BLG311E Formal Languages and Automata

Turing Machine and Introduction to Theory of Computation

A.Emre Harmancı Osman Kaan Erol Tolga Ovatman

2012

Classical automata (a.k.a. language recognizers(DFA,NFA,PDA)) sometimes is incapable of recognizing very simple language(e.g. $a^n b^n c^n : n \geq 0$).
There exists more general language recognizers which perform transformation between chains of tokens. Turing machine is an example to this kind of language recognizers.

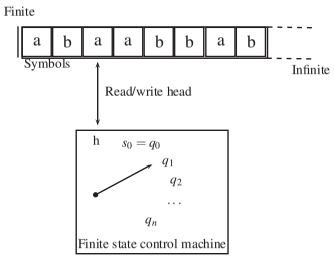
Church-Turing Thesis

In computability theory, the Church-Turing thesis is a combined hypothesis about the nature of functions whose values are effectively calculable; or, in more modern terms, functions whose values are algorithmically computable. In simple terms, the Church-Turing thesis states that a function is algorithmically computable if and only if it is computable by a Turing machine.
Informally the Church-Turing thesis states that if some method (algorithm) exists to carry out a calculation, then the same calculation can also be carried out by a Turing machine (as well as by a recursively definable function, and by a λ-function).

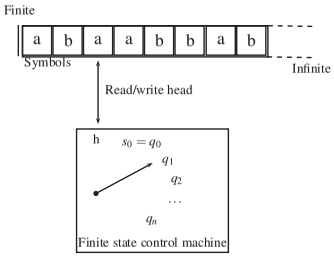
Classical automata (a.k.a. language recognizers(DFA,NFA,PDA)) sometimes is incapable of recognizing very simple language(e.g. $a^n b^n c^n : n \geq 0$).
There exists more general language recognizers which perform transformation between chains of tokens. Turing machine is an example to this kind of language recognizers.

Church-Turing Thesis

The Church-Turing thesis is a statement that characterizes the nature of computation and cannot be formally proven. Even though the three processes mentioned above proved to be equivalent, the fundamental premise behind the thesis - the notion of what it means for a function to be effectively calculable - is “a somewhat vague intuitive one”. Thus, the “thesis” remains a conjecture.



Read/write head reads the corresponding symbol on the tape and according to the state of the control machine it either writes a new symbol, or it moves left(L) or right(R). If the machine attempts to move further left from the leftmost symbol on the tape than this situation is denoted as the “machine halts”. Head may move as much as possible towards right.



Symbols:

- h: End of computation. Machine halts. Halt is not included in input, output and state alphabet.
- #: blank symbol
- L,R: Symbols indicating left and right movement. Those are not included in input or output alphabet.
- Input symbols are written on the leftmost side by convention.

Formal Definition of a TM

A turing machine(TM) is a quadruple $M = (S, \Sigma, \Delta, s_0)$, where:

- S: A finite, non-empty set of states. Usually $h \notin S$.
- Σ: Input/output alphabet. $\# \in \Sigma$ but $L, R \notin \Sigma$
- $s_0 \in S$: An initial state, an element of S.
- δ: The state-transition function $S \times \Sigma \rightarrow S \cup \{h\} \times (\Sigma \cup \{L, R\})$

$q \in S \wedge a \in \Sigma \wedge \delta(q, a) = (p, b)$ $q \rightarrow p$

i) (Read)$a \rightarrow b \in \Sigma$ (write on tape, in place of a)
ii) $b = L$ (head right)
iii) $b = R$ (head left)

Formal Definition of a TM

We can match modern computers with Turing Machines; RAM can be matched with the tape, computer programs can be matched with the state table, microprocessor(excluding I/O untis) can be matched with control unit of the TM. In order to improve understandibility of the state table, following notion can be used:
$q, \sigma, q', \sigma', HM$ where

- q stands for current state of the machine
- q' stands for the next state
- σ stands for read symbol
- σ' stands for symbol to be written (same with σ for no write)
- HM stands for head move where $HM \in -1 : L, 0 : None, 1 : R$

Example 1

A machine that erases all the symbols from left to right.
$S = \{q_0, q_1\}$
$\Sigma = \{a, \#\}$
$s_0 = q_0$

q	σ	$\delta(q, \sigma)$
q_0	a	$(q_1, \#)$
q_0	$\#$	$(h, \#)$
q_1	a	(q_0, a)[1]
q_1	$\#$	(q_0, R)

[1] This row is to conform the formal definition of a function

Example 1

A machine that erases all the symbols from left to right.
$S = \{q_0, q_1\}$
$\Sigma = \{a, \#\}$
$s_0 = q_0$
Now the same machine with a different notion and fewer lines

q	σ	$\delta(q, \sigma), HM$
q_0	a	$(q_0, \#), 1$
q_0	$\#$	$(h, \#), 0$

Example 2

A machine that moves towards left and recognizes a symbols, halting with the first # symbol
$S = \{q_0\}$
$\Sigma = \{a, \#\}$
$s_0 = q_0$

q	σ	$\delta(q, \sigma)$
q_0	a	(q_0, L)
q_0	$\#$	$(h, \#)$

Configuration of a TM

A configuration is an element of the following set:
$S \cup \{h\} \times \Sigma^* \times \Sigma \times (\Sigma^*(\Sigma - \{\#\}) \cup \{\Lambda\})$
For instance(assume the head is on the underlined symbol):
(q, aba, a, bab) or $(q, aba\underline{a}bab)$
$(h, \#\#, \#, \#a)$ or $(h, \#\#\underline{\#}\#a)$
(q, Λ, a, aba) or $(q, \Lambda\underline{a}aba)$ or $(q, \underline{a}aba)$
$(q, \#a\#, \#, \Lambda)$ or $(q, \#a\#\underline{\#}\Lambda)$ or $(q, \#a\#\underline{\#})$
Following situation doesn't conform with configuration definition:
$(q, baa, a, bc\#)$ or $(q, baa\underline{a}bc\#)$

Configuration of a TM

Yields in one step for TM

$(q_1, \omega_1, a_1, u_1) \vdash_M (q_2, \omega_2, a_2, u_2)$
Here $\delta(q_1, a_1) = (q_2, b)$ and $b \in \Sigma \cup \{L, R\}$

1) $b \in \Sigma, \omega_2 = \omega_1, u_2 = u_1, a_2 = b$ (a_2 is written over a_1)
2) $b = L, \omega_1 = \omega_2 a_2; (a_1 = \# \wedge u_1 = \Lambda \Rightarrow u_2 = \Lambda) \vee (a_1 \neq \# \vee u_1 \neq \Lambda \Rightarrow u_2 = a_1 u_1)$. Left movement: If the head is on a blank and no more symbols to the right blank is erased. If the head is on a symbol or some symbols exist on the right the situation is preserved.
3) $b = R, \omega_2 = \omega_1 a_1; (u_1 = a_2 u_2) \vee (u_1 = \Lambda \Rightarrow u_2 = \Lambda \wedge a_2 = \#)$. Right movement: If there are no more symbols on the right blank is written.

Example 3

$\omega, u \in \Sigma^*; a, b \in \Sigma$ (u cannot end with #)

$\delta(q_1, a) = (q_2, b) \Rightarrow (q_1, \omega \underline{a} u) \vdash_M (q_2, \omega \underline{b} u)$

$\delta(q_1, a) = (q_2, L) \Rightarrow i)(q_1, \omega b \underline{a} u) \vdash_M (q_2, \omega \underline{b} a u)$

$ii)(q_1, \omega b \underline{\#}) \vdash_M (q_2, \omega \underline{b})$

$\delta(q_1, a) = (q_2, R) \Rightarrow i)(q_1, \omega \underline{a} b u) \vdash_M (q_2, \omega a \underline{b} u)$

$ii)(q_1, \omega \underline{a}) \vdash_M (q_2, \omega a \underline{\#})$

n steps computation

A computation of length n

A computation of length n or in other words n steps computation can be defined as:
$C_0 \vdash_M C_1 \vdash_M C_2 \vdash_M \ldots C_{n-1} \vdash_M C_n$

Example 1: A machine that erases all the symbols from left to right.

q	σ	$\delta(q, \sigma)$
q_0	a	$(q_1, \#)$
q_0	#	$(h, \#)$
q_1	a	(q_0, a)
q_1	#	(q_0, R)

$(q_0, \underline{a}aaa) \vdash_M (q_1, \underline{\#}aaa) \vdash_M (q_0, \#\underline{a}aa) \vdash_M (q_1, \#\underline{\#}aa) \vdash_M (q_0, \#\#\underline{a}a) \vdash_M$
$(q_1, \#\#\underline{\#}a) \vdash_M (q_0, \#\#\#\underline{a}) \vdash_M (q_1, \#\#\#\underline{\#}) \vdash_M (q_0, \#\#\#\#\underline{\#}) \vdash_M (h, \#\#\#\#\underline{\#})$

Turing Computable Function

Turing Computable Function

Turing computable functions can perform transformations over character strings.
$M = (S, \Sigma, \delta, s_0)$
Let's distinguish Σ_I as input alphabet and Σ_O as output alphabet;
$f(\omega) = u \wedge \omega \in {\Sigma_I}^* \wedge u \in {\Sigma_O}^* \subseteq \Sigma^*$ and
$(s_0, \#\omega\underline{\#}) \vdash_M^* (h\#u\underline{\#}), \# \notin \Sigma_I \wedge \# \notin \Sigma_O$

Example 4

A machine that inverses strings (writes b in place of a ; a in place of b)
$S = \{q_0, q_1, q_2\}$
$\Sigma = \{a, b, \#\}$
$s_0 = q_0$

q	σ	$\delta(q, \sigma)$
q_0	a	(q_1, L)
q_0	b	(q_1, L)
q_0	#	(q_1, L)
q_1	a	(q_0, b)
q_1	b	(q_0, a)
q_1	#	(q_2, R)
q_2	a	(q_2, R)
q_2	b	(q_2, R)
q_2	#	$(h, \#)$

$(q_0, \#aab\underline{\#}) \vdash_M (q_1, \#aa\underline{b}) \vdash_M$
$(q_0, \#aa\underline{a}) \vdash_M (q_1, \#a\underline{a}a) \vdash_M$
$(q_0, \#a\underline{b}a) \vdash_M (q_1, \#\underline{a}ba) \vdash_M$
$(q_0, \#\underline{b}ba) \vdash_M (q_1, \#\underline{\#}bba) \vdash_M$
$(q_2, \#\underline{b}ba) \vdash_M (q_2, \#b\underline{b}a) \vdash_M$
$(q_2, \#bb\underline{a}) \vdash_M (q_2, \#bba\underline{\#}) \vdash_M$
$(h, \#bba\underline{\#})$

Example 5

Let's represent a number n with n I symbols like $III\ldots I$. Let's build a machine that computes $f(n) = n + 1$ $S = \{q_0\}$
$\Sigma = \{I, \#\}$
$s_0 = q_0$

q	σ	$\delta(q, \sigma)$
q_0	I	(h, R)
q_0	#	(q_0, I)

a) $(q_0, \#II\underline{\#}) \vdash_M (q_0, \#II\underline{I}) \vdash_M (h, \#III\underline{\#})$

b) $(q_0, \#I^n\underline{\#}) \vdash_M (h, \#I^{n+1}\underline{\#})$

c) $(q_0, \#\#) \vdash_M (q_0, \#\underline{I}) \vdash_M (h, \#I\underline{\#})$

Turing Decidable Machine

Turing Decidable Machine

Let Σ_0 be our alphabet and $\# \notin \Sigma_0$ and L be a language $L \subseteq {\Sigma_0}^*$
If we can compute the function x_L like:

$$\forall \omega \in \Sigma_0, F_L(\omega) = \begin{cases} \text{Ⓨ} \Rightarrow \omega \in L \\ \text{Ⓝ} \Rightarrow \omega \notin L \end{cases}$$

Turing Decidable Machine

Example:
$\Sigma_0 = \{a\}; L = \{\omega \in \Sigma_0^* \mid \ |\omega| \text{|even number}\}$
$S = \{q_0, \ldots, q_6\}$
$\Sigma = \{a, Ⓨ, Ⓝ, \#\}$
$s_0 = q_0$

q	σ	$\delta(q,\sigma)$
q_0	#	(q_1, L)
q_1	a	$(q_2, \#)$
q_1	#	(q_4, R)
q_2	#	(q_3, L)
q_3	a	$(q_0, \#)$
q_3	#	(q_6, R)
q_4	#	$(q_5, Ⓨ)$
q_5	Ⓨ	(h, R)
q_5	Ⓝ	(h, R)
q_6	#	$(q_5, Ⓝ)$

Turing Decidable Machine

q	σ	$\delta(q,\sigma)$
q_0	#	(q_1, L)
q_1	a	$(q_2, \#)$
q_1	#	(q_4, R)
q_2	#	(q_3, L)
q_3	a	$(q_0, \#)$
q_3	#	(q_6, R)
q_4	#	$(q_5, Ⓨ)$
q_5	Ⓨ	(h, R)
q_5	Ⓝ	(h, R)
q_6	#	$(q_5, Ⓝ)$

a)
$(q_0, \#aa\underline{\#}) \vdash_M (q_1, \#a\underline{a}) \vdash_M (q_2, \#\underline{a\#}) \vdash_M$
$(q_3, \#\underline{a}) \vdash_M (q_0, \underline{\#\#}) \vdash_M (q_1, \underline{\#}) \vdash_M$
$(q_4, \underline{\#\#}) \vdash_M (q_5, \#\underline{Ⓨ}) \vdash_M (h, \#Ⓨ\underline{\#})$

b)
$(q_0, \#aaa\underline{\#}) \vdash_M (q_1, \#aa\underline{a}) \vdash_M (q_2, \#a\underline{a\#}) \vdash_M$
$(q_3, \#\underline{aa}) \vdash_M (q_0, \#\underline{a\#}) \vdash_M (q_1, \#\underline{a}) \vdash_M$
$(q_2, \underline{\#\#}) \vdash_M (q_3, \underline{\#}) \vdash_M (q_6, \underline{\#\#}) \vdash_M$
$(q_5, \#\underline{Ⓝ}) \vdash_M (h, \#Ⓝ\underline{\#})$

a) $(q_0, \#a^n\underline{\#}) \vdash_M{}^* (h, \#Ⓨ\underline{\#}) \Rightarrow n$ is even

b) $(q_0, \#a^n\underline{\#}) \vdash_M{}^* (h, \#Ⓝ\underline{\#}) \Rightarrow n$ is odd

Some examples

1) $\#\omega\underline{\#} \Rightarrow \#\omega\#\omega\underline{\#}$
2) $\#\omega\underline{\#} \Rightarrow \#\omega\omega^R\underline{\#}$
3) $\#\omega\omega^R\underline{\#} \Rightarrow \#Ⓨ\underline{\#}$

Let's make a demonstration run for each of these machines and try to construct a transition table.

Some examples

1)$\#\omega\underline{\#} \Rightarrow \#\omega\#\omega\underline{\#}$
Example run: $\#aba\underline{\#} \rightarrow \#aba\underline{\gamma} \rightarrow \#ab\underline{a}\gamma \rightarrow \ldots \rightarrow$
$\underline{\#}aba\gamma \rightarrow \#\underline{a}ba\gamma \rightarrow \#\underline{\alpha}ba\gamma \rightarrow \ldots \rightarrow \#\alpha ba\underline{\gamma\#} \rightarrow \#\alpha ba\underline{\gamma a} \rightarrow \ldots \rightarrow$
$\#\underline{\alpha}ba\gamma a \rightarrow \#\underline{a}ba\gamma a \rightarrow \#a\underline{b}a\gamma a \rightarrow \#a\underline{\beta}a\gamma a \rightarrow \ldots \rightarrow$
$\#a\beta a\gamma a\underline{\#} \rightarrow \#a\beta a\gamma a\underline{b} \rightarrow \ldots \rightarrow$
$\#a\underline{\beta}a\gamma ab \rightarrow \#a\underline{b}a\gamma ab \rightarrow \#ab\underline{a}\gamma ab \rightarrow \#ab\underline{\alpha}\gamma ab \rightarrow \ldots \rightarrow$
$\#ab\alpha\gamma ab\underline{\#} \rightarrow \#ab\alpha\gamma ab\underline{a} \rightarrow \ldots \rightarrow$
$\#ab\underline{\alpha}\gamma aba \rightarrow \#ab\underline{a}\gamma aba \rightarrow \#aba\underline{\gamma}aba \rightarrow \#aba\underline{\#}aba \rightarrow \ldots \rightarrow \#aba\#aba\underline{\#}$
Let's consider a special situation:
Example run: $\#\underline{\#} \rightarrow \#\underline{\gamma} \rightarrow \underline{\#}\gamma \rightarrow \#\underline{\#} \rightarrow \#\#\underline{\#}$

Some examples

2)$\#\omega\underline{\#} \Rightarrow \#\omega\omega^R\underline{\#}$
Example run:
$\#abb\underline{\#} \rightarrow \#ab\underline{b} \rightarrow \#ab\underline{\beta} \rightarrow \#ab\beta\underline{\#} \rightarrow \#ab\beta\underline{b} \rightarrow \#ab\underline{\beta}b \rightarrow \#ab\underline{b}b \rightarrow \#a\underline{b}bb \rightarrow$
$\#a\underline{\beta}bb \rightarrow \#a\beta\underline{b}b \rightarrow \#a\underline{\beta}b\underline{b} \rightarrow \#a\beta bb\underline{\#} \rightarrow \#a\beta bb\underline{b} \rightarrow \ldots \rightarrow$
$\#a\underline{\beta}bbb \rightarrow \#a\underline{b}bbb \rightarrow \#\underline{a}bbbb \rightarrow$
$\#\underline{\alpha}bbbb \rightarrow \#\alpha bbb\underline{b} \rightarrow \#\alpha bbbb\underline{\#} \rightarrow \#\alpha bbbb\underline{a} \rightarrow \#\alpha bbb\underline{b}a \rightarrow ldots \rightarrow$
$\#\underline{\alpha}bbbba \rightarrow \#\underline{a}bbbba \rightarrow \underline{\#}abbbba \rightarrow$
$\#\underline{a}bbbba \rightarrow \ldots \rightarrow \#abbbb\underline{a} \rightarrow \#abb|bba\underline{\#}$

Some examples

3)$\#\omega\omega^R\underline{\#} \Rightarrow \#Ⓨ\underline{\#}$
Example run: $\#abbbba\underline{\#} \rightarrow \#abbbb\underline{a}\# \rightarrow \#abbb\underline{b} \rightarrow \ldots \rightarrow$
$\underline{\#}abbbb \rightarrow \#\underline{a}bbbb \rightarrow \#\underline{\theta}bbbb \rightarrow \ldots \rightarrow$
$\#\theta bbbb\underline{\#} \rightarrow \#\theta bbb\underline{b} \rightarrow \#\theta bb\underline{b} \rightarrow \ldots \rightarrow$
$\#\underline{\theta}bbb \rightarrow \#\theta\underline{b}bb \rightarrow \#\theta\underline{\theta}bb \rightarrow \ldots \rightarrow$
$\#\theta\theta b\underline{b} \rightarrow \#\theta\theta bb\underline{\#} \rightarrow \#\theta\theta b\underline{b} \rightarrow \#\theta\theta\underline{b} \rightarrow \#\theta\theta\underline{\theta}b \rightarrow \#\theta\theta\underline{b} \rightarrow \#\theta\theta\underline{\theta}^2 \; rightarrow$
$\#\theta\theta\underline{\theta} \rightarrow \#\theta\underline{\theta} \rightarrow \#\underline{\theta} \rightarrow \underline{\#} \rightarrow \#\underline{Ⓨ} \rightarrow \#Ⓨ\underline{\#}$

[2] Point where machine decides to output YES

Some examples

3)$\#\omega\omega^R\underline{\#} \Rightarrow \#\underline{Ⓨ}\underline{\#}$
Example run:
$\#\theta\theta bb\underline{b} \rightarrow \#\theta\underline{\theta}bb \rightarrow \#\theta\theta\theta\underline{\#} \rightarrow \#\theta\theta\underline{b} \rightarrow \#\theta\theta\underline{\theta} \rightarrow \#\theta\theta\underline{\#}^3 \rightarrow \#\theta\theta\# \rightarrow$
$\#\theta\theta\underline{\theta} \rightarrow \ldots \rightarrow \#\underline{\theta} \rightarrow \underline{\#} \rightarrow \#\underline{Ⓝ}\underline{\#}$

[3]Point where machine decides to output NO

Some examples

3)$\#\omega\omega^R\underline{\#} \Rightarrow \#\underline{Ⓨ}\underline{\#}$
Example run:
$\#ab\underline{\#} \rightarrow \#a\underline{b} \rightarrow \#\underline{a} \rightarrow \underline{\#}a \rightarrow \#\underline{a}^4 \rightarrow \#a\underline{\#} \rightarrow \#\underline{a} \rightarrow \underline{\#} \rightarrow \#\underline{Ⓝ} \rightarrow \#\underline{Ⓝ}\underline{\#}$

[4]Point where machine decides to output NO

Some examples

3)$\#\omega\omega^R\underline{\#} \Rightarrow \#\underline{Ⓨ}\underline{\#}$
Example run: $\#aaab\underline{\#} \rightarrow \#aaa\underline{b} \rightarrow \#aa\underline{a} \rightarrow \ldots \rightarrow$
$\underline{\#}aaa \rightarrow \#\underline{a}aa^5 \rightarrow \#aaa\underline{\#} \rightarrow \#aa\underline{a} \rightarrow \#\underline{a} \rightarrow \underline{\#} \rightarrow \#\underline{Ⓝ} \rightarrow \#\underline{Ⓝ}\underline{\#}$

[5]Point where machine decides to output NO