Turing Machines

CSE 211 (Theory of Computation)

Atif Hasan Rahman

Assistant Professor
Department of Computer Science and Engineering
Bangladesh University of Engineering & Technology

Adapted from slides by

Dr. Muhammad Masroor Ali

The Turing Machine

- Neither finite automata nor pushdown automata can be regarded as truly general models for computers.
- They are not capable of recognizing even such simple languages as $\{a^n b^n c^n : n \geq 0\}$.
- The Turing machine can recognize this and many more complicated languages.
- These devices are called Turing machines after their inventor Alan Turing (1912-1954).
- These are more general than the automata previously studied.
- Their basic appearance is similar to those automata.

Alan Turing

From Wikipedia, the free encyclopedia

"Turing" redirects here. For other uses, see Turing (disambiguation).

Alan Mathison Turing OBE FRS (/ˈtjʊərɪŋ/; 23 June 1912 – 7 June 1954) was an English mathematician, computer scientist, logician, cryptanalyst, philosopher and theoretical biologist.[6] Turing was highly influential in the development of theoretical computer science, providing a formalisation of the concepts of algorithm and computation with the Turing machine, which can be considered a model of a general-purpose computer.[7][8][9] Turing is widely considered to be the father of theoretical computer science and artificial intelligence.[10] Despite these accomplishments, he was never fully recognized in his home country during his lifetime due to his homosexuality, which was then a crime in the UK.

During the Second World War, Turing worked for the Government Code and Cypher School (GC&CS) at Bletchley Park, Britain's codebreaking centre that produced Ultra intelligence. For a time he led Hut 8, the section that was responsible for German naval cryptanalysis. Here he devised a number of techniques for speeding the breaking of German ciphers, including improvements to the pre-war Polish bombe method, an electromechanical machine that could find settings for the Enigma machine. Turing played a pivotal role in cracking intercepted coded messages that enabled the Allies to defeat the Nazis in many crucial engagements, including the Battle of the Atlantic, and in so doing helped win the war.[11][12] Counterfactual history is difficult with respect to the effect Ultra intelligence had on the length of the war,[13] but at the upper end it has been estimated that this work shortened the war in Europe by more than two years and saved over 14 million lives.[11]

After the war, Turing worked at the National Physical Laboratory, where he designed the ACE, among the first designs for a stored-program computer. In 1948 Turing joined Max Newman's Computing Machine Laboratory at the Victoria University of Manchester, where he helped develop the Manchester computers[14] and became interested in mathematical biology. He wrote a paper on the chemical basis of morphogenesis[2] and predicted oscillating chemical reactions such as the Belousov–Zhabotinsky reaction, first observed in the 1960s.

Turing was prosecuted in 1952 for homosexual acts, when by the Labouchere Amendment, "gross indecency" was a criminal offence in the UK. He accepted chemical castration treatment, with DES, as an alternative to prison. Turing died in 1954, 16 days before his 42nd birthday, from cyanide poisoning. An inquest determined his death as suicide, but it has been noted that the known evidence is also consistent with accidental poisoning.[15] In 2009, following an Internet campaign, British Prime Minister Gordon Brown made an official public apology on behalf of the British government for "the appalling way he was treated". Queen Elizabeth II granted him a posthumous pardon in 2013.[16][17][18] The Alan Turing law is now an informal term for a 2017 law in the United Kingdom that retroactively pardoned men cautioned or convicted under historical legislation that outlawed homosexual acts.[19]

Contents [hide]

Alan Turing
OBE FRS

Turing aged 16

Born	23 June 1912 Maida Vale, London, England
Died	7 June 1954 (aged 41) Wilmslow, Cheshire, England
Cause of death	Cyanide poisoning
Resting place	Ashes scattered near Woking Crematorium[1]
Residence	Wilmslow, Cheshire, England
Education	King's College, Cambridge (BA, MA) Princeton University (PhD)
Known for	Cryptanalysis of the Enigma Turing's proof Turing machine Turing test Unorganised machine

Turing Award

From Wikipedia, the free encyclopedia

The **ACM A.M. Turing Award** is an annual prize given by the Association for Computing Machinery (ACM) to an individual selected for contributions "of lasting and major technical importance to the computer field".[2] The Turing Award is generally recognized as the highest distinction in computer science and the "Nobel Prize of computing".[3][4][5][6]

The award is named after Alan Turing, a British mathematician and reader in mathematics at the University of Manchester. Turing is often credited as being the key founder of theoretical computer science and artificial intelligence.[7] From 2007 to 2013, the award was accompanied by an additional prize of US $250,000, with financial support provided by Intel and Google.[2] Since 2014, the award has been accompanied by a prize of US $1 million, with financial support provided by Google.[1][8]

The first recipient, in 1966, was Alan Perlis, of Carnegie Mellon University. The first female recipient was Frances E. Allen of IBM in 2006.[9]

Contents [hide]

Recipients [edit]

Year	Recipient	Picture	Rationale
1966	Alan J. Perlis		For his influence in the area of advanced computer programming techniques and compiler construction.[10]
1967	Maurice Wilkes		Professor Wilkes is best known as the builder and designer of the EDSAC, the first computer with an internally stored program. Built in 1949, the EDSAC used a mercury delay line memory. He is also known as the author, with Wheeler and Gill, of a volume on "Preparation of Programs for Electronic Digital Computers" in 1951, in which program libraries were effectively introduced.[11]
1968	Richard Hamming		For his work on numerical methods, automatic coding systems, and error-detecting and error-correcting codes.[12]
1969	Marvin Minsky		For his central role in creating, shaping, promoting, and advancing the field of artificial intelligence.[13]
1970	James H. Wilkinson		For his research in numerical analysis to facilitate the use of the high-speed digital computer, having received special recognition for his work in computations in linear algebra and "backward" error analysis.[14]

ACM Turing Award

Stephen Kettle's slate statue of Alan Turing at Bletchley Park

Awarded for	Outstanding contributions in computer science
Country	United States
Presented by	Association for Computing Machinery (ACM)
Reward(s)	US $1,000,000[1]
First awarded	1966; 53 years ago
Last awarded	2017
Website	amturing.acm.org

The Turing Machine

- The Turing machine is essentially a finite automaton that has a single tape of infinite length on which it may read and write data.
- What can be computed can be represented using a simple notation much like the ID's of a PDA.

Notation for the Turing Machine

We may visualize a Turing machine as in figure.

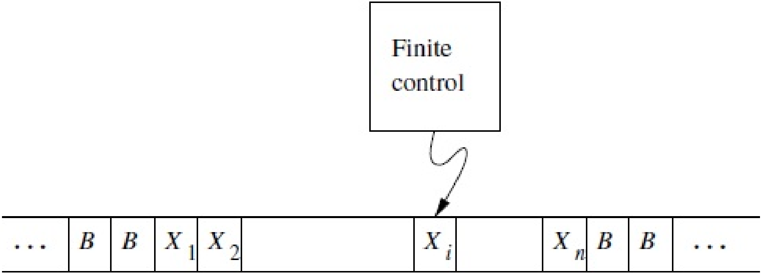
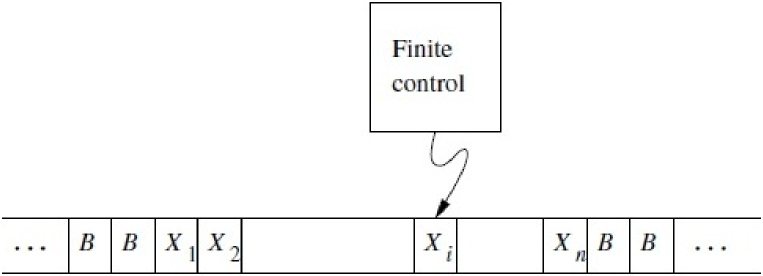


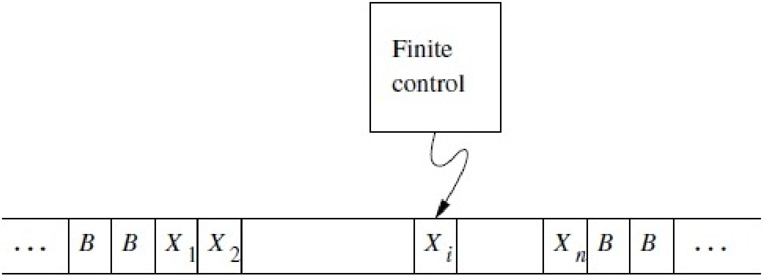
Figure 8.8: A Turing machine

Notation for the Turing Machine



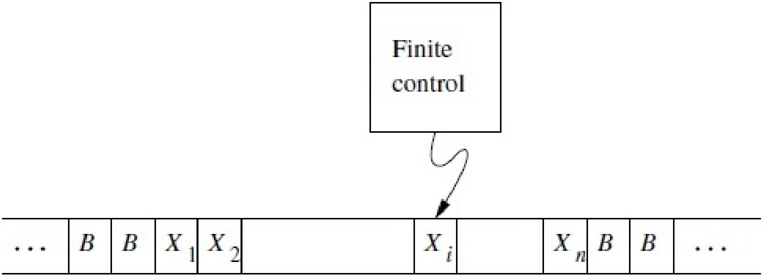
- The machine consists of a *finite control*, which can be in any of a finite set of states.

Notation for the Turing Machine



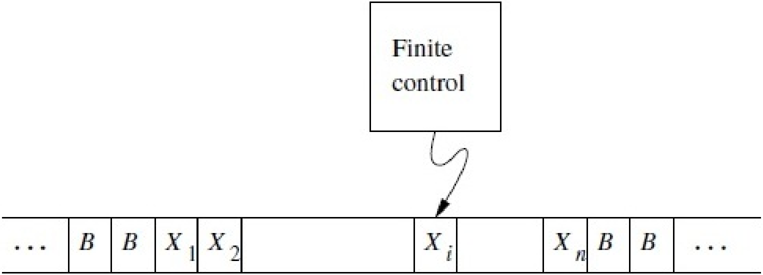
- There is a *tape* divided into squares or cells.
- Each cell can hold any one of a finite number of symbols.

Notation for the Turing Machine



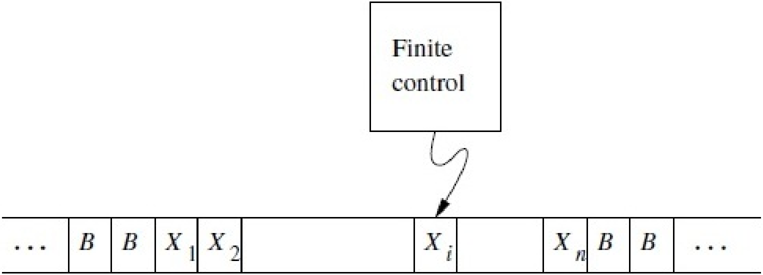
- Initially, the *input*, which is a finite-length string of symbols chosen from the *input alphabet*, is placed on the tape.

Notation for the Turing Machine



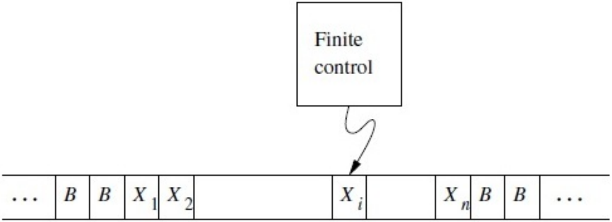
- All other tape cells, extending infinitely to the left and right, initially hold a special symbol called the *blank*.

Notation for the Turing Machine



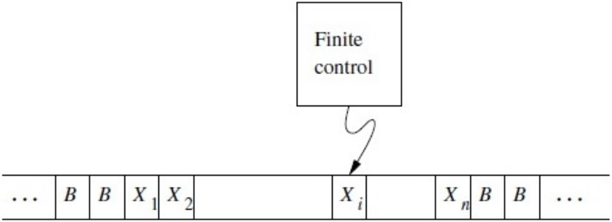
- The blank is a *tape symbol* but not an input symbol.
- There may be other tape symbols besides the input symbols and the blank, as well.

Notation for the Turing Machine



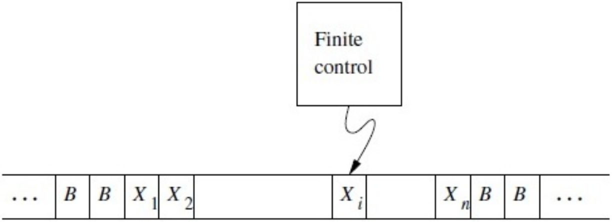
- There is a *tape head* that is always positioned at one of the tape cells.
- The Turing machine is said to be *scanning* that cell.
- Initially, the tape head is at the leftmost cell that holds the input.

Notation for the Turing Machine



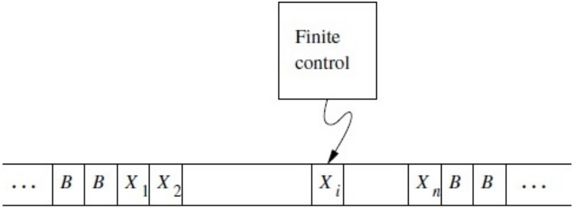
- There is a *tape head* that is always positioned at one of the tape cells.
- The Turing machine is said to be *scanning* that cell.
- Initially, the tape head is at the leftmost cell that holds the input.

Notation for the Turing Machine



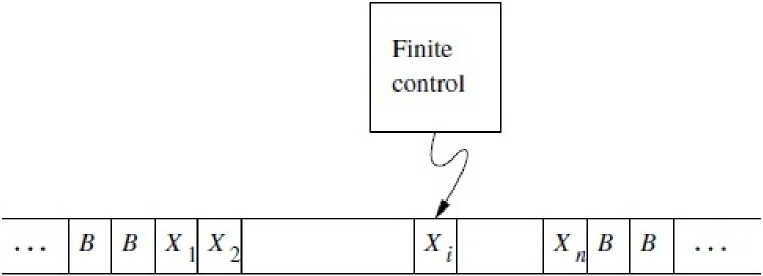
- There is a *tape head* that is always positioned at one of the tape cells.
- The Turing machine is said to be *scanning* that cell.
- Initially, the tape head is at the leftmost cell that holds the input.

Notation for the Turing Machine



- A *move* of the Turing machine is a function of the state of the finite control and the tape symbol scanned.

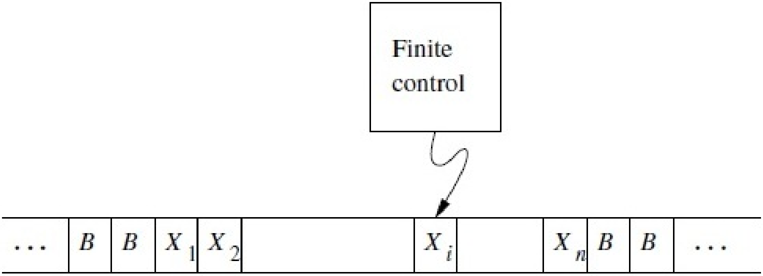
Notation for the Turing Machine



In one move, the Turing machine will:

- Change state.
  - The next state optionally may be the same as the current state.

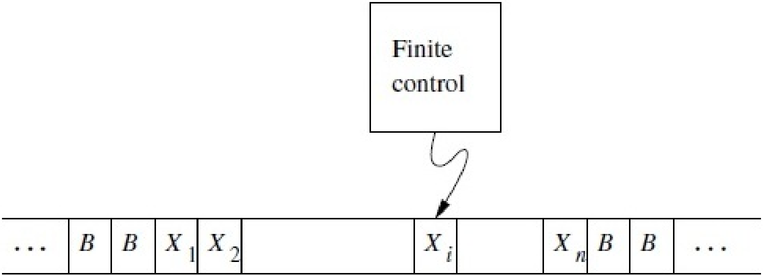
Notation for the Turing Machine



In one move, the Turing machine will:

- Write a tape symbol in the cell scanned.
  - This tape symbol replaces whatever symbol was in that cell.
  - Optionally, the symbol written may be the same as the symbol currently there.

Notation for the Turing Machine



In one move, the Turing machine will:

- Move the tape head left or right.
  - In our formalism we require a move, and do not allow the head to remain stationary.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

Q: The finite set of *states* of the finite control.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

Σ: The finite set of *input symbols*.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

Γ: The complete set of *tape symbols*.

- Σ is always a subset of Γ.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

δ : The *transition function.*

- The arguments of $\delta\,(q, X)$ are a state q and a tape symbol X.
- The value of $\delta\,(q, X)$, if it is defined, is a triple (p, Y, D), where:
  1. p is the next state, in Q.
  2. Y is the symbol, in Γ, written in the cell being scanned, replacing whatever symbol was there.
  3. D is a *direction*, either L or R, standing for "left" or "right," respectively, and telling us the direction in which the head moves.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

q_0: The *start state*, a member of Q, in which the finite control is found initially.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

B: The *blank* symbol.

- This symbol is in Γ but not in Σ; i.e., it is *not* an input symbol.
- The blank appears initially in all but the finite number of initial cells that hold input symbols.

Notation for the Turing Machine

- The formal notation we shall use for a Turing machine (TM) is similar to that used for finite automata or PDA's.
- We describe a TM by the 7-tuple

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

whose components have the following meanings:

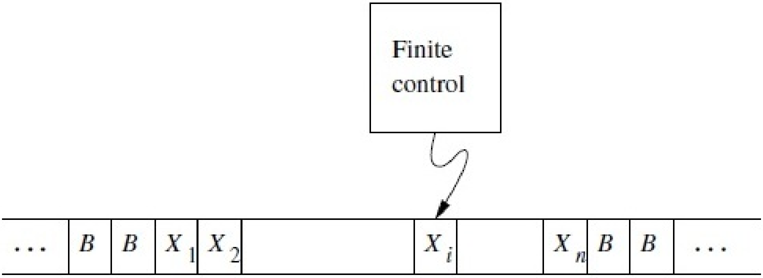
F: The set of *final* or *accepting* states, a subset of Q.

Instantaneous Descriptions for Turing Machines

- In order to describe formally what a Turing machine does, we need to develop a notation for configurations or *instantaneous descriptions* (ID's).
- These are like the notation we developed for PDA's.
- Since a TM, in principle, has an infinitely long tape, we might imagine that it is impossible to describe the configurations of a TM succinctly.
- However, after any finite number of moves, the TM can have visited only a finite number of cells.
- The number of cells visited can eventually grow beyond any finite limit.

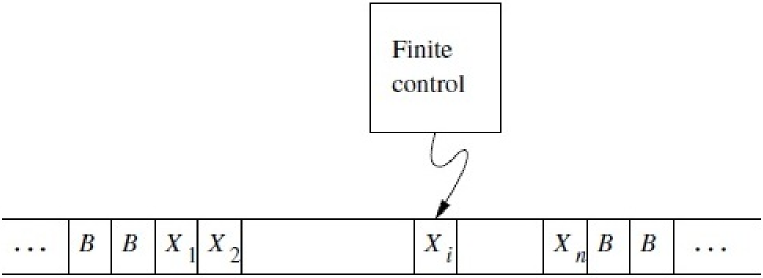
Instantaneous Descriptions for Turing Machines

- In order to describe formally what a Turing machine does, we need to develop a notation for configurations or *instantaneous descriptions* (ID's).
- These are like the notation we developed for PDA's.
- Since a TM, in principle, has an infinitely long tape, we might imagine that it is impossible to describe the configurations of a TM succinctly.
- However, after any finite number of moves, the TM can have visited only a finite number of cells.
- The number of cells visited can eventually grow beyond any finite limit.



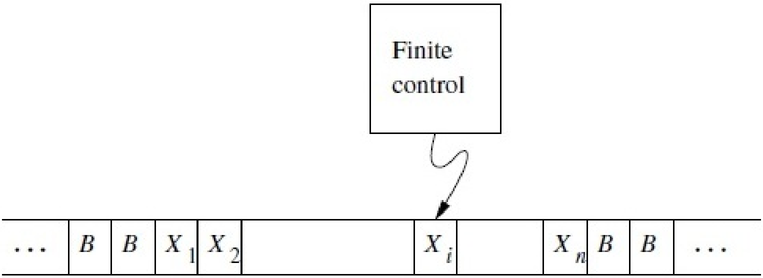
- Thus, in every ID there is an infinite prefix and an infinite suffix of cells that have never been visited.
- These cells must all hold either blanks or one of the finite number of input symbols.
- We thus show in an ID only the cells between the leftmost and the rightmost nonblanks.

Instantaneous . . . Turing Machines



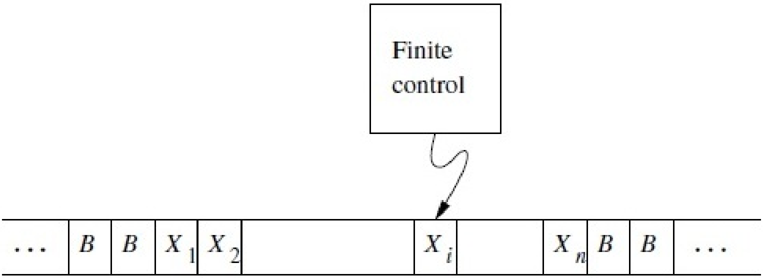
- Under special conditions, the head may be scanning one of the leading or trailing blanks.
- In this case a finite number of blanks to the left or right of the nonblank portion of the tape must also be included in the ID.

Instantaneous . . . Turing Machines



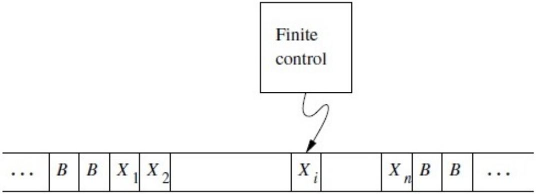
- In addition to representing the tape, we must represent the finite control and the tape-head position.
- To do so, we embed the state in the tape, and place it immediately to the left of the cell scanned.

Instantaneous . . . Turing Machines



- To disambiguate the tape-plus-state string, we have to make sure that we do not use as a state any symbol that is also a tape symbol.
- However, it is easy to change the names of the states so they have nothing in common with the tape symbols.
- The operation of the TM does not depend on what the states are called.

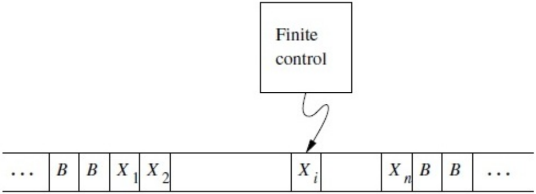
Instantaneous . . . Turing Machines



Thus, we shall use the string $X_1X_2 \ldots X_{i-1}qX_iX_{i+i} \ldots X_n$ to represent an ID in which

1. q is the state of the Turing machine.
2. The tape head is scanning the ith symbol from the left.
3. $X_1X_2 \ldots X_n$ is the portion of the tape between the leftmost and the rightmost nonblank.
   - As an exception, if the head is to the left of the leftmost nonblank or to the right of the rightmost nonblank, then some prefix or suffix of $X_1X_2 \ldots X_n$ will be blank.
   - And i will be 1 or n, respectively.

Instantaneous . . . Turing Machines



Thus, we shall use the string $X_1X_2 \dots X_{i-1}qX_iX_{i+i} \dots X_n$ to represent an ID in which

1. q is the state of the Turing machine.
2. The tape head is scanning the ith symbol from the left.
3. $X_1X_2 \dots X_n$ is the portion of the tape between the leftmost and the rightmost nonblank.
    - As an exception, if the head is to the left of the leftmost nonblank or to the right of the rightmost nonblank, then some prefix or suffix of $X_1X_2 \dots X_n$ will be blank.
    - And i will be 1 or n, respectively.

Instantaneous . . . Turing Machines

- We describe moves of a Turing machine $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$by the $\vdash_M$ notation that was used for PDA's.
- When the TM M is understood, we shall use just $\vdash$ to reflect moves.
- As usual, $\vdash_M^*$ or just $\vdash^*$ will be used to indicate zero, one, or more moves of the TM M.

Instantaneous . . . Turing Machines

Before:

X_1	X_2	$\ldots$	X_{i-2}	X_{i-1}	X_i	X_{i+1}	$\ldots$	X_n

After:

X_1	X_2	$\ldots$	X_{i-2}	X_{i-1}	Y	X_{i+1}	$\ldots$	X_n

- Suppose $\delta\,(q, X_i) = (p, Y, L)$; i.e., the next move is *leftward*.
- Then

$$X_1 X_2 \ldots X_{i-1} q X_i X_{i+1} \ldots X_n \vdash_M X_1 X_2 \ldots X_{i-2} p X_{i-1} Y X_{i+1} \ldots X_n$$

- Notice how this move reflects the change to state p and the fact that the tape head is now positioned at cell $i - 1$.

There are two important exceptions:

1. If $i = 1$, then M moves to the blank to the left of X_1.
   - In that case,

$$qX_1X_2\ldots X_n \vdash_M pBYX_2\ldots X_n$$

Before:

B	X_1	X_2	$\ldots$	X_n

After:

B	Y	X_2	$\ldots$	X_n

There are two important exceptions:

2. If $i = n$ and $Y = B$, then the symbol B written over X_n joins the infinite sequence of trailing blanks and does not appear in the next ID.
   - Thus,

$$X_1X_2 \dots X_{n-1}qX_n \vdash_M X_1X_2 \dots X_{n-2}pX_{n-1}$$

Before:

X_1	X_2	$\dots$	X_{n-2}	X_{n-1}	X_n

After:

X_1	X_2	$\dots$	X_{n-2}	X_{n-1}	B

Instantaneous . . . Turing Machines

Before:

X_1	X_2	$\dots$	X_{i-2}	X_{i-1}	X_i	X_{i+1}	$\dots$	X_n

After:

X_1	X_2	$\dots$	X_{i-2}	X_{i-1}	Y	X_{i+1}	$\dots$	X_n

- Now, suppose $\delta\,(q, X_i) = (p, Y, R)$; i.e., the next move is *rightward*.
- Then
$$\begin{aligned} &X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n \vdash_M \\ &\quad X_1 X_2 \dots X_{i-2} X_{i-1} Y p X_{i+1} \dots X_n \end{aligned}$$
- Here, the move reflects the fact that the head has moved to cell $i + 1$.

Again there are two important exceptions:

1. If $i = n$, then the $i + 1$st cell holds a blank, and that cell was not part of the previous ID.
   - Thus, we instead have

$$X_1X_2 \ldots X_{n-1}qX_n \vdash_M X_1X_2 \ldots X_{n-1}YpB$$

Before:

X_1	X_2	$\ldots$	X_{n-1}	X_n	B

After:

X_1	X_2	$\ldots$	X_{n-1}	Y	B

Again there are two important exceptions:

2. If $i = 1$ and $Y = B$, then the symbol B written over X_1 joins the infinite sequence of leading blanks and does not appear in the next ID.
    - Thus,

$$qX_1X_2 \ldots X_n \vdash_M pX_2 \ldots X_n$$

Before:

X_1	X_2	$\ldots$	X_n

After:

B	X_2	$\ldots$	X_n

Example

- Let us design a Turing machine and see how it behaves on a typical input.
- The TM we construct will accept the language $\{0^n 1^n \mid n \geq 1\}$.
- Initially, it is given a finite sequence of 0's and 1's on its tape, preceded and followed by an infinity of blanks.
- Alternately, the TM will change a 0 to an X and then a 1 to a Y, until all 0's and 1's have been matched.

Example-continued

- In more detail, starting at the left end of the input,
  - it repeatedly changes a 0 to an X and
  - moves to the right over whatever 0's and Y's it sees,
  - until it comes to a 1.
- It changes the 1 to a Y, and moves left, over Y's and 0's, until it finds an X.
- At that point, it looks for a 0 immediately to the right, and if it finds one, changes it to X and repeats the process, changing a matching 1 to a Y.

X	X	0	0	Y	1	1	1

Example-continued

- If the nonblank input is not in $\mathbf{0}^*\mathbf{1}^*$, then the TM will eventually fail to have a next move and will die without accepting.

X	X	0	0	Y	0	1	1

- However, if it finishes changing all the 0's to X's on the same round it changes the last 1 to a Y, then it has found its input to be of the form $0^n 1^n$ and accepts.

X	X	X	X	Y	Y	Y	Y

Example-continued

0	0	0	0	1	1	1	1

State	0	1	X	Y	B

Example-continued

The formal specification of the TM M is

$$M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0, 1, X, Y, B\}, \delta, q_0, B, \{q_4\})$$

where δ is given by the table.

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

Figure 8.9: A Turing machine to accept $\{0^n1^n \mid n \geq 1\}$

X	X	0	0	Y	1	1	1

- As M performs its computation, the portion of the tape, where M's tape head has visited, will always be a sequence of symbols described by the regular expression $\mathbf{X}^*\mathbf{0}^*\mathbf{Y}^*\mathbf{1}^*$.
- That is, there will be some 0's that have been changed to X's, followed by some 0's that have not yet been changed to X's.
- Then there are some 1's that were changed to Y's, and 1's that have not yet been changed to Y's.
- There may or may not be some 0's and 1's following.

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- State q_0 is the initial state, and M also enters state q_0 every time it returns to the leftmost remaining 0.
- If M is in state q_0 and scanning a 0, the rule in the upper-left corner tells it to go to state q_1, change the 0 to an X, and move right.

X	X	0	0	Y	1	1	1

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	$-$	$-$	(q_3, Y, R)	$-$
q_1	$(q_1, 0, R)$	(q_2, Y, L)	$-$	(q_1, Y, R)	$-$
q_2	$(q_2, 0, L)$	$-$	(q_0, X, R)	(q_2, Y, L)	$-$
q_3	$-$	$-$	$-$	(q_3, Y, R)	(q_4, B, R)
q_4	$-$	$-$	$-$	$-$	$-$

- Once in state q_1, M keeps moving right over all 0's and Y's that it finds on the tape, remaining in state q_1.
- If M sees an X or a B, it dies.
- However, if M sees a 1 when in state q_1, it changes that 1 to a Y, enters state q_2, and starts moving left.

X	X	0	0	Y	1	1	1

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- In state q_2, M moves left over 0's and Y's, remaining in state q_2.
- When M reaches the rightmost X, which marks the right end of the block of 0's that have already been changed to X, M returns to state q_0 and moves right.

X	X	0	0	Y	Y	1	1

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

There are two cases:

1. If M now sees a 0, then it repeats the matching cycle we have just described.

X	X	0	0	Y	Y	1	1

State	Symbol				
	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

There are two cases:

2. If M sees a Y, then it has changed all the 0's to X's.
   - If all the 1's have been changed to Y's, then the input was of the form 0^n1^n, and M should accept.
   - Thus, M enters state q_3, and starts moving right, over Y's.
   - If the first symbol other than a Y that M sees is a blank, then indeed there were an equal number of 0's and 1's, so M enters state q_4 and accepts.

X	X	X	X	Y	Y	Y	Y	B

State	0	1	Symbol X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

There are two cases:

2. *(continued)*
    - On the other hand, if M encounters another 1, then there are too many 1's, so M dies without accepting.

X	X	X	X	Y	Y	Y	Y	1

State	Symbol				
	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

There are two cases:

2. *(continued)*
    - If it encounters a 0, then the input was of the wrong form, and *M* also dies.

X	X	X	X	Y	Y	Y	Y	0

$q_0 0011$

$\vdash Xq_1 011$

$\vdash X0q_1 11$

$\vdash Xq_2 0Y1$

$\vdash q_2 X0Y1$

$\vdash Xq_0 0Y1$

$\vdash XXq_1 Y1$

$\vdash XXYq_1 1$

$\vdash XXq_2 YY$

$\vdash Xq_2 XYY$

$\vdash XXq_0 YY$

$\vdash XXYq_3 Y$

$\vdash XXYYq_3 B$

$\vdash XXYYBq_4 B$

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- Here is an example of an accepting computation by M.
- Its input is 0011.
- Initially, M is in state q_0, scanning the first 0, i.e., M's initial ID is $q_0 0011$.

$q_0 0011$

$\vdash Xq_1 011$

$\vdash X0q_1 11$

$\vdash Xq_2 0Y1$

$\vdash q_2 X0Y1$

$\vdash Xq_0 0Y1$

$\vdash XXq_1 Y1$

$\vdash XXYq_1 1$

$\vdash XXq_2 YY$

$\vdash Xq_2 XYY$

$\vdash XXq_0 YY$

$\vdash XXYq_3 Y$

$\vdash XXYYq_3 B$

$\vdash XXYYBq_4 B$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	$-$	$-$	(q_3, Y, R)	$-$
q_1	$(q_1, 0, R)$	(q_2, Y, L)	$-$	(q_1, Y, R)	$-$
q_2	$(q_2, 0, L)$	$-$	(q_0, X, R)	(q_2, Y, L)	$-$
q_3	$-$	$-$	$-$	(q_3, Y, R)	(q_4, B, R)
q_4	$-$	$-$	$-$	$-$	$-$

- M finds the first 0, changes to it X and moves to the right.
- M also switches to q_1.

$q_0 0011$

$\vdash X q_1 011$

$\vdash X0 q_1 11$

$\vdash X q_2 0Y1$

$\vdash q_2 X0Y1$

$\vdash X q_0 0Y1$

$\vdash XX q_1 Y1$

$\vdash XXY q_1 1$

$\vdash XX q_2 YY$

$\vdash X q_2 XYY$

$\vdash XX q_0 YY$

$\vdash XXY q_3 Y$

$\vdash XXYY q_3 B$

$\vdash XXYYB q_4 B$

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M moves to the right over whatever 0's and Y's it sees, until it comes to a 1 and changes the 1 to a Y and moves left.
- M also switches to q_2.

$q_0 0011$

$\vdash X q_1 011$

$\vdash X 0 q_1 11$

$\vdash X q_2 0 Y 1$

$\vdash q_2 X 0 Y 1$

$\vdash X q_0 0 Y 1$

$\vdash X X q_1 Y 1$

$\vdash X X Y q_1 1$

$\vdash X X q_2 Y Y$

$\vdash X q_2 X Y Y$

$\vdash X X q_0 Y Y$

$\vdash X X Y q_3 Y$

$\vdash X X Y Y q_3 B$

$\vdash X X Y Y B q_4 B$

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M moves left, over Y's and 0's, until it finds an X.
- This rightmost X marks the right end of the block of 0's that have already been changed to X.
- M returns to state q_0 and moves right.

$q_0 0011$
$\vdash Xq_1 011$
$\vdash X0q_1 11$
$\vdash Xq_2 0Y1$
$\vdash q_2 X0Y1$
$\vdash Xq_0 0Y1$
$\vdash XXq_1 Y1$
$\vdash XXYq_1 1$
$\vdash XXq_2 YY$
$\vdash Xq_2 XYY$
$\vdash XXq_0 YY$
$\vdash XXYq_3 Y$
$\vdash XXYYq_3 B$
$\vdash XXYYBq_4 B$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M is on the first 0 yet to be changed.
- M changes to it X and moves to the right.
- M also switches to q_1.

$q_0 0011$
$\vdash Xq_1 011$
$\vdash X0q_1 11$
$\vdash Xq_2 0Y1$
$\vdash q_2 X0Y1$
$\vdash Xq_0 0Y1$
$\vdash XXq_1 Y1$
$\vdash XXYq_1 1$
$\vdash XXq_2 YY$
$\vdash Xq_2 XYY$
$\vdash XXq_0 YY$
$\vdash XXYq_3 Y$
$\vdash XXYYq_3 B$
$\vdash XXYYBq_4 B$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M moves to the right over whatever 0's and Y's it sees, until it comes to a 1 and changes the 1 to a Y and moves left.
- M also switches to q_2.

$q_0 0011$
$\vdash X q_1 011$
$\vdash X 0 q_1 11$
$\vdash X q_2 0 Y 1$
$\vdash q_2 X 0 Y 1$
$\vdash X q_0 0 Y 1$
$\vdash X X q_1 Y 1$
$\vdash X X Y q_1 1$
$\vdash X X q_2 Y Y$
$\vdash X q_2 X Y Y$
$\vdash X X q_0 Y Y$
$\vdash X X Y q_3 Y$
$\vdash X X Y Y q_3 B$
$\vdash X X Y Y B q_4 B$

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M moves left, over Y's and 0's, until it finds an X.
- This rightmost X marks the right end of the block of 0's that have already been changed to X.
- M returns to state q_0 and moves right.

$q_0 0011$
$\vdash X q_1 011$
$\vdash X0 q_1 11$
$\vdash X q_2 0Y1$
$\vdash q_2 X0Y1$
$\vdash X q_0 0Y1$
$\vdash XX q_1 Y1$
$\vdash XXY q_1 1$
$\vdash XX q_2 YY$
$\vdash X q_2 XYY$
$\vdash XX q_0 YY$
$\vdash XXY q_3 Y$
$\vdash XXYY q_3 B$
$\vdash XXYYB q_4 B$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- M sees a Y, that means it has changed all the 0's to X's.
- If all the 1's have been changed to Y's, then the input was of the form 0^n1^n, and M should accept.
- Thus, M enters state q_3, and starts moving right, over Y's.

$q_0 0011$

$\vdash Xq_1 011$

$\vdash X0q_1 11$

$\vdash Xq_2 0Y1$

$\vdash q_2 X0Y1$

$\vdash Xq_0 0Y1$

$\vdash XXq_1 Y1$

$\vdash XXYq_1 1$

$\vdash XXq_2 YY$

$\vdash Xq_2 XYY$

$\vdash XXq_0 YY$

$\vdash XXYq_3 Y$

$\vdash XXYYq_3 B$

$\vdash XXYYBq_4 B$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- The first symbol other than a Y that M sees is a blank, then indeed there were an equal number of 0's and 1's, so M enters state q_4 and accepts.

(q_4 is the accepting state)

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- For another example, consider what M does on the input 0010, which is not in the language.

  $$q_0 0010 \vdash Xq_1 010 \vdash X0q_1 10 \vdash Xq_2 0Y0$$
  $$\vdash q_2 X0Y0 \vdash Xq_0 0Y0 \vdash XXq_1 Y0 \vdash XXYq_1 0 \vdash XXY0q_1 B$$

- The behavior of M on 0010 resembles the behavior on 0011, until in ID $XXYq_1 0$ M scans the final 0 for the first time.
- M must move right, staying in state q_1, which takes it to the ID $XXY0q_1 B$.
- However, in state q_1 M has no move on tape symbol B.
- Thus M dies and does not accept its input.

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

- For another example, consider what M does on the input 0010, which is not in the language.

$$q_0 0010 \vdash Xq_1 010 \vdash X0q_1 10 \vdash Xq_2 0Y0$$
$$\vdash q_2 X0Y0 \vdash Xq_0 0Y0 \vdash XXq_1 Y0 \vdash XXYq_1 0 \vdash XXY0q_1 B$$

- The behavior of M on 0010 resembles the behavior on 0011, until in ID $XXYq_1 0$ M scans the final 0 for the first time.
- M must move right, staying in state q_1, which takes it to the ID $XXY0q_1 B$.
- However, in state q_1 M has no move on tape symbol B.
- Thus M dies and does not accept its input.

Transition Diagrams for Turing Machines

- We can represent the transitions of a Turing machine pictorially.
- A transition diagram consists of a set of nodes corresponding to the states of the TM.
- An arc from state q to state p is labeled by one or more items of the form X/YD, where X and Y are tape symbols, and D is a direction, either L or R.
- That is, whenever $\delta(q, X) = (p, Y, D)$, we find the label X/YD on the arc from q to p.
- However, in our diagrams, the direction D is represented pictorially by $\leftarrow$ for "left" and $\rightarrow$ for "right."

Transition Diagrams for Turing Machines

- As for other kinds of transition diagrams, we represent the start state by the word “Start” and an arrow entering that state.
- Accepting states are indicated by double circles.
- Thus, the only information about the TM one cannot read directly from the diagram is the symbol used for the blank.
- We shall assume that symbol is B unless we state otherwise.

Example

Figure shows the transition diagram for the Turing machine of previous example.

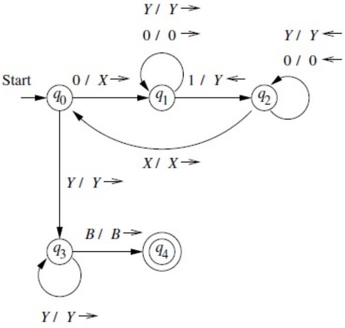


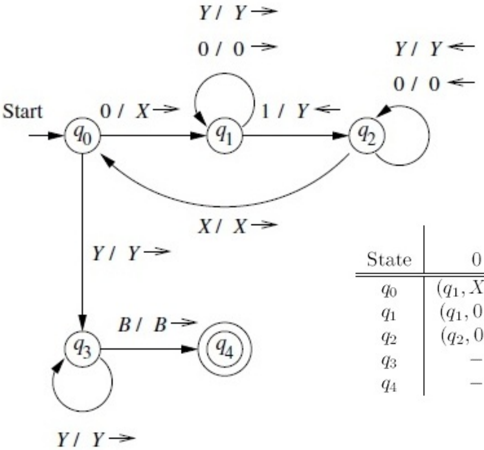
Figure 8.10: Transition diagram for a TM that accepts strings of the form 0^n1^n

Example

Figure shows the transition diagram for the Turing machine of previous example.

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	—	—	(q_3, Y, R)	—
q_1	$(q_1, 0, R)$	(q_2, Y, L)	—	(q_1, Y, R)	—
q_2	$(q_2, 0, L)$	—	(q_0, X, R)	(q_2, Y, L)	—
q_3	—	—	—	(q_3, Y, R)	(q_4, B, R)
q_4	—	—	—	—	—

- A transition diagram consists of a set of nodes corresponding to the states of the TM.
- An arc from state q to state p is labeled by one or more items of the form X/YD, where X and Y are tape symbols, and D is a direction, either L or R. That is, whenever $\delta(q, X) = (p, Y, D)$, we find the label X/YD on the arc from q to p.
- The direction D is represented pictorially by $\leftarrow$ for "left" and $\rightarrow$ for "right."
- We represent the start state by the word "Start" and an arrow entering that state.
- Accepting states are indicated by double circles.



	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–

TM as a computer of integer-valued functions

- Today we find it most convenient to think of Turing machines as recognizers of languages, or equivalently, solvers of problems.
- Turing's original view of his machine was as a computer of integer-valued functions.
- In his scheme, integers were represented in unary, as blocks of a single character.
- The machine computed by changing the lengths of the blocks or by constructing new blocks elsewhere on the tape.

Example

- In this simple example, we shall show how a Turing machine might compute the function $\dot{-}$, which is called monus or proper subtraction.
- This is defined by $m \dot{-} n = \max(m - n, 0)$.
- That is, $m \dot{-} n$ is $m - n$ if $m \geq n$ and 0 if $m < n$.
- A TM that performs this operation is specified by

$$M = (\{q_0, q_1, \ldots, q_6\}, \{0,1\}, \{0,1,B\}, \delta, q_0, B)$$

- Since this TM is not used to accept inputs, we have omitted the seventh component, which is the set of accepting states.
- M will start with a tape consisting of $0^m 1 0^n$ surrounded by blanks.
- M halts with $0^{m \dot{-} n}$ on its tape, surrounded by blanks.

Example

- In this simple example, we shall show how a Turing machine might compute the function $\dot{-}$, which is called monus or proper subtraction.
- This is defined by $m \dot{-} n = \max(m - n, 0)$.
- That is, $m \dot{-} n$ is $m - n$ if $m \geq n$ and 0 if $m < n$.
- A TM that performs this operation is specified by

$$M = (\{q_0, q_1, \ldots, q_6\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B)$$

- Since this TM is not used to accept inputs, we have omitted the seventh component, which is the set of accepting states.
- M will start with a tape consisting of 0^m10^n surrounded by blanks.
- M halts with $0^{m \dot{-} n}$ on its tape, surrounded by blanks.

Example

- In this simple example, we shall show how a Turing machine might compute the function $\dot{-}$, which is called monus or proper subtraction.
- This is defined by $m \dot{-} n = \max(m - n, 0)$.
- That is, $m \dot{-} n$ is $m - n$ if $m \geq n$ and 0 if $m < n$.
- A TM that performs this operation is specified by

$$M = (\{q_0, q_1, \ldots, q_6\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B)$$

- Since this TM is not used to accept inputs, we have omitted the seventh component, which is the set of accepting states.
- M will start with a tape consisting of 0^m10^n surrounded by blanks.
- M halts with $0^{m \dot{-} n}$ on its tape, surrounded by blanks.

Example-continued

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

0	0	0	1	0	0

$\overset{\dots}{\Rightarrow}$

B	0	0	1	1	0

......

B	B	B	1	1	1

$\overset{\dots}{\Rightarrow}$

B	B	0	B	B	B

Example-continued

The repetition ends if either:

1. Searching right for a 0, M encounters a blank.
    - Then the n 0's in 0^m10^n have all been changed to 1's.
    - And $n+1$ of the m 0's have been changed to B.
    - M replaces the $n+1$ 1's by one 0 and n B's, leaving $m-n$ 0's on the tape.
    - Since $m \geq n$ in this case, $m-n = m \dot{-} n$.
2. Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.
    - Then $n \geq m$, so $m \dot{-} n = 0$.
    - M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

Example-continued

Figure gives the rules of the transition function δ .

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	$-$
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	$-$
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	$-$	$-$	$-$

A Turing machine that computes the proper-subtraction

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\ldots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\ldots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\ldots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\ldots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\ldots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

1-a

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

2,3-b

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

2,3-b

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

4-c

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

5-d

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

6-e

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

7,8-f

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 1 0^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

7,8-f

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

9-g

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

10,11-h

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

10,11-h

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 1 0^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

12-i

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

13-j

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- *M* repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- *M* then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

14-k

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

The repetition ends if, searching right for a 0, M encounters a blank.

- Then the n 0's in $0^m 10^n$ have all been changed to 1's.
- And $n + 1$ of the m 0's have been changed to B.
- M replaces the $n + 1$ 1's by one 0 and n B's, leaving $m - n$ 0's on the tape.
- Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

15,16,17-I

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

The repetition ends if, searching right for a 0, M encounters a blank.

- Then the n 0's in $0^m 10^n$ have all been changed to 1's.
- And $n + 1$ of the m 0's have been changed to B.
- M replaces the $n + 1$ 1's by one 0 and n B's, leaving $m - n$ 0's on the tape.
- Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

15,16,17-I

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\ldots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\ldots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\ldots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\ldots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\ldots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

The repetition ends if, searching right for a 0, M encounters a blank.

- Then the n 0's in $0^m 10^n$ have all been changed to 1's.
- And $n + 1$ of the m 0's have been changed to B.
- M replaces the $n + 1$ 1's by one 0 and n B's, leaving $m - n$ 0's on the tape.
- Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

15,16,17-I

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

The repetition ends if, searching right for a 0, M encounters a blank.

- Then the n 0's in $0^m 10^n$ have all been changed to 1's.
- And $n + 1$ of the m 0's have been changed to B.
- M replaces the $n + 1$ 1's by one 0 and n B's, leaving $m - n$ 0's on the tape.
- Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

18-m

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 1 0^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

The repetition ends if, searching right for a 0, M encounters a blank.

- Then the n 0's in $0^m 1 0^n$ have all been changed to 1's.
- And $n+1$ of the m 0's have been changed to B.
- M replaces the $n+1$ 1's by one 0 and n B's, leaving $m-n$ 0's on the tape.
- Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\cdots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\cdots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\cdots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\cdots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\cdots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

$0^4 10^2$, $m = 4$, $n = 2$

Strictly speaking, extra B's at ends are not to be shown.

- The machine halts.
- Or it would initiate next step.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

q_0: This state begins the cycle.

- It also breaks the cycle when appropriate.
- If M is scanning a 0, the cycle must repeat.
- The 0 is replaced by B, the head moves right, and state q_1 is entered.
- On the other hand, if M is scanning 1, then all possible matches between the two groups of 0's on the tape have been made, and M goes to state q_5 to make the tape blank.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\ldots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\ldots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\ldots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\ldots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\ldots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

q_1: In this state, M searches right, through the initial block of 0's, looking for the leftmost 1.

- When found, M goes to state q_2.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\ldots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\ldots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\ldots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\ldots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\ldots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

1. Searching right for a 0, M encounters a blank.
   - Then the n 0's in 0^m10^n have all been changed to 1's.
   - And $n+1$ of the m 0's have been changed to B.
   - M replaces the $n+1$ 1's by one 0 and n B's, leaving $m-n$ 0's on the tape.
   - Since $m \geq n$ in this case, $m - n = m \dot{-} n$.

q_2: M moves right, skipping over 1's, until it finds a 0.

- It changes that 0 to a 1, turns leftward, and enters state q_3.
- However, it is also possible that there are no more 0's left after the block of 1's, in that case, M in state q_2 encounters a blank.
- We have case (1) described above, where n 0's in the second block of 0's have been used to cancel n of the m 0's in the first block, and the subtraction is complete.
- M enters state q_4, whose purpose is to convert the 1's on the tape to blanks.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\ldots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\ldots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\ldots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\ldots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\ldots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

q_3: M moves left, skipping over 0's and 1's, until it finds a blank.

- When it finds B, it moves right and returns to state q_0, beginning the cycle again.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

q_4: Here, the subtraction is complete, but one unmatched 0 in the first block was incorrectly changed to a B.

- M therefore moves left, changing 1's to B's, until it encounters a B on the tape.
- It changes that B back to 0, and enters state q_6, wherein M halts.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
$\dots$
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
$\dots$
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
$\dots$
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
$\dots$
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
$\dots$
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

2. Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.
   - Then $n \geq m$, so $m \dot{-} n = 0$.
   - M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

q_5: State q_5 is entered from q_0 when it is found that all 0's in the first block have been changed to B.

- In this case, described in (2) above, the result of the proper subtraction is 0.
- M changes all remaining 0's and 1's to B and enters state q_6.

$q_0 0000100$
$\vdash Bq_1 000100$
$\vdash B0q_1 00100$
...
$\vdash B000q_1 100$
$\vdash B0001q_2 00$
$\vdash B000q_3 110$
$\vdash B00q_3 0110$
...
$\vdash q_3 B000110$
$\vdash Bq_0 000110$
...
$\vdash BBq_0 00111$
$\vdash BBBq_1 0111$
...
$\vdash BBB011q_2 1$
$\vdash BBB0111q_2 B$
$\vdash BBB011q_4 1B$
...
$\vdash BBBq_4 0BBB$
$\vdash BBBq_4 B0BB$
$\vdash BBB0q_6 0BB$

The repetition ends if either:

q_6: The sole purpose of this state is to allow M to halt when it has finished its task.

- If the subtraction had been a subroutine of some more complex function, then q_6 would initiate the next step of that larger computation.

Example-continued

We have represented δ as a transition diagram.

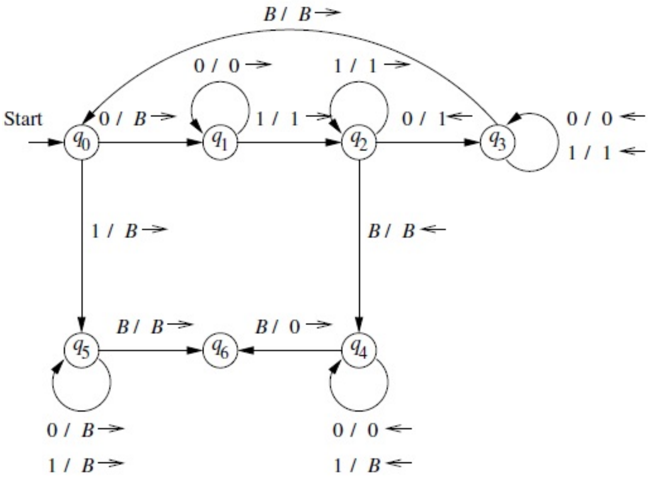
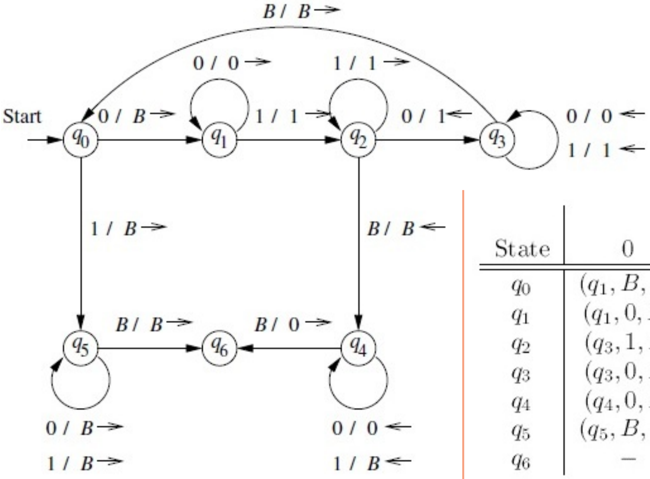


Figure 8.12: Transition diagram for the TM of Example 8.4

Example-continued



State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	–
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	–
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	–	–	–

$q_0 0010000$

$\cdots$

$\vdash Bq_0 011000$

$\cdots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 1 0^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

1-2-a

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\ldots$

$\vdash Bq_0 011000$

$\ldots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 10^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

1-2-a

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\ldots$

$\vdash Bq_0 011000$

$\ldots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 1 0^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

- M repeatedly finds its leftmost remaining 0 and replaces it by a blank.
- It then searches right, looking for a 1.
- After finding a 1, it continues right, until it comes to a 0, which it replaces by a 1.
- M then returns left, seeking the leftmost 0, which it identifies when it first meets a blank and then moves one cell to the right.

3-p

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\cdots$

$\vdash Bq_0 011000$

$\cdots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 10^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.

- Then $n \geq m$, so $m \dot{-} n = 0$.
- M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

4,5-q

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\ldots$

$\vdash Bq_0 011000$

$\ldots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 10^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.

- Then $n \geq m$, so $m \dot{-} n = 0$.
- M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

4,5-q

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\cdots$

$\vdash Bq_0 011000$

$\cdots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 1 0^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.

- Then $n \geq m$, so $m \dot{-} n = 0$.
- M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

6,7-r

	Symbol		
State	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\dots$

$\vdash Bq_0 011000$

$\dots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 1 0^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

Beginning the cycle, M cannot find a 0 to change to a blank, because the first m 0's already have been changed to B.

- Then $n \geq m$, so $m \dot{-} n = 0$.
- M replaces all remaining 1's and 0's by B and ends with a completely blank tape.

6,7-r

State	Symbol 0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\dots$

$\vdash Bq_0 011000$

$\dots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 1 0^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

- The machine halts.
- Or it would initiate next step.

8-s

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	—
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	—
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	—	—	—

$q_0 0010000$

$\ldots$

$\vdash Bq_0 011000$

$\ldots$

$\vdash BBq_0 11100$

$\vdash BBBq_5 1100$

$\vdash BBBBq_5 100$

$\vdash BBBBBq_5 00$

$\vdash BBBBBBq_5 0$

$\vdash BBBBBBBq_5 B$

$\vdash BBBBBBBq_6$

$0^2 10^4$, $m = 2$, $n = 4$

Strictly speaking, extra B's at ends are not to be shown.

The Language of a Turing Machine

- We have intuitively suggested the way that a Turing machine accepts a language.
- The input string is placed on the tape, and the tape head begins at the leftmost input symbol.
- If the TM eventually enters an accepting state, then the input is accepted, and otherwise not.

The Language of a Turing Machine

- More formally, let $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$be a Turing machine.
- Then $L(M)$ is the set of strings w in Σ^* such that $q_0 w \vdash^* \alpha p \beta$ for some state p in F and any tape strings α and β.
- This definition was assumed when we discussed the Turing machine which accepts strings of the form $0^n 1^n$.

The Language of a Turing Machine

- More formally, let $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$be a Turing machine.
- Then $L(M)$ is the set of strings w in Σ^* such that $q_0 w \vdash^* \alpha p \beta$ for some state p in F and any tape strings α and β.
- This definition was assumed when we discussed the Turing machine which accepts strings of the form $0^n 1^n$.

The Language of a Turing Machine

- The set of languages we can accept using a Turing machine is often called the *recursively enumerable languages* or RE languages.
- The term “recursively enumerable” comes from computational formalisms that predate the Turing machine.
- These formalisms define the same class of languages or arithmetic functions.

Notational Conventions for Turing Machines

The symbols we normally use for Turing machines resemble those for the other kinds of automata we have seen.

1. Lower-case letters at the beginning of the alphabet stand or input symbols.
2. Capital letters, typically near the end of the alphabet, are used for tape symbols that may or may not be input symbols.

   However, B is generally used for the blank symbol.
3. Lower-case letters near the end of the alphabet are strings of input symbols.
4. Greek letters are strings of tape symbols.
5. Letters such as q, p, and nearby letters are states.

Turing Machines and Halting

- There is another notion of “acceptance” that is commonly used for Turing machines: acceptance by halting.
- We say a TM *halts* if it enters a state q, scanning a tape symbol X, and there is no move in this situation.
- In this case $\delta\,(q, X)$ is undefined.

Example

- The Turing machine M for monus computation was not designed to accept a language.
- Rather we viewed it as computing an arithmetic function.
- In this machine the seventh component was omitted,

$$M = (\{q_0, q_1, \ldots, q_6\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B)$$

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

Example-continued

- However, M halts on all strings of 0's and 1's.
- No matter what string M finds on its tape, it will eventually cancel its second group of 0's.
- If it can find such a group, against its first group of 0's, and thus must reach state q_6 and halt.

State	Symbol 0	Symbol 1	Symbol B
q_0	(q_1, B, R)	(q_5, B, R)	$-$
q_1	$(q_1, 0, R)$	$(q_2, 1, R)$	$-$
q_2	$(q_3, 1, L)$	$(q_2, 1, R)$	(q_4, B, L)
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	$(q_4, 0, L)$	(q_4, B, L)	$(q_6, 0, R)$
q_5	(q_5, B, R)	(q_5, B, R)	(q_6, B, R)
q_6	$-$	$-$	$-$