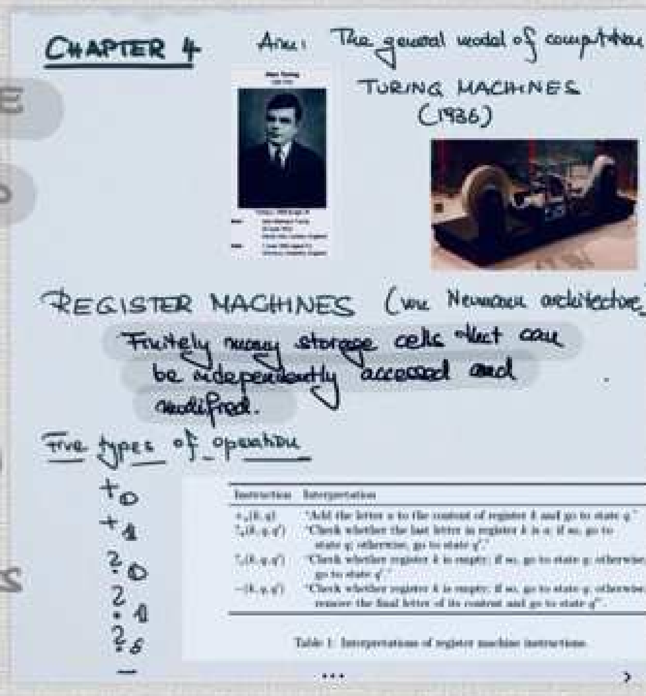
XIV

Fourteenth Lecture
14 NOVEMBER 2024

Automata & Formal Languages

RECAP (Chapter 4)

LECTURE XIII RECAP



CHAPTER 4

Aim: The general model of computation

TURING MACHINES (1936)

REGISTER MACHINES (von Neumann architecture)

Finitely many storage cells that can be independently accessed and modified.

Five types of operation

$+_0$
$+_1$
$?_0$
$?_1$
$?_\varepsilon$
$-$

Instruction	Interpretation
$+_a(k,q)$	"Add the letter a to the content of register k and go to state q."
$?_a(k,q,q')$	"Check whether the last letter in register k is a; if so, go to state q; otherwise, go to state q'."
$?(k,q,q')$	"Check whether register k is empty; if so, go to state q; otherwise, go to state q'."
$-(k,q,q')$	"Check whether register k is empty; if so, go to state q; otherwise, remove the final letter of its content and go to state q'".

Table 1: Interpretations of register machine instructions

Register Machine

$M = (Q, P)$

Q set of states with $q_S \neq q_H$ START & HALT STATE

P is the PROGRAM assigning Q-instructions

Program has finitely many registers mentioned: UPPER REGISTER INDEX

CONFIGURATION: $(q, w_0, \ldots, w_n)$

A REGISTER MACHINE TRANSFORMS A CONFIGURATION

COMPUTATION SEQUENCE: $C(k, M, \vec{w})$

If the state of some $C(k, M, \vec{w})$ is q_H, we say COMPUTATION HALTS and the register content at the time is the OUTPUT.

§4.2 Performing operations & answering questions

$f: X \dashrightarrow Y$ partial function ~~from~~ from X to Y.

We say that M **performs the operation** $F: \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}^{n+1}$ if

$F(\vec{w})\uparrow \iff$ M does not halt with input $\vec{w}$

$F(\vec{w})\downarrow = \vec{v} \iff$ M halt with input $\vec{w}$ and has reg. content $\vec{v}$ at time of halting.

Examples

NEVER HALT Operation: $F: \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}^{n+1}$ with $\mathrm{dom}(F) = \emptyset$

RM: $q_S \longmapsto +_{\mathbb{0}}(0, q_S)$

Or any RM that never reaches the state q_H.

HALT W/O CHANGING ANYTHING Operation $F: \mathbb{B}^{n+1} \longrightarrow \mathbb{B}^{n+1}$ $F(\vec{w}) = \vec{w}$

RM: $q_S \longmapsto ?_\varepsilon(0, q_H, q_H)$

Or any RM that immediately halts without change. OR OTHERS!

DELETE THE FINAL LETTER OF REGISTER i, IF IT EXISTS $q_S \longmapsto -(i, q_H, q_H)$

DELETE THE CONTENT OF REGISTER i $q_S \longmapsto -(i, q_H, q_S)$

ADD $\mathbb{0}$ TO REGISTER i $q_S \longmapsto +_{\mathbb{0}}(i, q_H)$

$\mathbb{1}$ $+_{\mathbb{1}}$

LEMMA 4.6 (Subroutine Lemma)

If M performs F, M' performs F', then there is $\hat{M}$ that performs $F' \circ F$

Proof. W.l.o.g., assume that $Q \cap Q' = \{q_H\}$ and $q_H = q'_S$

Let $\hat{Q} := Q \cup Q'$.

Let $\hat{P} := P^* \cup P'$ where P^* is P with $(q_H, P(q_H))$ removed.

Then $\hat{M} = (\hat{Q}, \hat{P})$ performs $F' \circ F$.

q_S start state

q'_H halt state

q.e.d.

Example

ADD w TO REGISTER i.

By "add 0/1", we have machines adding single letters; let $|w| = n$; apply the Subroutine Lemma n times with appropriate machines.

Answering Questions

A question with $k+1$ answers about $n+1$-tuples is a partition $W = \{A_0, \dots, A_k\}$ of $\mathbb{B}^{n+1}$.

$$\Big[\, \bigcup A_i = \mathbb{B}^{n+1} \;;\; \underset{i \neq j}{A_i \cap A_j = \emptyset} \,\Big]$$

A RM M <u>answers a question W</u> if it has $k+1$ <u>answer states</u> $\hat{q}_0, \dots, \hat{q}_k$ and for every $\vec{w} \in \mathbb{B}^{n+1}$ M reaches upon receiving input $\vec{w}$ in a finite amount of steps, say k, one of the answer states $\hat{q}_i$ and

$$C(k, M, \vec{w}) = (\hat{q}_i, \vec{w})$$

and $\vec{w} \in A_i \iff$ that state is $\hat{q}_i$.

Answering Questions

EXAMPLES

IS REGISTER i EMPTY ?

$$A_0 := \{\vec{w}\,;\ w_i = \varepsilon\}$$
$$A_1 := \{\vec{w}\,;\ w_i \neq \varepsilon\}$$
$$q_S \longmapsto ?_\varepsilon(i, \hat{q}_0, \hat{q}_1)$$

DOES REGISTER i END WITH $\mathbb{0}$? $\mathbb{1}$

$$A_0 := \{\vec{w}\,;\ w_i = v\mathbb{0}\}$$
$$A_1 := \mathbb{B}^{n+1} \setminus A_0$$

$$q_S \longmapsto ?_{\underset{\mathbb{1}}{\mathbb{0}}}(i, \hat{q}_0, \hat{q}_1)$$

WHAT IS THE FINAL LETTER OF REGISTER i ?

$$A_0 := \{\vec{w}\,;\ w_i = v\mathbb{0}\}$$
$$A_1 := \{\vec{w}\,;\ w_i = v\mathbb{1}\}$$
$$A_2 := \{\vec{w}\,;\ w_i = \varepsilon\}$$

Take machines $M_{\mathbb{0}}$ checking $\mathbb{0}$ $\quad q_S^{\mathbb{0}}\ q_H^{\mathbb{0}}\ \hat{q}_0^{\mathbb{0}}\ \hat{q}_1^{\mathbb{0}}$

$M_{\mathbb{1}}$ checking $\mathbb{1}$ $\quad q_S^{\mathbb{1}}\ q_H^{\mathbb{1}}\ \hat{q}_0^{\mathbb{1}}\ \hat{q}_1^{\mathbb{1}}$

As before, assume $Q_{\mathbb{0}} \cap Q_{\mathbb{1}} = \{q_S^{\mathbb{1}}\}$

with $\hat{q}_1^{\mathbb{0}} = q_S^{\mathbb{1}}$.

Combine $Q := Q_{\mathbb{0}} \cup Q_{\mathbb{1}}$

and P by combining $P_{\mathbb{0}}, P_{\mathbb{1}}$ removing superfluous instructions.

$$q_S := q_S^{\mathbb{0}} \qquad \hat{q}_0 := \hat{q}_0^{\mathbb{0}} \qquad \hat{q}_1 := \hat{q}_0^{\mathbb{1}} \qquad \hat{q}_2 := \hat{q}_1^{\mathbb{1}}$$

Lemma 4.7 (Case Distinction Lemma). Let $W = \{A_i\,;\, i \leq k\}$ be a question with $k+1$ answers and $f_i\colon \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}^{n+1}$ be operations for $i \leq k$. If W is answered by a register machine $M = (Q, P)$ and f_i is performed by $M_i := (Q_i, P_i)$ (for $i \leq k$), then we can construct a register machine that performs the operation defined by $g(\vec{w}) := f_i(\vec{w})$ if and only of $\vec{w} \in A_i$.

Let $W = \{A_0, A_1\}$ be a question about n-tuples with two answers and $F\colon \mathbb{B}^n \dashrightarrow \mathbb{B}^n$ an operation. Define by recursion $F^0(\vec{w}) := \vec{w}$ and $F^{m+1}(\vec{w}) := F(F^m(\vec{w}))$ and

$$R_{F,W}(\vec{w}) := \begin{cases} F^m(\vec{w}) & \text{if } m \text{ is the least number such that } F^m(\vec{w}) \in A_1 \text{ and} \\ \uparrow & \text{if there is no such number.} \end{cases}$$

The operation $R_{F,W}$ can be described as "repeat F until the answer to W is A_1".

Lemma 4.8 (Repeat Lemma). If $W = \{A_0, A_1\}$ be a question about n-tuples with two answers that can be answered by a register machine and $F\colon \mathbb{B}^n \dashrightarrow \mathbb{B}^n$ an operation performed by a register machine. Then $R_{F,W}$ is performed by a register machine.

LEMMA 4.7 Case Distinction Lemma

$g(\vec{w}) := f_i(\vec{w})$ where $\vec{w} \in A_i$.

Question W with $k+1$ answers

$f_i : \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}^{n+1}$

Claim. If W answered by RM then g performed by RM

f_i performed by RM

Proof

M answers W $\quad Q \quad P \quad q_s \quad \hat{q}_i$

M_i performs f_i $\quad Q_i \quad P_i \quad q_s^i \quad q_H^i$.

W.l.o.g., assume that $Q \cap \bigcap_{i \le k} Q_i = \{ q_s^i, i \le k \}$

with $\hat{q}_i = q_s^i$.

As before, combine the machines, removing unnecessary instructions for $\hat{q}_i$.

q.e.d.

LEMMA 4.8 REPEAT LEMMA

If F is performed by a machine,
W is answered by a machine
then $R_{F,W}$ is performed by machine.

Proof M perform F Q P q_S q_H

M' answers W Q' P' q'_S $\hat{q}_0$ $\hat{q}_1$

Construct $\hat{M}$ by letting q'_S be the start state, identifying $\hat{q}_1$ and q_S and q_H and q'_S and letting $\hat{q}_0$ be the halt state. q.e.d.

If $F : \mathbb{B}^{n+1} \dashrightarrow \mathbb{B}^{n+1}$,

iterate $F^0(\vec{w}) := \vec{w}$

$$F^{n+1}(\vec{w}) := F(F^n(\vec{w}))$$

If W is question with two answers A_0, A_1

$$R_{F,W}(\vec{w}) := \begin{cases} F^m(\vec{w}) & \text{if } m \text{ is least s.t. } F^m(\vec{w}) \in A_0 \\ \uparrow & \text{o/w} \end{cases}$$

Repeat of F, W

MORE EXAMPLES (Example 4.9)

REPLACE CONTENT OF REG i WITH w.

Empty register i.
Add w to register i.

COPY FINAL LETTER OF REG i TO REG j, IF IT EXISTS

Determine final letter.
If 0, write 0 in j.
If 1, write 1 in j.
If ε, do nothing.

MOVE FINAL LETTER OF REG i TO REG j, IF IT EXISTS

Copy final letter from i to j.
Remove final letter from i.

MOVE CONTENT OF REG i TO REG j IN REVERSE ORDER

Repeat:
move final letter
Until empty.

MOVE CONTENT OF REG i TO REG j.

Take unused reg. k.
Empty it.
Move i to k in reverse order.
Move k to j in reverse order.

COPY CONTENT OF REG i TO REG j IN REVERSE ORDER

COPY CONTENT OF REG i TO REG j.

WILL BE DONE IN LECTURE XV