, \epsilon)$ e per ipotesi A_{rs} porta da r a s con pila vuota ⇒ $A_{pq} \to x$ e x soddisfa l'asserto.
  - $A_{pq} \to A_{pr}A_{rq}$, per ipotesi: $\begin{cases} A_{pr} \to x \\ A_{rq} \to y \end{cases} \Rightarrow xy$ porta da p a q.

[⇐]

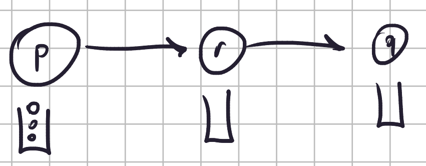
Caso base: 1 passo, puo' essere solo ϵ, ma $A_{pp} \to \epsilon \in R \Rightarrow$ da p a p senza ... //

Ip. In. : Se in k passi x porta da p a q, allora $A_{pq} \to x$

Passo Ind: x porta da p a q con pila vuota, 2 casi:

* $x = ayb$ e a fa inserire u, b lo rimuove, $A_{rs} \to y$, ma $A_{pq} \to aA_{rs}b \in R$ quindi $A_{pq} \to x$

* la pila si svuota in uno stato r



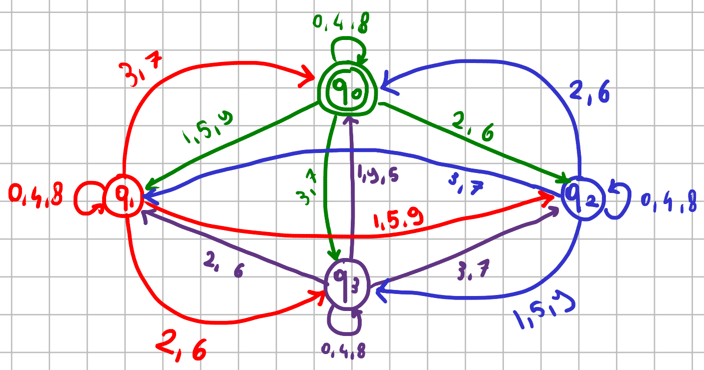
Per costruzione $A_{pq} \to A_{pr}A_{rq}$ e $\begin{cases} A_{pr} \to y \\ A_{rq} \to z \end{cases} \Rightarrow A_{pq} \to x$ ∎

Gli stati sono $\{q_{i\%K} \mid i \in \{0 \dots 9\}\}$ e sono K. q_0 e' lo stato accettante.

$\delta(q_{i\%K}, j) = q_{i+j\%K}$ $\forall i, j$. Nel caso $K = 4$:

$Q = \{q_0, q_1, q_2, q_3\}$

$\delta(q_0, 0) = q_0$	$\delta(q_1, 0) = q_1$	$\delta(q_2, 0) = q_2$	$\delta(q_3, 0) = q_3$
$\delta(q_0, 1) = q_1$	$\delta(q_1, 1) = q_2$	$\delta(q_2, 1) = q_3$	$\delta(q_3, 1) = q_0$
$\delta(q_0, 2) = q_2$	$\delta(q_1, 2) = q_3$	$\delta(q_2, 2) = q_0$	$\delta(q_3, 2) = q_1$
$\delta(q_0, 3) = q_3$	$\delta(q_1, 3) = q_0$	$\delta(q_2, 3) = q_1$	$\delta(q_3, 3) = q_2$
$\delta(q_0, 4) = q_0$	$\delta(q_1, 4) = q_1$	$\delta(q_2, 4) = q_2$	$\delta(q_3, 4) = q_3$
$\delta(q_0, 5) = q_1$	$\delta(q_1, 5) = q_2$	$\delta(q_2, 5) = q_3$	$\delta(q_3, 5) = q_0$
$\delta(q_0, 6) = q_2$	$\delta(q_1, 6) = q_3$	$\delta(q_2, 6) = q_0$	$\delta(q_3, 6) = q_1$
$\delta(q_0, 7) = q_3$	$\delta(q_1, 7) = q_0$	$\delta(q_2, 7) = q_1$	$\delta(q_3, 7) = q_2$
$\delta(q_0, 8) = q_0$	$\delta(q_1, 8) = q_1$	$\delta(q_2, 8) = q_2$	$\delta(q_3, 8) = q_3$
$\delta(q_0, 9) = q_1$	$\delta(q_1, 9) = q_2$	$\delta(q_2, 9) = q_3$	$\delta(q_3, 9) = q_0$



2 **Parte Seconda** 10 Points

Sia $HALTS\_ON\_ALL_{TM}$ il linguaggio che consiste di tutte le stringhe della forma $\langle M \rangle$ tali che M è una macchina di Turing che termina sempre per ogni possibile scelta dell'input. Mostrare che $HALTS\_ON\_ALL_{TM}$ è indecidibile.

Dimostrare che il linguaggio $EQ_{TM} = \{\langle M_1, M_2 \rangle : L(M_1) = L(M_2)\}$, dove M_1 ed M_2 sono macchine di Turing, è indecidibile.

$HALTS\_ON\_ALL_{TM}$ e' l'insieme di tutti i decisori.

Definisco R su input $\langle M, w \rangle$

- Crea M' t.c.
  - Su input x
  - Se $x \neq w$, rifiuta

  se M(w) rifiuta, va in loop
  - Sennò accetta

$\Rightarrow A_{TM} \leq_m$ HA... $\Rightarrow$ e' indecidibile

- Ritorna M'

Definisco R per $A_{TM} \leq EQ_{TM}$

- Su input $\langle M, w \rangle$
- Definisce M' che
  - accetta sempre
- Definisce M'' che
  - Se $x \neq w$ accetta
  - esegue M(w) e fa la stessa cosa

Ritorna $\langle M', M'' \rangle$

3 Parte Terza 10 Points

- Dimostrare che $NTIME(n^k) \subsetneq PSPACE$ per ogni $k \in \mathbb{N}$. Questo implica che $NP \subsetneq PSPACE$? Motivare la risposta.
- Definire il concetto di riduzione. Dimostrare che $3\text{-}COL \le_m^p SAT$, dove $3\text{-}COL$ è il linguaggio dei grafi 3-colorabili.

Riduzione: $A \le_m^p B$ se $\exists R$ t.c. $w \in A \Leftrightarrow R(w) \in B$ e $L(M) \in P$

Definisco una riduzione per 3-COL

x_i	x_i'	COL
0	0	1
0	1	2
1	0	3

$\forall$ nodo $i \in V(G)$ si hanno x_i, x_i', codificano un colore

$\forall (i,j) \in E(G)$, $(x_i, x_i') \neq (x_j, x_j')$ e $\forall_i$ $\overline{(x_i \wedge x_i')}$ non devono codificare il colore $(1,1)$. $\phi_1 = \bigwedge_i \overline{(x_i \wedge x_i')}$

$\forall (i,j) \in E(G)$ c'e' $\phi_{i,j}$ $\overline{(x_i \leftrightarrow x_i') \wedge (x_j \leftrightarrow x_j')}$

$\phi = (\bigwedge_{i,j} \phi_{i,j}) \wedge \phi_1$

il tempo limita lo spazio, $NTime(n^k) \subseteq NSpace(n^k)$ $\forall k$ ma $NSPACE(n^k) = DSPACE(n^{2k}) \subseteq PSPACE \Rightarrow \forall k$ $NTime(n^k) \subseteq PSPACE$. Devo mostrare che $NTime(n^k) \neq PSPACE$ $\forall k$. Ossia, che esiste $L \in PSPACE$ t.c. $L \notin NTime(n^k)$ $\forall k$.