Macchina a Stato (Minsky 1956, Wang 1957)

- **Sintassi** (ovviamente, astratta)

  $\mathcal{Z} = \texttt{Zero R} \mid \texttt{Inc R} \mid \texttt{DJO R N} \mid \texttt{Halt}$

  - **Termini:** $\mathcal{Z}$ Insieme (numerabile) degli Statements, $\texttt{z}, \texttt{z}_\texttt{i}$.
    - **Azzeramento** registro R;
    - **Incremento** registro R;
    - **Decremento o salto** a N se R contiene 0;
    - **Arresto** esecuzione;
  - **Registri:** $\texttt{R}_\texttt{i}$. Insieme finito di Registri a *capacità infinita*.
  - **N:** interi nonegativi
  - **Programmi,** $\mathcal{P}$**:**
    - Ogni sequenza finita di $\mathcal{Z}$;
    - Ogni statement accessibile attraverso la propria posizione nella sequenza

- **Esempi**

  $\texttt{Zero R}_0; \texttt{Halt}$

  $\texttt{Zero R}_0; \texttt{DJO R}_1\ 4; \texttt{Inc R}_2; \texttt{DJO R}_0\ 1; \texttt{Halt}$

Macchina a Stato: Semantica

$\mathcal{Z} = \texttt{Zero R} \mid \texttt{Inc R} \mid \texttt{DJO R N} \mid \texttt{Halt}$

- Semantica SOS
  - Definizione di **stato**
    - Registri (variabili con nomi predefiniti) rappresentati da una sequenza di coppie, ρ, della forma: $(\texttt{R}_0, \texttt{n}_0)...(\texttt{R}_\texttt{k}, \texttt{n}_\texttt{k})$
    - Program Counter: Posizione PC del successivo statement da eseguire (inizialmente 0).
    - Uno stato è rappresentato da una coppia $\{\texttt{PC}, \rho\}$
  - Definizione di **Transizione** $\rightarrow$
    - Esecuzione di Statement z nello stato σ della macchina
      - $\langle \texttt{z}, \sigma \rangle \rightarrow \sigma'$, indicante ...
      - $\langle \texttt{z}, \sigma \rangle \rightarrow \langle \texttt{z}', \sigma' \rangle$, indicante ...

$\mathcal{Z} = \texttt{Zero R} \mid \texttt{Inc R} \mid \texttt{DJO R N} \mid \texttt{Halt}$

- Semantica SOS
  - Definizione di **stato**
  - Definizione di **Transizione**

  - Definizione di **Computazione** di un programma di $\mathtt{p} \in \mathcal{P}$:
    - sequenza componente registro degli stati attraversati nell'esecuzione di $\mathtt{p}$.

      $$\langle \mathtt{z}, \{\mathtt{n}, \rho\}\rangle \to \langle \mathtt{z}_1, \{\mathtt{n}_1, \rho_1\}\rangle \to \langle \mathtt{z}_2, \{\mathtt{n}_2, \rho_2\}\rangle \to \ldots$$
    - La semantica di un programma è la sua computazione:

      $$\mathtt{Sem(p)} \equiv \rho, \rho_1, \rho_2, \ldots,.$$

  - **Notazione**.
    - $\{\mathtt{n}, (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_k, \mathtt{n}_k)\}$ stato con $\mathtt{PC} \equiv \mathtt{n}$ e k registri
    - ρ, τ (anche con pedici) sono componente registro dello stato
    - $\rho(\mathtt{R}_i) = 0$ quando $\rho = (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_i, 0), \cdots, (\mathtt{R}_k, \mathtt{n}_k)$,
    - $\rho(\mathtt{R}_i) \neq 0$ quando $\rho = (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_i, \mathtt{n}_i \neq 0), \cdots, (\mathtt{R}_k, \mathtt{n}_k)$,
    - $\rho[\mathtt{R}_i \leftarrow 0] = (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_i, 0), \cdots, (\mathtt{R}_k, \mathtt{n}_k)$
    - $\rho[\mathtt{R}_i \leftarrow \mathtt{R}_i + 1] = (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_i, \mathtt{n}_i + 1), \cdots, (\mathtt{R}_k, \mathtt{n}_k)$
      quando $\rho = (\mathtt{R}_1, \mathtt{n}_1), \cdots, (\mathtt{R}_i, \mathtt{n}_i), \cdots, (\mathtt{R}_k, \mathtt{n}_k)$
    - $\mathtt{P[n]}$ $\mathtt{n}$-esimo statement del programma $\mathtt{P}$
    - $\#\mathtt{P}$ numero di statements del programma (posizionati da 0 a $\#\mathtt{P} - 1$)

Macchina a Stato: Transizioni

$$\mathcal{Z} = \texttt{Zero R} \mid \texttt{Inc R} \mid \texttt{DJO R N} \mid \texttt{Halt}$$

$$\langle \texttt{Halt}, \{\texttt{n}, \rho\} \rangle \rightarrow \{\texttt{n}, \rho\}$$

$$\frac{0 \leq \texttt{n} < \#\texttt{P}, \quad \texttt{z}_\texttt{n} = \texttt{P}[\texttt{n}]}{\langle \texttt{Zero}\, \texttt{R}_\texttt{i}, \{\texttt{n}, \rho\} \rangle \rightarrow \langle \texttt{z}_\texttt{n}, \{\texttt{n}+1, \rho[\texttt{R}_\texttt{i} \leftarrow 0]\} \rangle}$$

$$\frac{0 \leq \texttt{n} < \#\texttt{P}, \quad \texttt{z}_\texttt{n} = \texttt{P}[\texttt{n}]}{\langle \texttt{Inc}\, \texttt{R}_\texttt{i}, \{\texttt{n}, \rho\} \rangle \rightarrow \langle \texttt{z}_\texttt{n}, \{\texttt{n}+1, \rho[\texttt{R}_\texttt{i} \leftarrow \texttt{R}_\texttt{i} + 1]\} \rangle}$$

$$\frac{0 \leq \texttt{m} < \#\texttt{P}, \quad \texttt{z}_\texttt{m} = \texttt{P}[\texttt{m}], \quad \rho(\texttt{R}_\texttt{i}) = 0}{\langle \texttt{DJO}\, \texttt{R}_\texttt{i}\, \texttt{m}, \{\texttt{n}, \rho\} \rangle \rightarrow \langle \texttt{z}_\texttt{m}, \{\texttt{m}+1, \rho\} \rangle} \qquad \frac{0 \leq \texttt{n} < \#\texttt{P}, \quad \texttt{z}_\texttt{n} = \texttt{P}[\texttt{n}], \quad \rho(\texttt{R}_\texttt{i}) \neq 0}{\langle \texttt{DJO}\, \texttt{R}_\texttt{i}\, \texttt{m}, \{\texttt{n}, \rho\} \rangle \rightarrow \langle \texttt{z}_\texttt{n}, \{\texttt{n}+1, \rho[\texttt{R}_\texttt{i} \leftarrow \texttt{R}_\texttt{i} - 1]\} \rangle}$$

Esercizi

1. Si mostri la computazione di $\langle \mathtt{P}, \{0, \rho_0\}\rangle$, dove $\rho_0 \equiv \{0, (\mathtt{R}_0, 5), (\mathtt{R}_1, 3), (\mathtt{R}_2, 4)\}$ $\mathtt{P} \equiv \mathtt{Zero}\ \mathtt{R}_0; \mathtt{DJ0}\ \mathtt{R}_1\ 4; \mathtt{Inc}\ \mathtt{R}_2; \mathtt{DJ0}\ \mathtt{R}_0\ 1; \mathtt{Halt}$. Si assuma il primo statement di P alla posizione 0.
2. Si mostri la computazione del programma P sopra, nello stato $\{0, \rho\}$, dove ρ_0 sia un'arbitraria configurazione dei registri $\mathtt{R}_0, \mathtt{R}_1, \mathtt{R}_2$.
3. Si scriva un programma che calcolato in uno stato iniziale arbitrario ρ_0 termini avendo copiato nel registro $\mathtt{R}_1$ il contenuto del registro $\mathtt{R}_2$.
4. Lo stesso esercizio precedente salvo che nello stato finale ρ_k deve valere:
   $\rho_k(\mathtt{R}_1) = \rho_0(\mathtt{R}_1) = \rho_k(\mathtt{R}_2)$.
5. Si scriva un programma che calcolato in uno stato iniziale ρ_0 termini scrivendo 0 nel registro $\mathtt{R}_0$ se $\rho_0(\mathtt{R}_1) \neq \rho_0(\mathtt{R}_2)$. Il programma termina scrivendo 1, in caso contrario.
6. Si mostri la computazione del programma P sotto, nello stato $\{0, \rho_0\}$, dove ρ_0 sia un arbitraria configurazione dei registri $\mathtt{R}_0, \mathtt{R}_1, \mathtt{R}_2$.
   $\mathtt{P} \equiv \mathtt{Zero}\ \mathtt{R}_0; \mathtt{Inc}\ \mathtt{R}_1; \mathtt{DJ0}\ \mathtt{R}_0\ 1; \mathtt{Halt}$.
7. Si scriva un programma che calcolato in uno stato iniziale ρ_0 termini in uno stato ρ_k tale che $\rho_k(\mathtt{R}_2) = \rho_0(\mathtt{R}_2) - \rho_0(\mathtt{R}_1)$.
8. Cosa implica l'assunzione che i registri della Macchina a Stato sebbene in numero finito abbiano capacità illimitata?

Altri esercizi in: EserciziL5