# Tur2Spell

## Turing Machine Implementation for Spelling Turkish Words

Taalai ALMASOVA

Şevval MEHDER

# WHAT IS TURING MACHINE?

- A **Turing machine** is a mathematical model of computation that defines an abstract machine which manipulates symbols on a strip of tape according to a table of rules. [1]

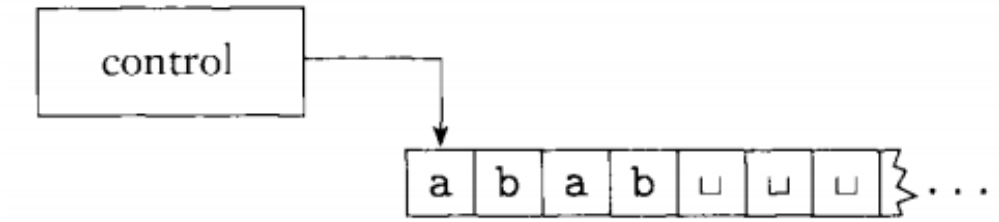



Figure 1.1 Schematic of a Turing machine[2]

# FORMAL DEFINITION OF TURING MACHINE

A Turing Machine is a 7-tuple, $(Q, \sum, \Gamma, \delta, q_0, q_{accept}, q_{reject})$, where Q, $\sum$, $\Gamma$ are all finite sets and

1. Q is the set of states,
2. $\sum$ is the input alphabet not containing the blank symbol #
3. $\Gamma$ is the tape alphabet, where # $\in \Gamma$
4. $\delta$ : Q x $\Gamma$ -> Q x $\Gamma$ x {L, R} is the transition function,
5. $q_0 \in$ Q is the start state,
6. $q_{accept} \in$ Q is the accept state and
7. $q_{reject} \in$ Q is the reject state, where $q_{accept} \neq q_{reject}$

# HOW TO SPELL TURKISH WORDS?

There are 3 rules according to TDK[3] :

1. If there is consonant between two vowels in word, this consonant has a syllable with the next vowel.
   For example: araba –> a-ra-ba
2. If there are two repeated consonant in word, the first one has a syllable with the previous vowel and the second one has a syllable with the next vowel.
   For example: sevmek -> sev-mek
3. If there are three repeated consonant in word, the last one has a syllable with the next vowel.
   For example: korkmak -> kork-mak

# WHAT ARE THE TUPLES OF TUR2SPELL

Q : $\{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}\}$

∑ : {'consonants': ['b', 'c', 'd', 'g', 'ğ', 'j', 'l', 'm', 'n', 'r', 'v', 'y', 'y', 'z', 'ç', 'f', 'h', 'k', 'p', 's', 'ş', 't'],
'vowels': ['a', 'ı', 'o', 'u', 'e', 'i', 'ö', 'ü'] }

Γ : {'consonants': ['b', 'c', 'd', 'g', 'ğ', 'j', 'l', 'm', 'n', 'r', 'v', 'y', 'y', 'z', 'ç', 'f', 'h', 'k', 'p', 's', 'ş', 't'],
'vowels': ['a', 'ı', 'o', 'u', 'e', 'i', 'ö', 'ü'] ,
'other': ['-', '#'] }

δ : Q x Γ -> Q x Γ x {L, R} (the next page),

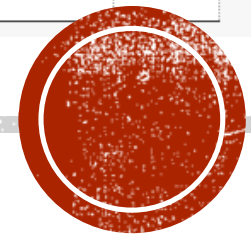
$q_0$ is the start state,

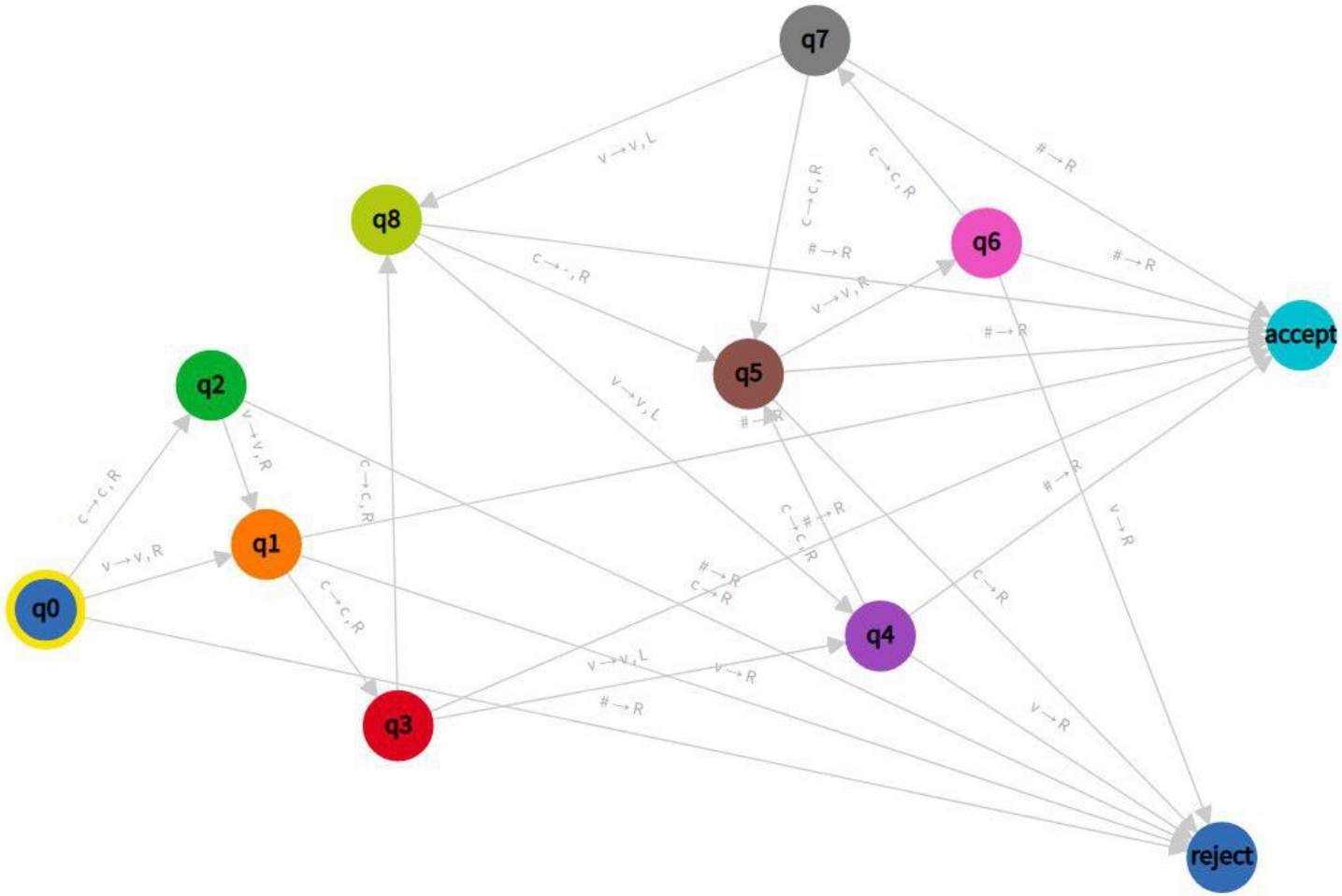
$q_9$ is accept state and

$q_{10}$ is the reject state.

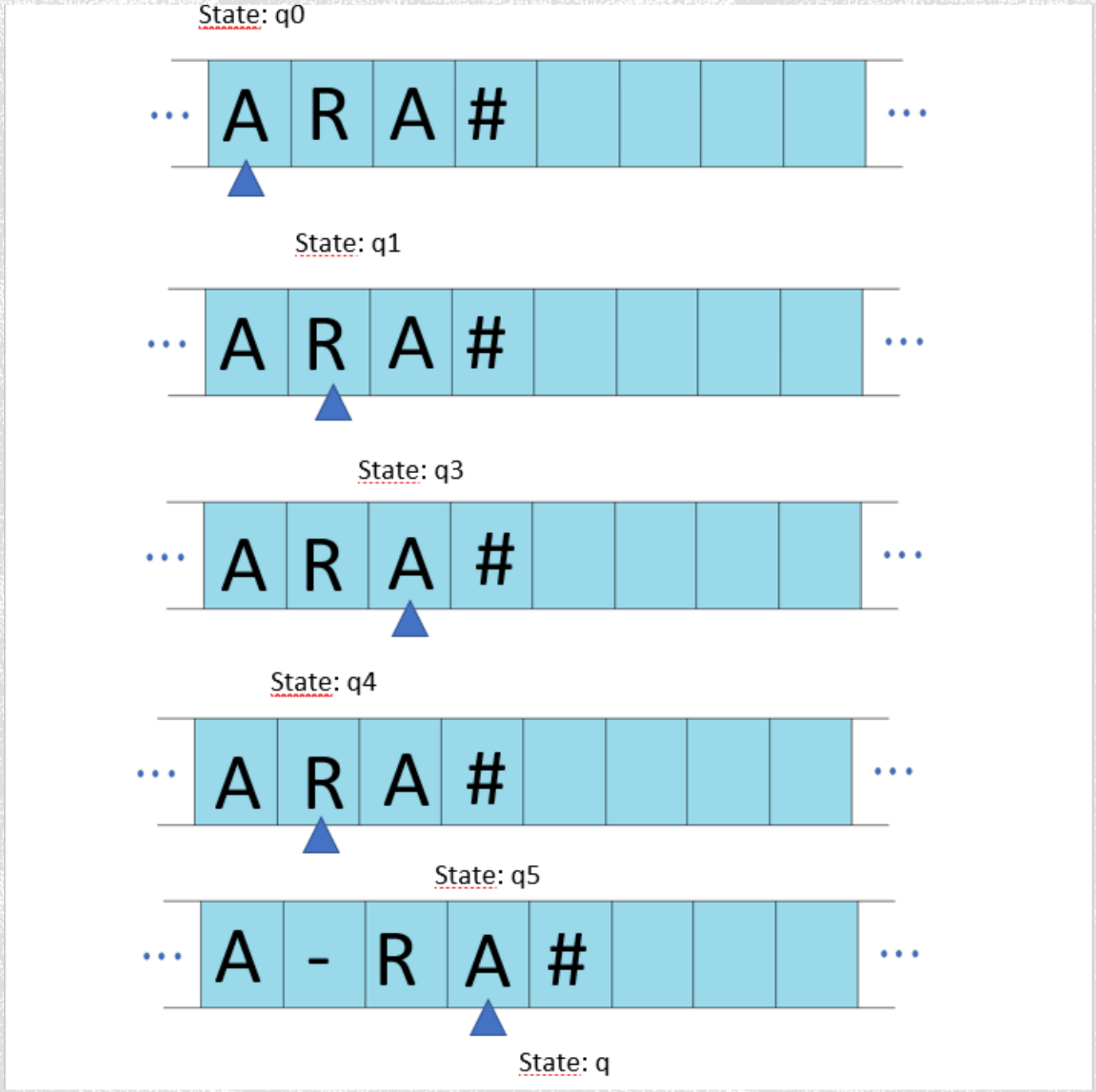
# THE TRANSITION FUNCTION

| | | c | v | # |
|---|---|---|---|---|
| 1 | | | | |
| 2 | **q0** | *(q2, c, R)* | *(q1, v, R)* | *(reject)* |
| 3 | **q1** | *(q3, c, R)* | *(reject)* | *(accept)* |
| 4 | **q2** | *(reject)* | *(q1, v, R)* | *(reject)* |
| 5 | **q3** | *(q8, c, R)* | *(q4, v, L)* | *(accept)* |
| 6 | **q4** | *(q5, c, R)* | *(reject)* | *(accept)* |
| 7 | **q5** | *(reject)* | *(q6, v, R)* | *(accept)* |
| 8 | **q6** | *(q7, c, R)* | *(reject)* | *(accept)* |
| 9 | **q7** | *(q5, c, R)* | *(q8, v, L)* | *(accept)* |
| 10 | **q8** | *(q3, -c, R)* | *(q4, v, L)* | *(accept)* |
| 11 | **accept** | | | |
| 12 | **reject** | | | |






# THE MACHINE


State: q0
A R A #
State: q1
A R A #
State: q3
A R A #
State: q4
A R A #
State: q5
A - R A #
State: q

# EXAMPLE

# REFERENCES


- [1] https://en.wikipedia.org/wiki/Turing_machine
- [2] **Introduction to the Theory of** Computation, Second Edition. by Michael Sipser.
- [3] http://www.tdk.gov.tr/index.php?id=208:Hece..&option=com_content