# CSE211-Formal Languages and Automata Theory

## U3L6 – Turing Machine as Computer

**Dr. P. Saravanan**
School of Computing
SASTRA Deemed University

# Agenda

- Showing how a TM computes.
- Indicating that TM's are as powerful as conventional computers.
- Even some extended TM's can be simulated by the original TM

# Turing Machine as Computer

- TM's may be used as a computer as well, not just a language recognizer
- Design a TM to compute a function denoted by " $\dot{-}$ " called *monus*, or *proper subtraction* defined by

$$m \dot{-} n = m - n \qquad \text{if } m \geq n;$$

$$= 0 \qquad \text{if } m < n.$$

# Turing Machine as Computer

- A number is represented in binary format in different finite automata like 5 is represented as (101) but in case of addition using a Turing machine unary format is followed
- In unary format a number is represented by either all ones or all zeroes.
- For example, 5 will be represented by a sequence of five zeroes or five ones. 5 = 1 1 1 1 1 or 0 0 0 0 0.
- Lets use zeroes for representation.

# Subtraction by Turing Machine

- Assume input integers *m* and *n* are put on the input tape separated by a 1 as $0^m10^n$ *(two unary numbers using 0's separated by a special symbol 1).*
- The TM is $M = (\{q_0, q_1, \ldots, q_6\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B)$.
- *No final state is needed.*

# Subtraction by Turing Machine…

Steps:

- 1. find its leftmost 0 and replaces it by a blank;
- 2. move right, and look for a 1;
- 3. after finding a 1, move right continuously
- 4. after finding a 0, replace it by a 1;
- 5. move left until finding a blank, & then move one cell to the right to get a 0;
- 6. repeat the above process.

# Subtraction by Turing Machine…

- Example

2 - 1 = 1
Starting:
0010
Ending:
B0BB

- $q_0\underline{0}010$
- $\Rightarrow Bq_1\underline{0}10$
- $\Rightarrow B0q_1\underline{1}0$
- $\Rightarrow B01q_2\underline{0}$
- $\Rightarrow B0q_3\underline{1}1$
- $\Rightarrow Bq_3\underline{0}11$
- $\Rightarrow q_3\underline{B}011$
- $\Rightarrow Bq_0\underline{0}11$
- $\Rightarrow BBq_1\underline{1}1$
- $\Rightarrow BB1q_2\underline{1}$
- $\Rightarrow BB11q_2\underline{B}$
- $\Rightarrow BB1q_4\underline{1}$
- $\Rightarrow BBq_4\underline{1}B$
- $\Rightarrow Bq_4\underline{B}BB$
- $\Rightarrow B0q_6\underline{B}B$ halt! (with one 0 left, correct)

# Subtraction by Turing Machine…

- Example

3 - 1 = 2
Starting:
00010
Ending:
B00BB

- $q_0\underline{00}010$
- $\Rightarrow Bq_1\underline{00}10$
- $\Rightarrow B0q_1\underline{0}10$
- $\Rightarrow B00q_1\underline{1}0$
- $\Rightarrow B0001q_2\underline{0}$
- $\Rightarrow B00q_3\underline{1}1$
- $\Rightarrow B0q_3\underline{0}11$
- $\Rightarrow q_3\underline{B0}011$
- $\Rightarrow Bq_0\underline{00}11$
- $\Rightarrow BB0q_1\underline{1}1$
- $\Rightarrow BB01q_2\underline{1}$
- $\Rightarrow BB011q_2\underline{B}$
- $\Rightarrow BB01q_4\underline{1}$
- $\Rightarrow BB0q_4\underline{1}B$
- $\Rightarrow BB0q_4\underline{1}B$
- $\Rightarrow BBq_4\underline{0}BB$
- $\Rightarrow Bq_4\underline{B0}BB$
- $\Rightarrow B00q_6\underline{B}B$

halt!

# Subtraction by Turing Machine...

- 2 - 1 = 1:
- $q_0\underline{0}010$
- $\Rightarrow Bq_1\underline{0}10$
- $\Rightarrow B0q_1\underline{1}0$
- $\Rightarrow B01q_2\underline{0}$
- $\Rightarrow B0q_3\underline{1}1$
- $\Rightarrow Bq_3\underline{0}11$
- $\Rightarrow q_3\underline{B}011$
- $\Rightarrow Bq_0\underline{0}11$
- $\Rightarrow BBq_1\underline{1}1$
- $\Rightarrow BB1q_2\underline{1}$
- $\Rightarrow BB11q_2\underline{B}$
- $\Rightarrow BB1q_4\underline{1}$
- $\Rightarrow BBq_4\underline{1}B$
- $\Rightarrow Bq_4\underline{B}BB$
- $\Rightarrow B0q_6\underline{B}B$ halt! (with one 0 left, correct)

Transition table for subtraction

| | symbol | | |
|---|---|---|---|
| state | 0 | 1 | $B$ |
| $q_0$ | $(q_1, B, R)$ | $(q_5, B, R)$ | - |
| $q_1$ | $(q_1, 0, R)$ | $(q_2, 1, R)$ | - |
| $q_2$ | $(q_3, 1, L)$ | $(q_2, 1, R)$ | $(q_4, B, L)$ |
| $q_3$ | $(q_3, 0, L)$ | $(q_3, 1, L)$ | $(q_0, B, R)$ |
| $q_4$ | $(q_4, 0, L)$ | $(q_4, B, L)$ | $(q_6, 0, R)$ |
| $q_5$ | $(q_5, B, R)$ | $(q_5, B, R)$ | $(q_6, B, R)$ |
| $q_6$ | - | - | - |

# Turing Machine for Addition

- A number is represented in binary format in different finite automata like 5 is represented as (101) but in case of addition using a Turing machine unary format is followed
- In unary format a number is represented by either all ones or all zeroes.
- For example, 5 will be represented by a sequence of five zeroes or five ones. 5 = 1 1 1 1 1 or 0 0 0 0 0.
- Lets use zeroes for representation.

# Turing Machine for Addition

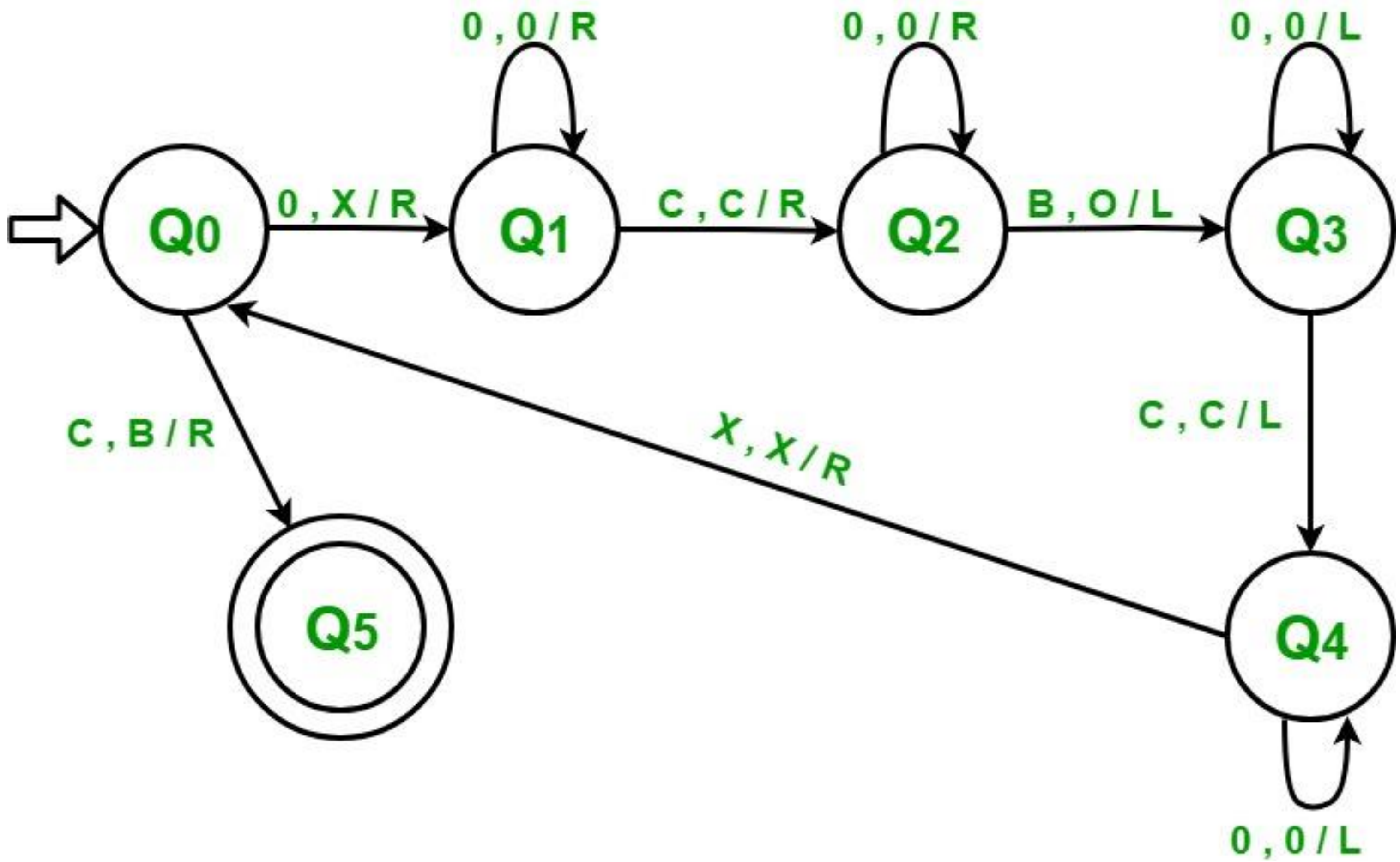
- For adding 2 numbers using a Turing machine, both these numbers are given as input to the Turing machine separated by a "c".
- **Examples –** (2 + 3) will be given as 0 0 c 0 0 0:
- Input : 0 0 c 0 0 0 // 2 + 3
- Output : 0 0 0 0 0 // 5

- Input : 0 0 0 0 c 0 0 0 // 4 + 3
- Output : 0 0 0 0 0 0 0 // 7

# Turing Machine for Addition

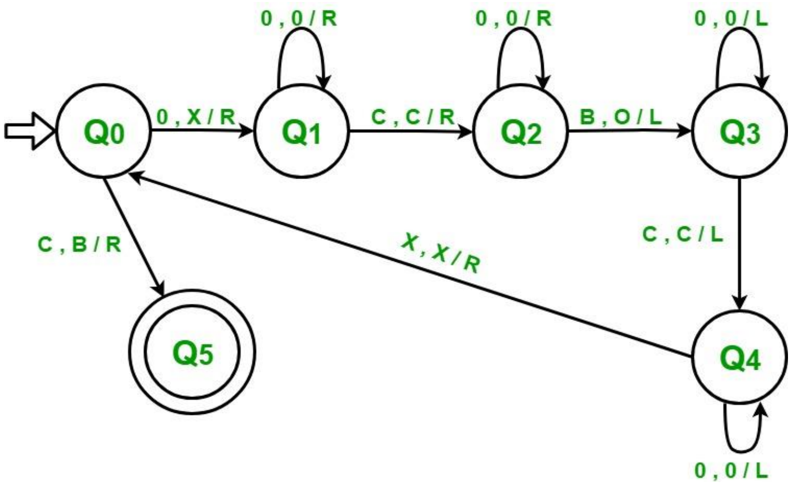
- **Step-1:** Convert 0 into X and goto step 2. If symbol is “c” then convert it into blank(B), move right and goto step-6.
- **Step-2:** Keep ignoring 0’s and move towards right. Ignore “c”, move right and goto step-3.
- **Step-3:** Keep ignoring 0’s and move towards right. Convert a blank(B) into 0, move left and goto step-4.
- **Step-4:** Keep ignoring 0’s and move towards left. Ignore “c”, move left and goto step-3.
- **Step-5:** Keep ignoring 0’s and move towards left. Ignore an X, move left and goto step-1.
- **Step-6:** End.

# Turing Machine for Addition



# Turing Machine for Addition

Example:



# References


- John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, *Introduction to Automata Theory*, Languages, and Computation, Pearson, 3rd Edition, 2011.
- Peter Linz, An Introduction to Formal Languages and Automata, Jones and Bartle Learning International, United Kingdom, 6th Edition, 2016.

Next Class: **Unit III**

# Programming Techniques for TM

**Thank you.**