

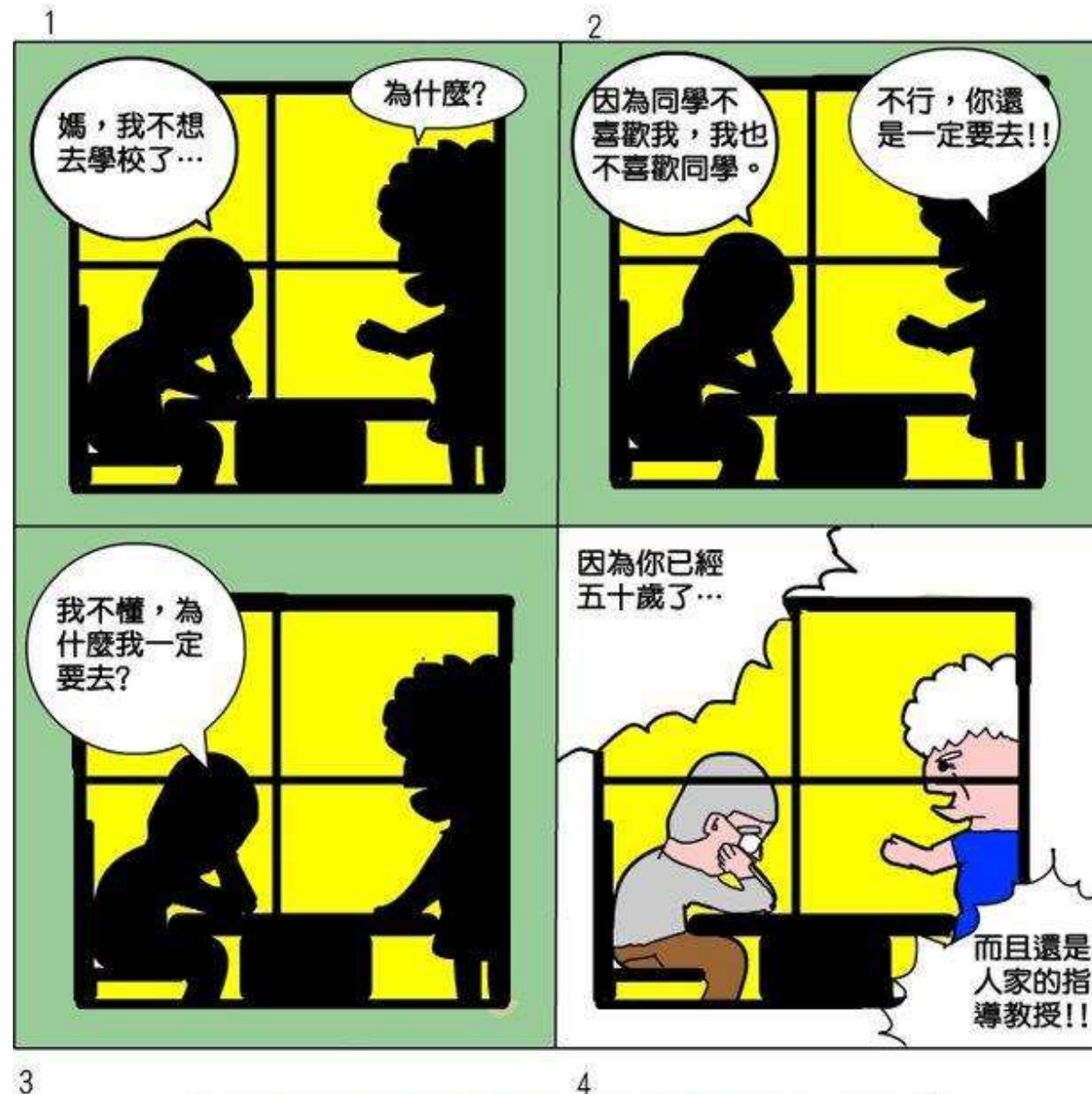
2023

Program Design II

Kun-Ta Chuang
Department of Computer Science and Information Engineering
National Cheng Kung University



開學振作點



研究生不死，只是生不如死

Preliminaries

- **Course Information**
 - **Course Name:** Program Design II
(程式設計II)
 - **Time:** Tuesday 9:10 AM ~ 12:00 AM
 - **Course Website:** ncku moodle

Preliminaries

- **Instructor**

- Name: Kun-Ta Chuang (莊坤達)
- E-mail: ktchuang@mail.ncku.edu.tw
- Office Hours: Tuesday 2:00 pm ~ 4:00 pm
- Office Location: CSIE 6F 608

- **Teaching Assistants**

- 梁樂謙、蔡易彤、鄭驊成
- EMAIL TO ncku.pd2.ta@netdb.csie.ncku.edu.tw

- **Undergraduate TA (UTA)**

- We will arrange 5~6 UTAs and schedule 2~3 3-hours slots every week

Biosketch

- 成功大學資訊工程學系 副教授
- 成功大學數位生活科技研發中心 主任
- 行政院 科技會報辦公室 科技政策專家室智慧資通群組
領域專家
- 台南市 研考委員
- 教育部 大數據分析人才培育中心 主持人
- 成功大學 區塊鏈社 指導老師
- Research Interest - Data Science/Machine Learning/Blockchain
- Applied Domain - Energy/Healthcare
- 產學合作 - 東電、Google、Microsoft、NXP、台積電、
帆宣、台電、中國人壽等

Preliminaries



- Startup -- Youthwant (1999-2001)
- Startup -- UniPattern (2002-2004)
- Join Synopsys (2006 – 2011)
 - Served as a Senior Engineer (國防役)
- Synopsys is the world leader in EDA
- Division of Synopsys Design-Rule Check
 - R&D teams in US/Taiwan (Taiwan: ~40 engineers)
 - Customers includes
 - IC gaints, e.g., Intel, Samsung, TI, Nvidia, Qualcomm, Broadcom, Novatek;
 - Major IDMs, e.g., Panasonic, Toshiba;
 - Key foundries, TSMC, UMC, Global Foundries



SNPS - NASDAQ
新思科技

\$333.26 ↑4,233.68% +325.57 最長

9月14日, 上午11:59:28 [UTC-4] · USD · NASDAQ · 免責事項

1天 5天 1個月 6個月 YTD 1年 5年 最長



SYNOPSYS®



current or previous clients, and I do not hold equity positions in any of them.

The Best Software Companies To Work For

The following table ranks the PWC List of Top 100 Global Software Leaders by the percentage of employees who would recommend their employer to a friend. Agfa HealthCare, CompuGROUP Holding, Constellation Software, DATEV, Hexagon, TOTVS and Visma aren't included in this table as the number of reviews on Glassdoor are very small.

Company	% of employees who would recommend this company to a friend	% of Employees who approve of the CEO as of July 12, 2013 on Glassdoor
Google	90%	95%
InterSystems	88%	93%
Citrix Systems	85%	92%
SAP	84%	94%
Adobe	84%	70%
SAS	84%	87%
Intel	83%	92%
Synopsys	83%	96%
Informatica	83%	92%
NetApp	82%	91%
Apple	81%	93%
Mentor Graphics	81%	86%
Bentley Systems	80%	92%
Intuit	79%	91%
Red Hat	79%	94%
Teradata	78%	80%
Microsoft	77%	47%
Ericsson	77%	88%

媽,我在這裡



Louis Columbus

Contributor

+ Follow (191)

I'm serving as Product Marketing Manager for Plex Systems, a leading provider of Cloud-based ERP systems for manufacturers where my responsibilities include new product introductions, messaging, marketing strategy, business development, and competitive strategy. Previous positions include senior analyst at AMR

+ show more

The author is a Forbes contributor. The opinions expressed are those of the writer.

LOUIS COLUMBUS' POPULAR POSTS

The Best Enterprise Software Companies And CEOs To Work For In 2013 129,432 views

Hype Cycle for Cloud Computing Shows Enterprises Finding Value in Big Data, Virtualization 43,886 views

The Best Cloud Computing Companies and CEOs to Work For in 2013 41,115 views

Gartner Hype Cycle for CRM Sales, 2012: Sales Turns to the Cloud for Quick Relief 39,974 views

Cloud Computing and Enterprise Software Forecast Update, 2012 38,584 views

MORE FROM LOUIS COLUMBUS

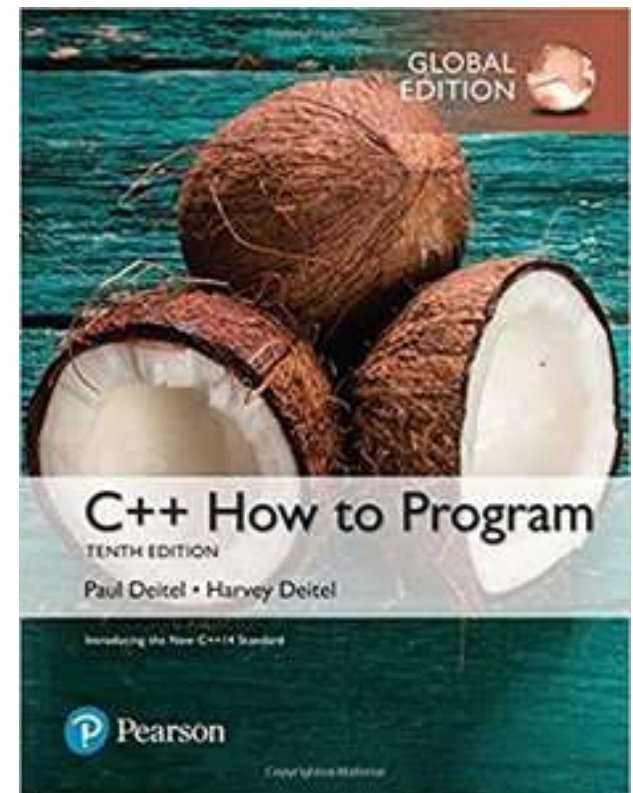
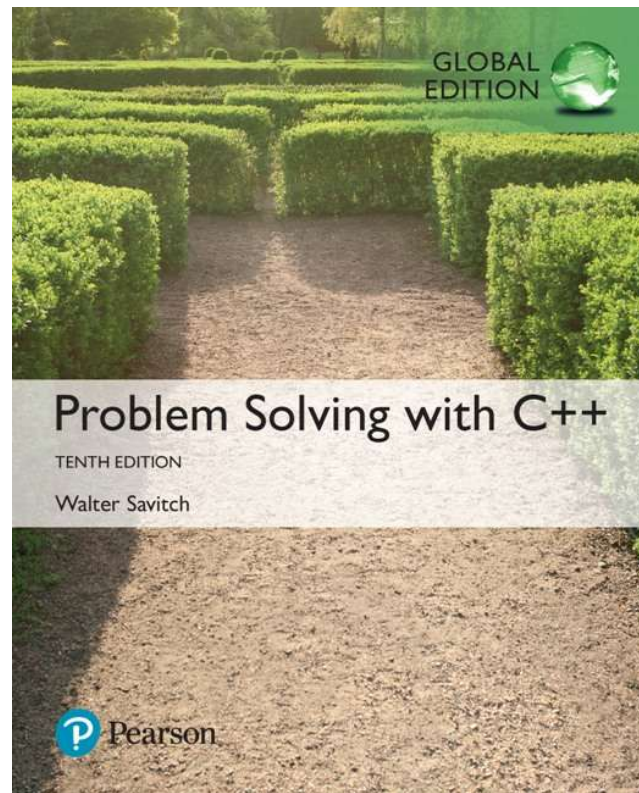


Preliminaries



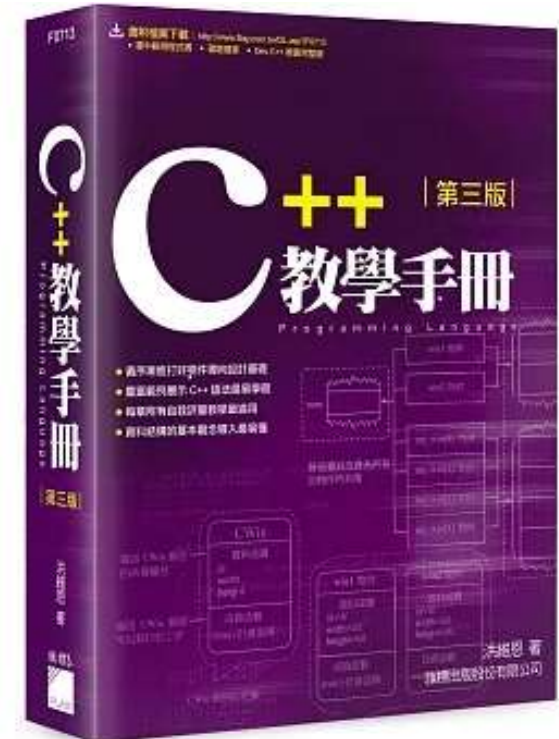
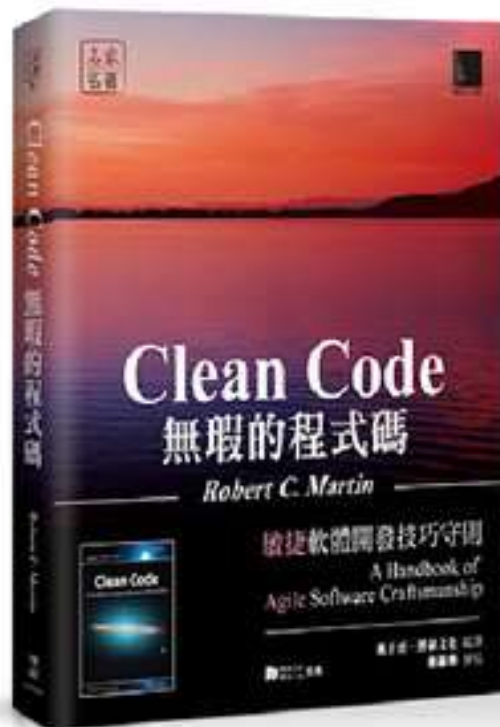
Reference Books

- P. Deitel, H. Deitel, “C++ How to Program”, 10/e, 2016
- W. Savitch, “Problem Solving with C++”, 10/e, 2018



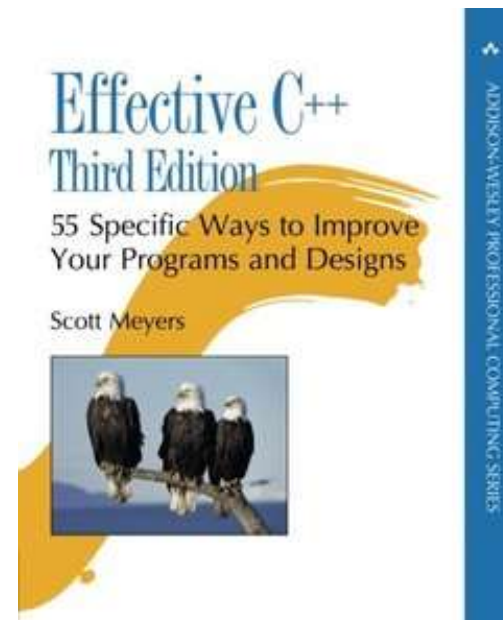
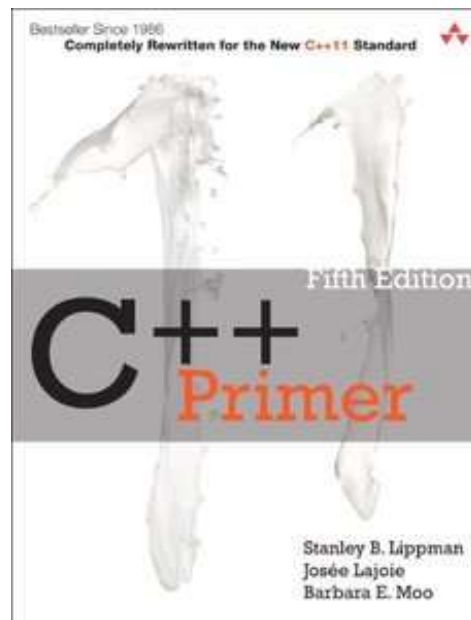
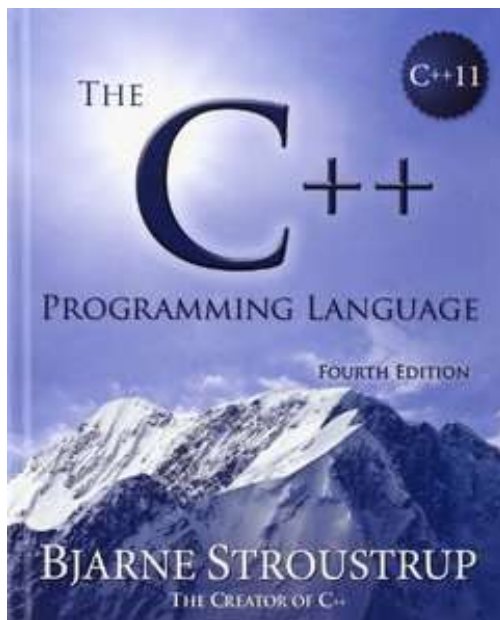
Good Handbook

- 洪維恩, “C++教學手冊”, 3/e, 2010
- 人月神話：軟體專案管理之道
- Robert C. Martin, Clean Code, 2013



If you need Bibles

- B. Stroustrup, “The C++ Programming Language”, 4/e, 2013
- Stanley B. Lippman, Josée Lajoie, Barbara E. Moo, “C++ Primer”, 5/e, 2012
- Scott Meyers , “Effective C++”, 3/e, 2005



Other Good Materials from..

- W3School

[https://www.w3schools.com/cpp/cpp_intro.
asp](https://www.w3schools.com/cpp/cpp_intro.asp)



Before we talk about Syllabus

學海無涯，資工更慘

YI

請問C++的課要教哪些內容? 可以用英文回答嗎?



Yes, of course! A typical C++ course may cover the following topics:

1. Introduction to programming concepts and the C++ language
2. Data types, variables, and basic operations
3. Control structures (e.g., conditional statements, loops)
4. Functions and function prototypes
5. Arrays and pointers
6. Classes and object-oriented programming (OOP) concepts
7. File Input/Output (I/O)
8. Exception handling
9. Standard Template Library (STL)
10. Debugging and testing techniques.

This list is by no means exhaustive, and the specific content of a C++ course may vary depending on the instructor's focus, the level of the course, and the intended audience.

I ask ChatGPT before I prepare the slide

But to show 工人智慧 “~~must/will/always/could/may/is likely to~~” win 人
工智慧...

小學老師說：

你要是**不好好讀書**
將來就**考不上國中**



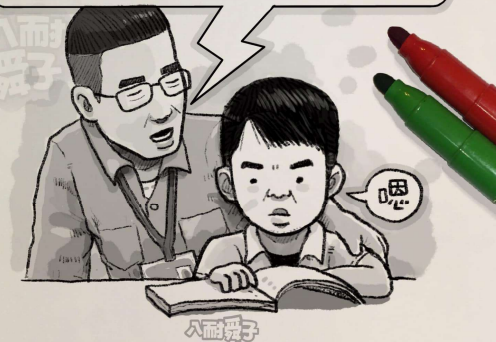
國中老師說：

你要是**不好好讀書**
將來就**考不上高中**



高中老師說：

你要是**不好好讀書**
將來就**考不上大學**



大學教授說：

你要是**不好好讀書**
將來就**找不到工作**



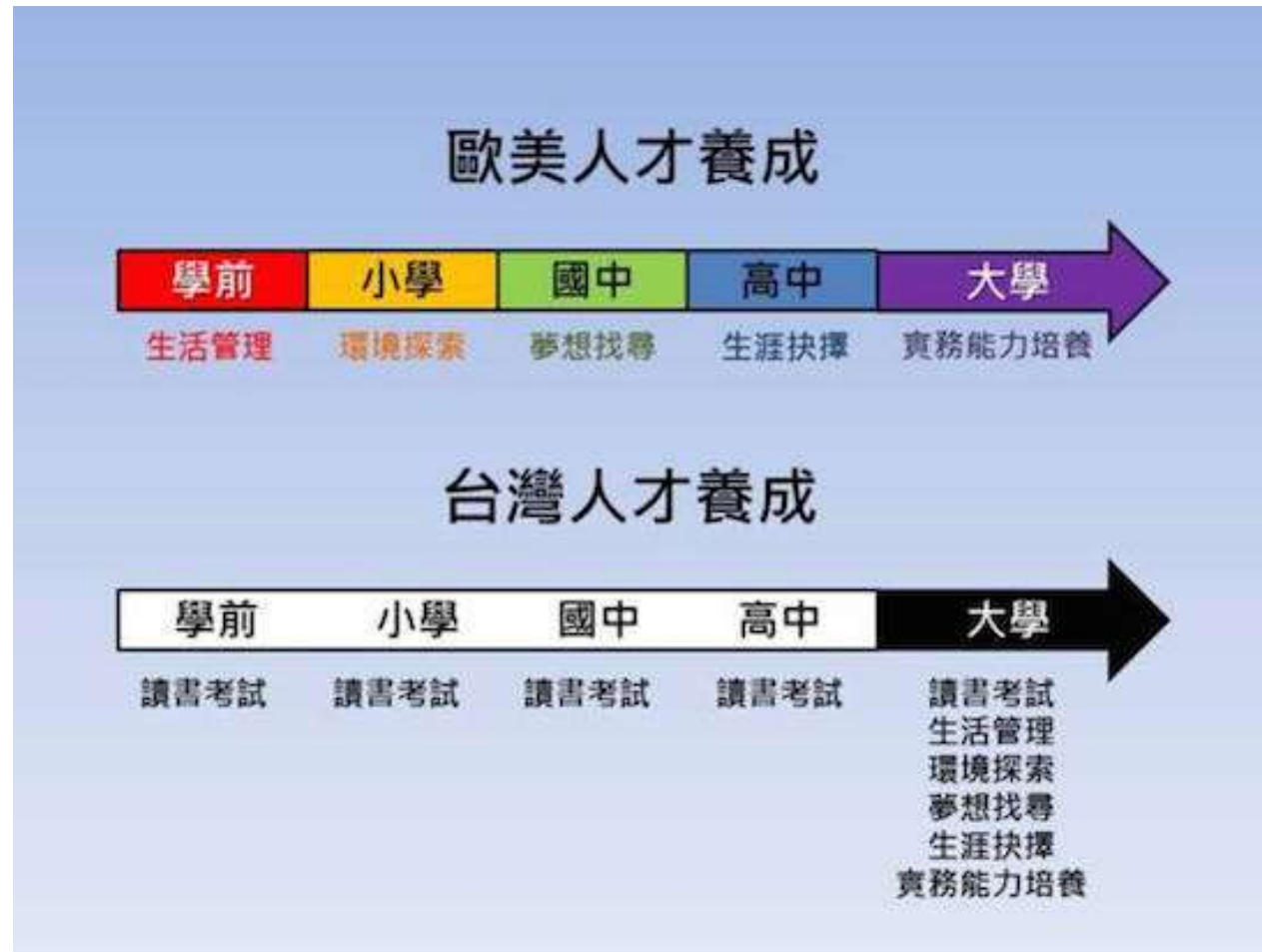
老闆說：

我的天啊!!
你在學校的時候...



是不是只會讀書啊!

八耐癡子



趕進度的碼農人生

We will build your "Infra" Strongly and Solidly – Conventional and Modern

Syllabus

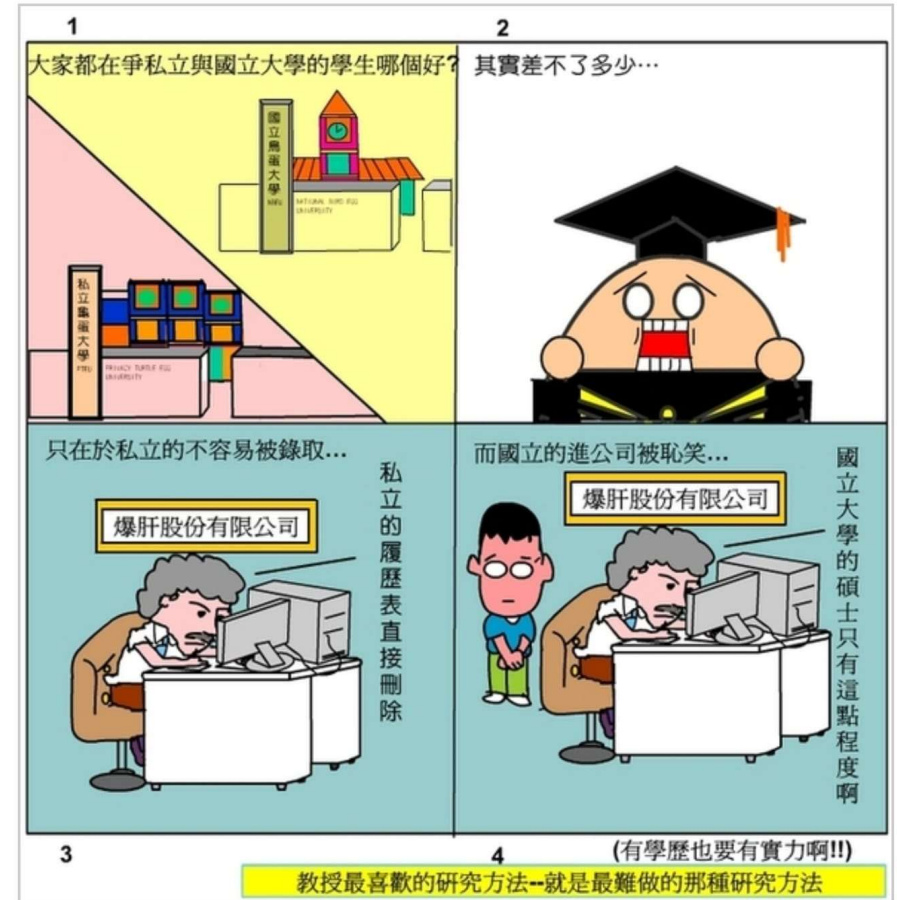
- **Foundation**
 - Preliminary (1W)
 - Linux ToolChain for C/C++ (1-2W)
 - Essential C++ Object-Oriented Programming (5W)
- **Practice for Data Structure/Algorithm (5W)**
 - Trees, Sorting, DP, Matching, Encoding
- **Advanced Programming Stack (5W)**
 - Modern C++ (Lambda, Container, a little of Parallelism)
 - Design Pattern and API Architecture
 - Testing Flow/CI
 - DevOps/Cloud
 - Open Source Ecosystem

ToolChain

- All in Linux
- Individual account will be created and submit to you during these two weeks

Grading

- 7 Small Labs (5%*7)
 - Check code correctness
 - We also consider plagiarism and the use of ChatGPT
- Two Term Projects (15%*2)
 - Ex. Search Engine Practice with indexing large dataset
 - Grading is ordered by your execution speed/correction as compared to your classmates
- Midterm (15%)
- Finalterm (20%)
- Industry Interview as your final oral testing (10% bonus)





Let's Coding with Happiness

老師你教的都是一些
沒有用的東西!!



我不許你們這樣說自己



Any Question?

History of Computation

Brief History of Computation

- In 1936, **Turing** proposed the model of **universal algorithm machine**, which later was called **Turing machine**.
- Some similar ideas were proposed by other scholars, including Stephen C. **Kleene**, Alonzo **Church**, Emil **Post**, etc., in different forms of models.



University of Cambridge
(N/A)



照片來源：[flickr](#)

圖靈測試（英語：Turing test，又譯圖靈試驗）是圖靈於1950年提出的一個關於判斷機器是否能夠思考的著名試驗，測試某機器是否能表現出與人等價或無法區分的智力。如果一個人（代號C）使用測試對象皆理解的語言去詢問兩個他不能看見的對象任意一串問題。對象為：一個是正常思維的人（代號B）、一個是機器（代號A）。如果經過若干詢問以後，C不能得出實質的區別來分辨A與B的不同，則此機器A通過圖靈測試。 - [維基百科](#)



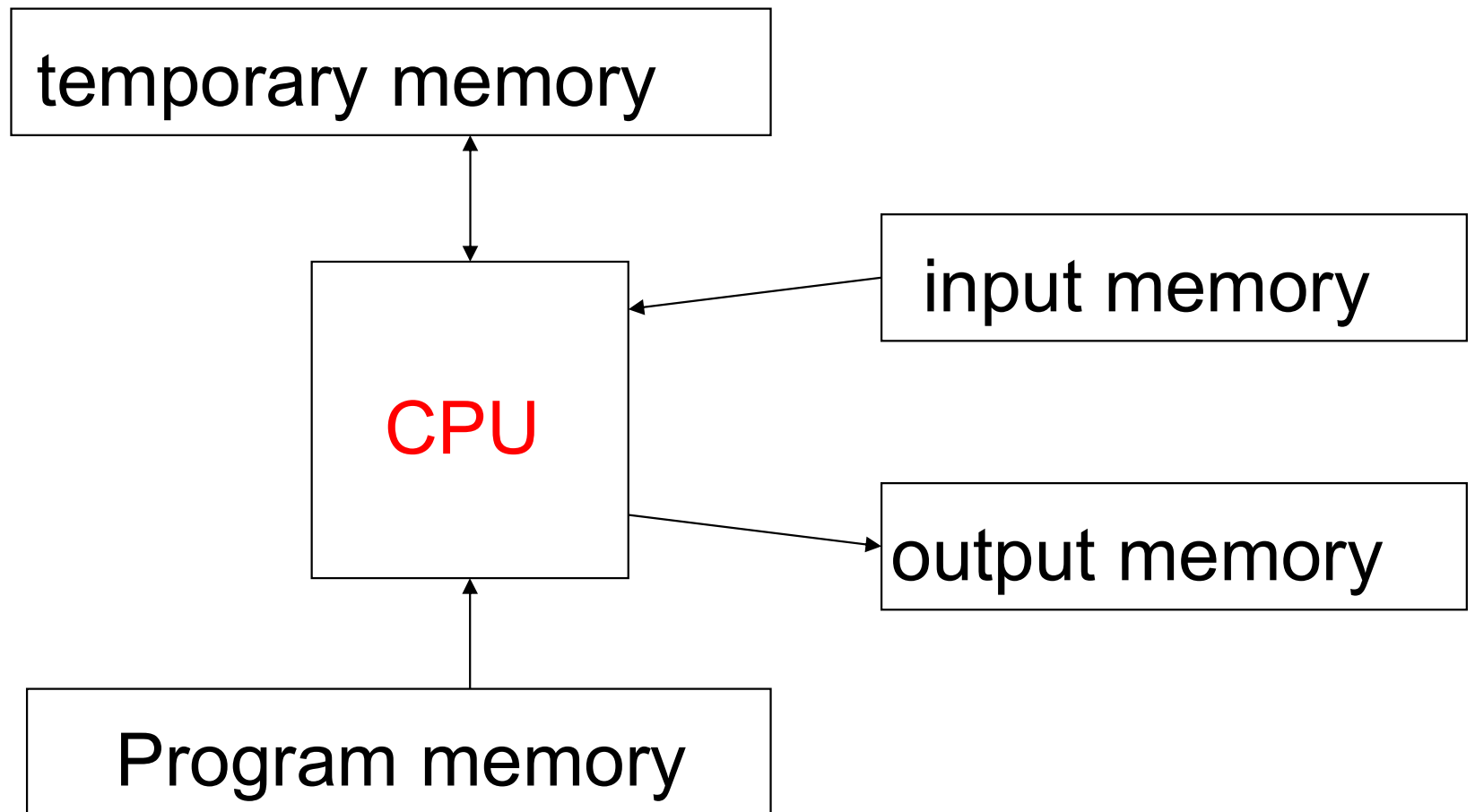
模仿遊戲 2015

Brief History of Computation

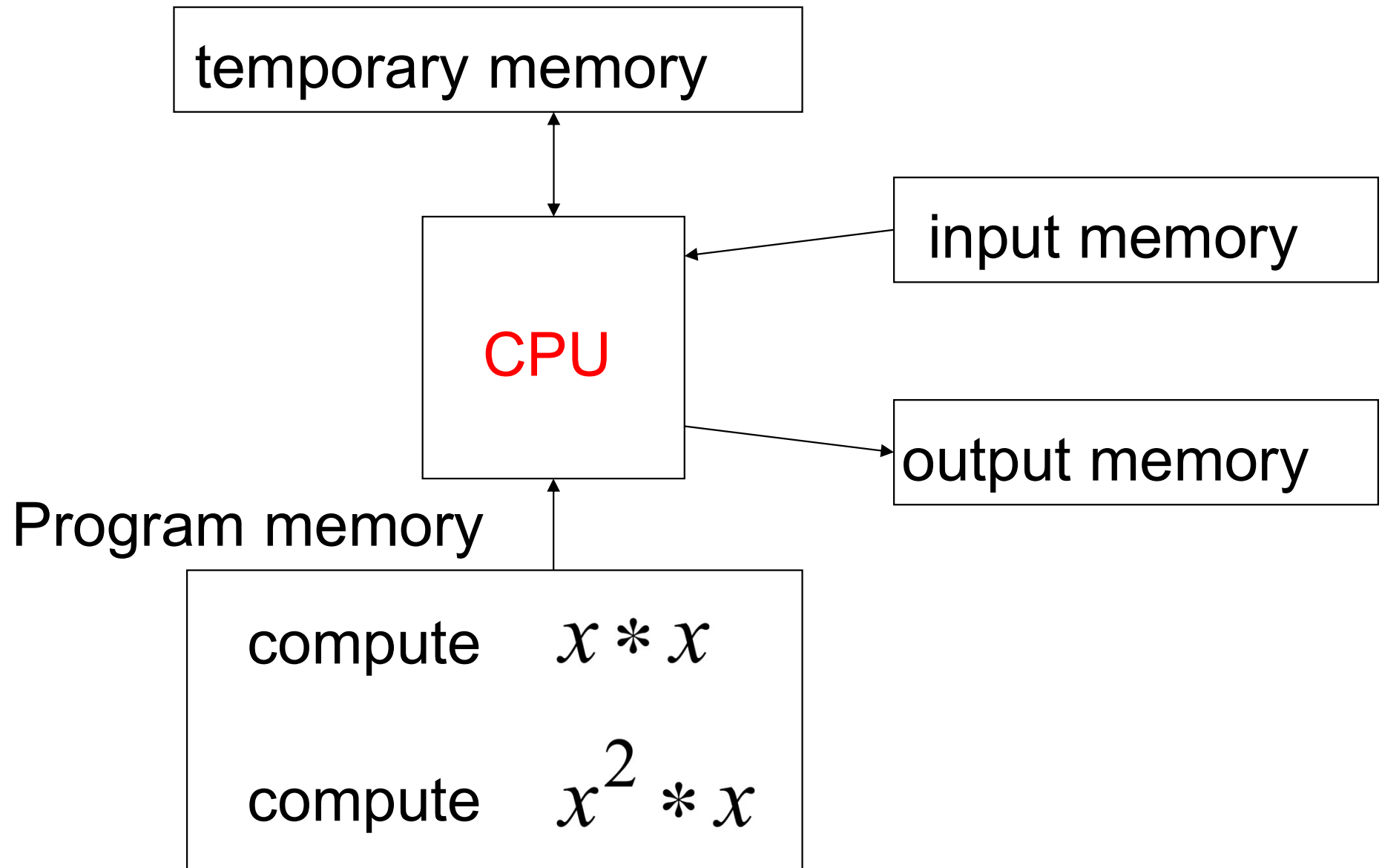
- In late 1940's, Von Neumann proposed the idea of **stored program** for computer models.
- In 1951, a **real computer** following this idea was constructed (**UNIVAC I**, the world's first commercially available computer, by Eckert-Mauchly Computer Company).



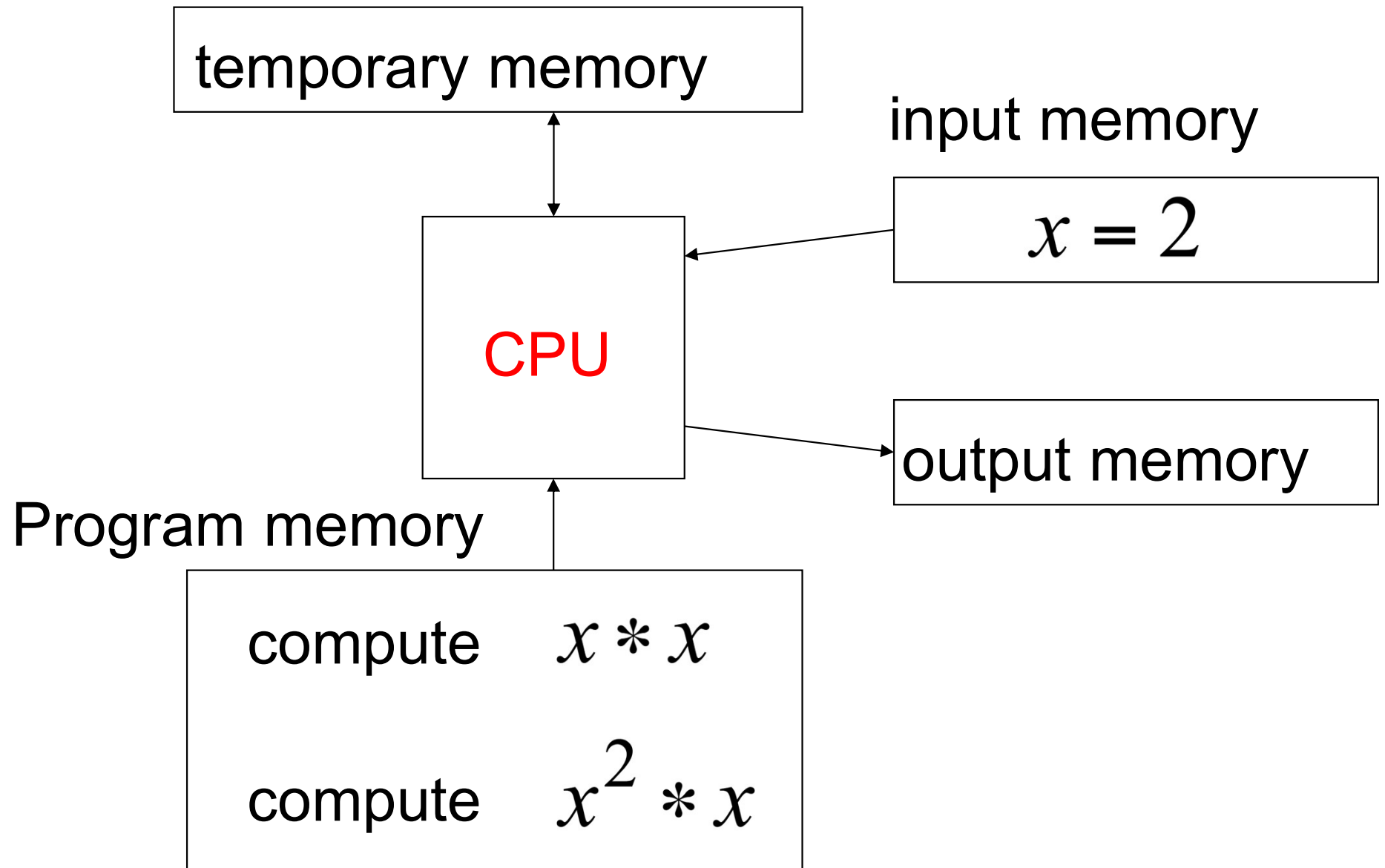
Princeton University
(N/A)

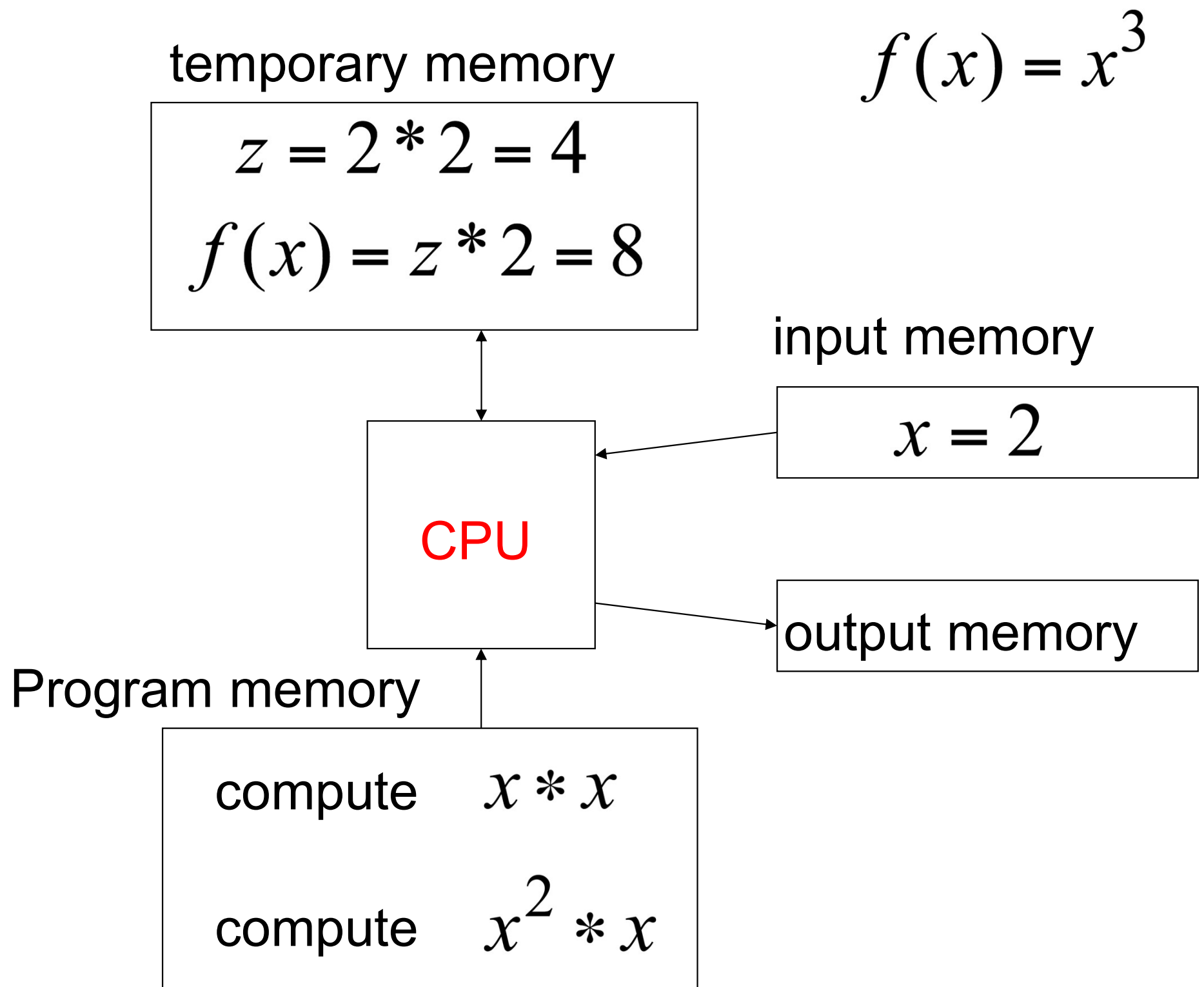


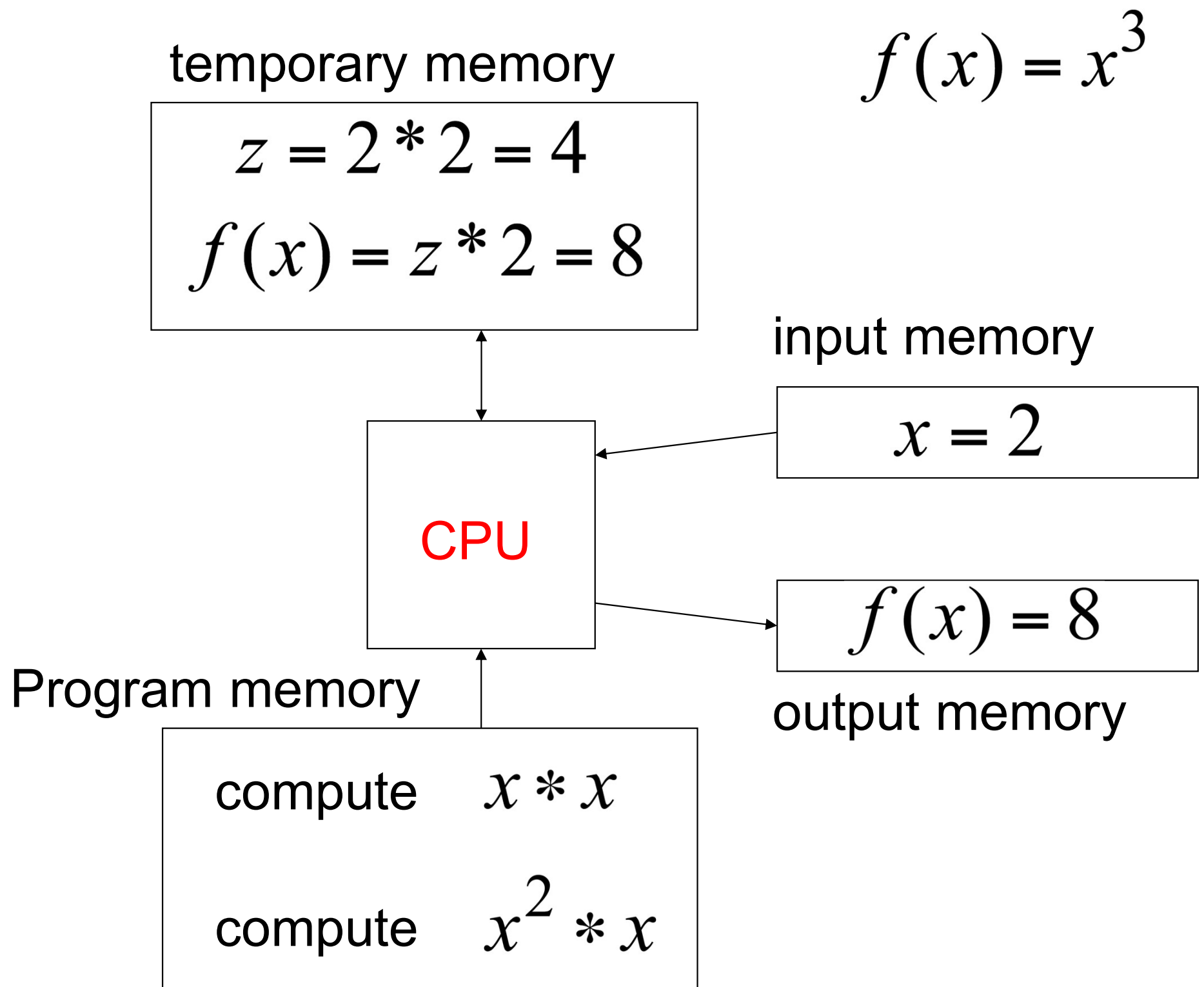
Example: $f(x) = x^3$



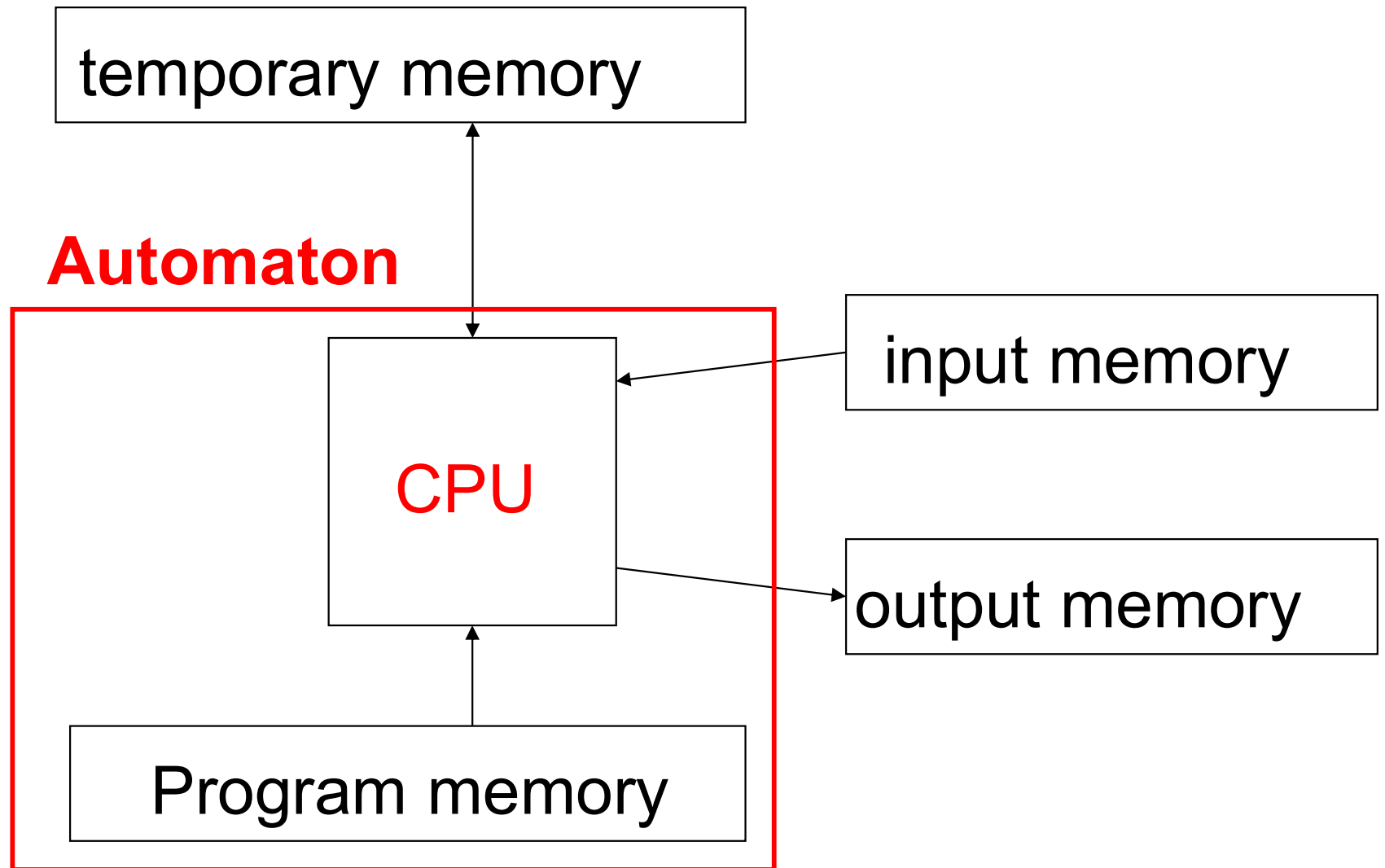
$$f(x) = x^3$$





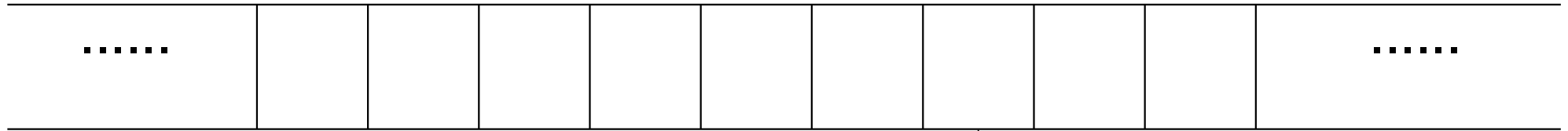


Automaton



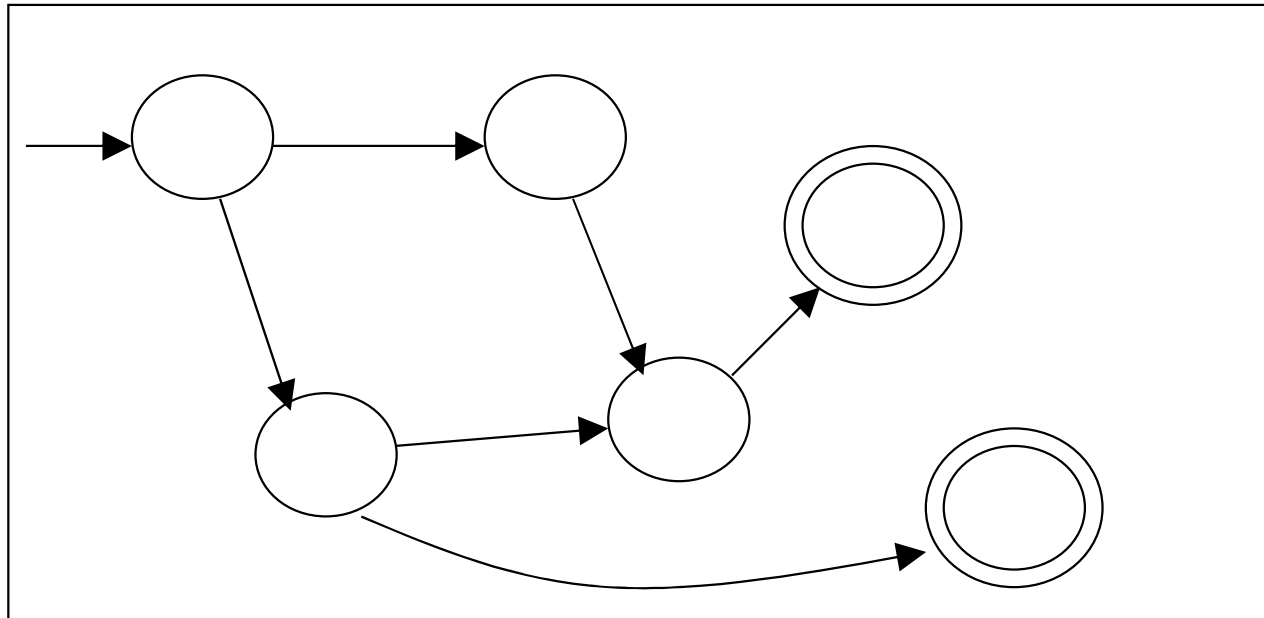
A Turing Machine

Tape



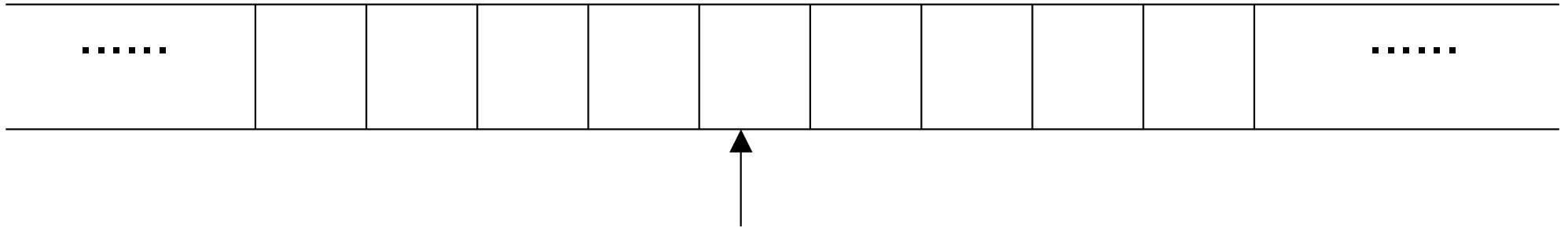
Read-Write head

Control Unit



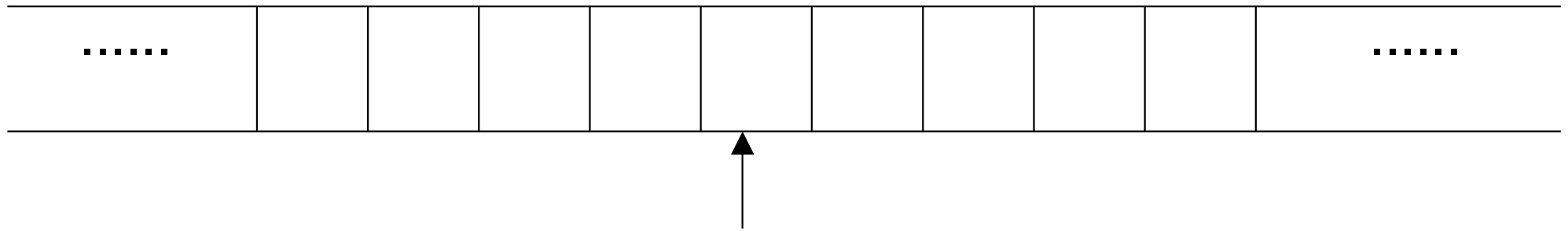
The Tape

No boundaries -- infinite length



Read-Write head

The head moves Left or Right



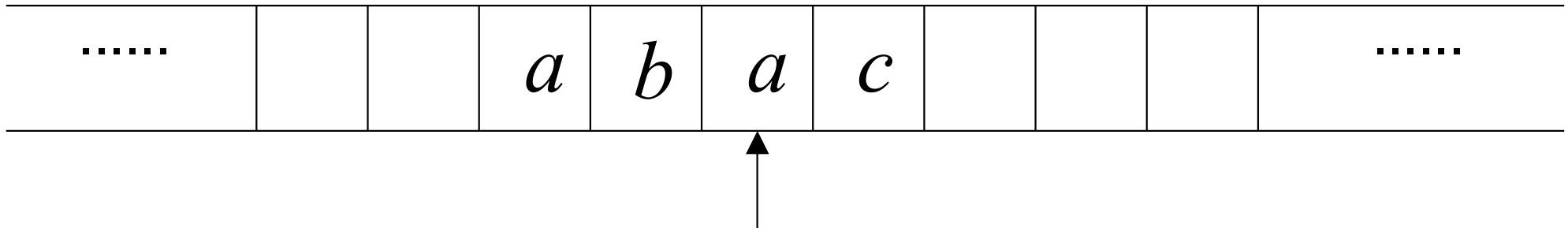
Read-Write head

The head at each time step:

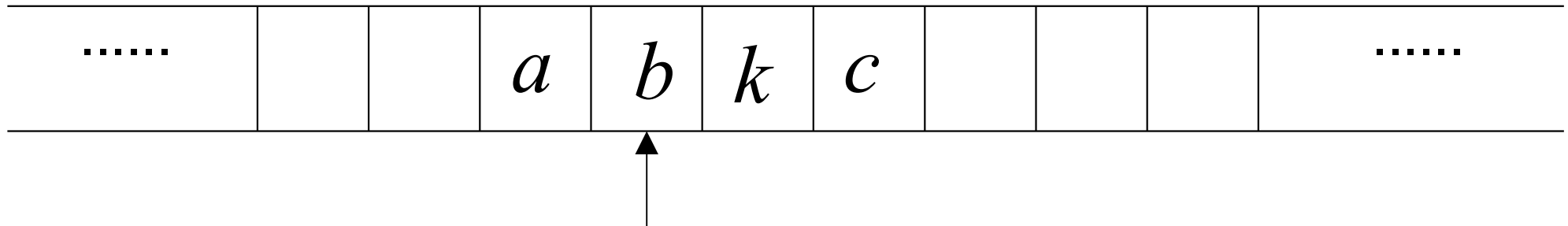
1. Reads a symbol
2. Writes a symbol
3. Moves Left or Right

Example:

Time 0

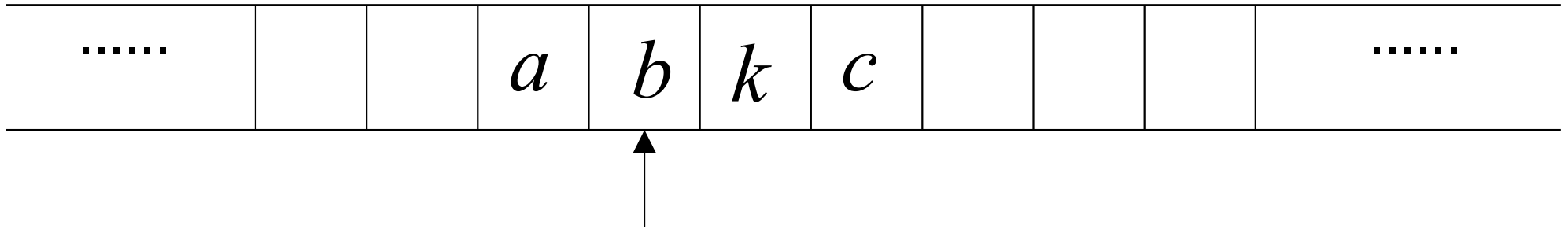


Time 1

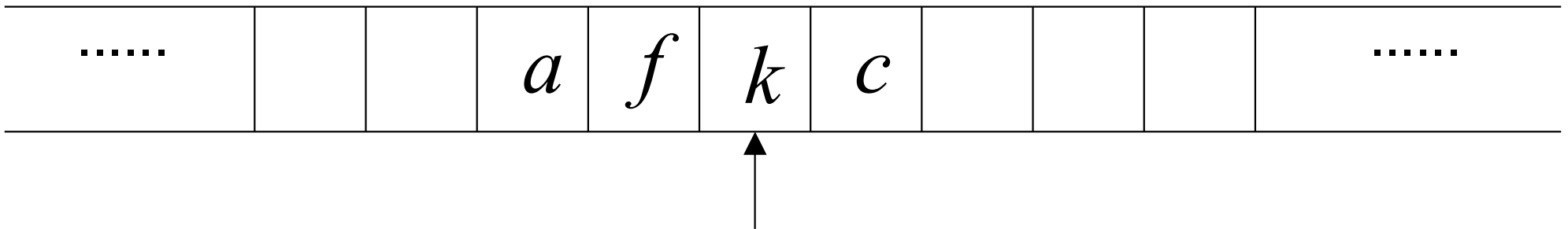


1. Reads a
2. Writes k
3. Moves Left

Time 1

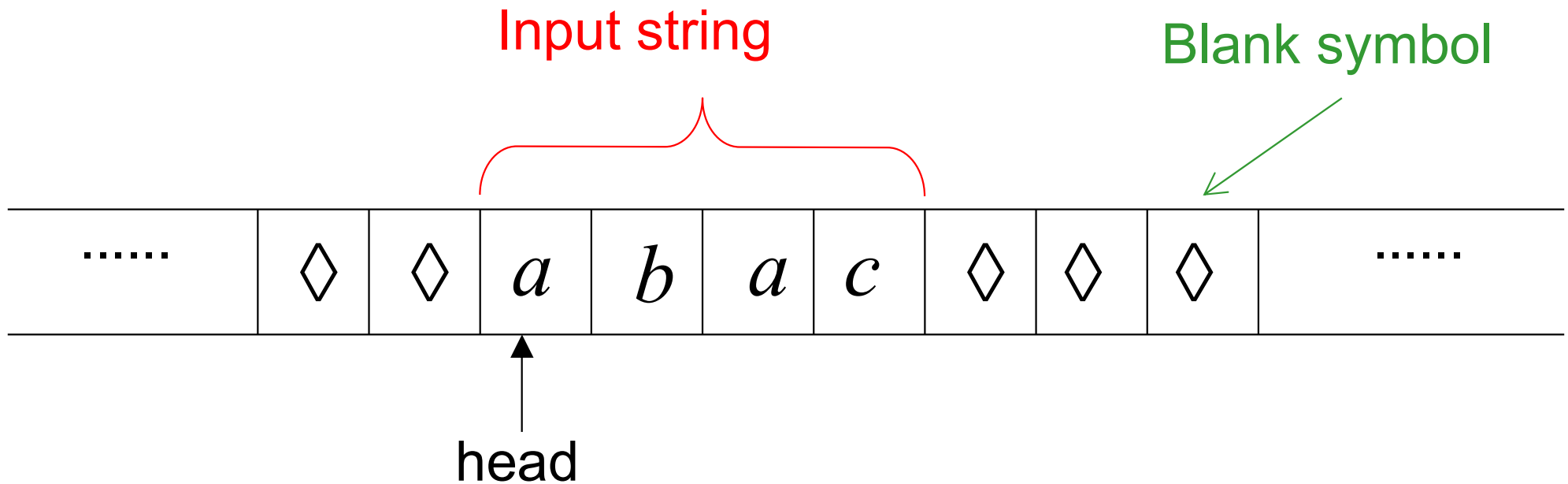


Time 2

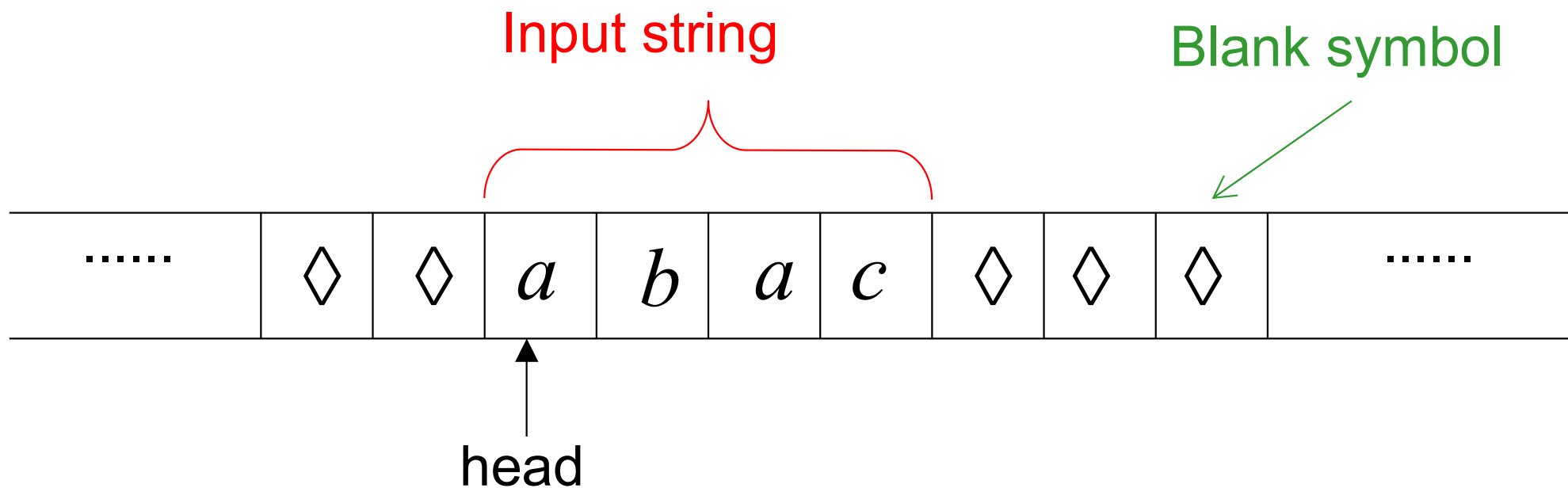


1. Reads b
2. Writes f
3. Moves Right

The Input String



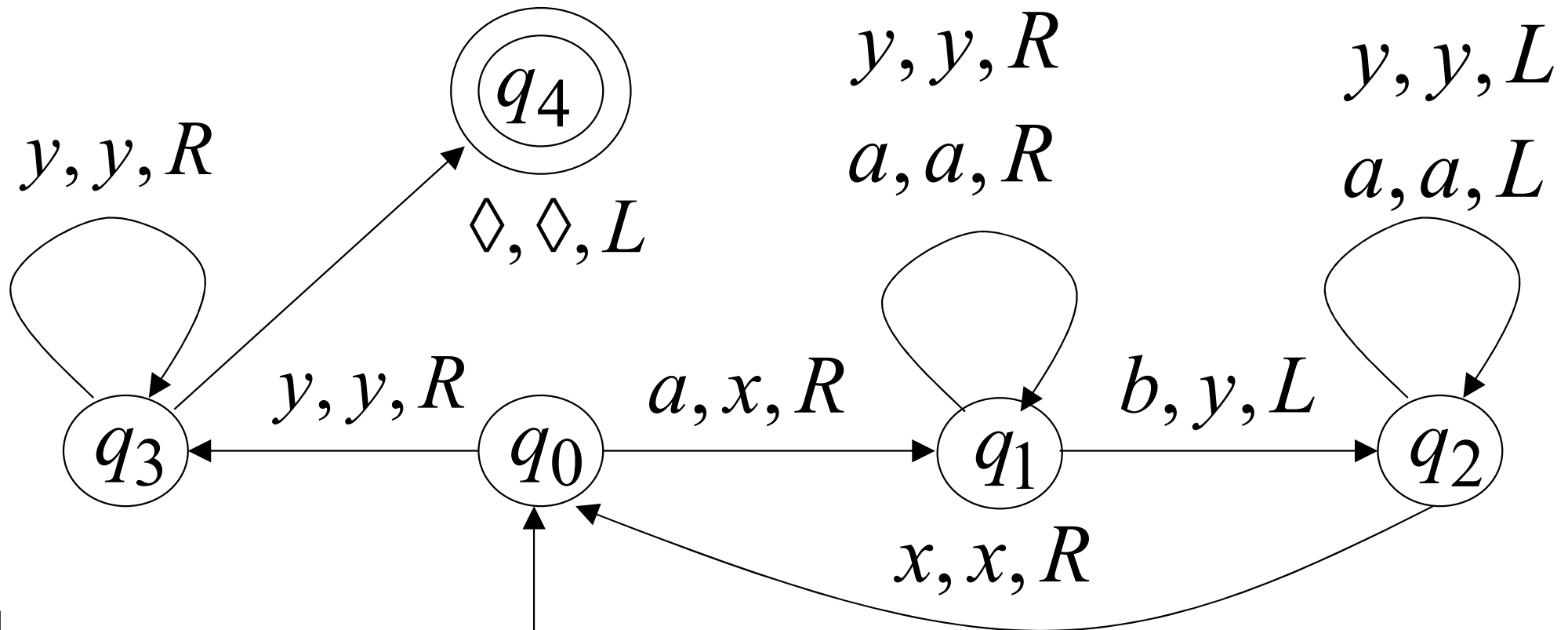
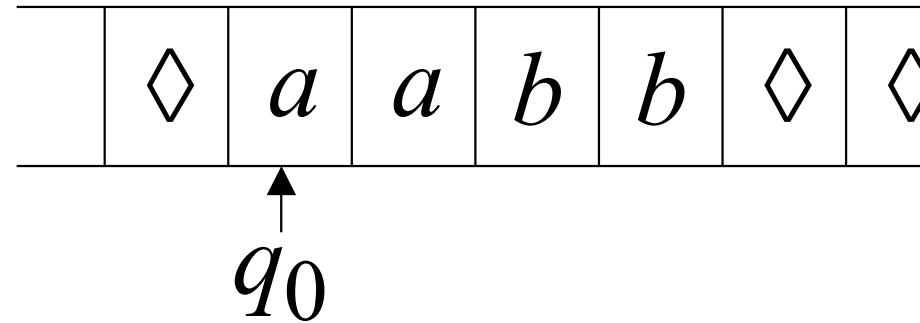
Head starts at the leftmost position
of the input string



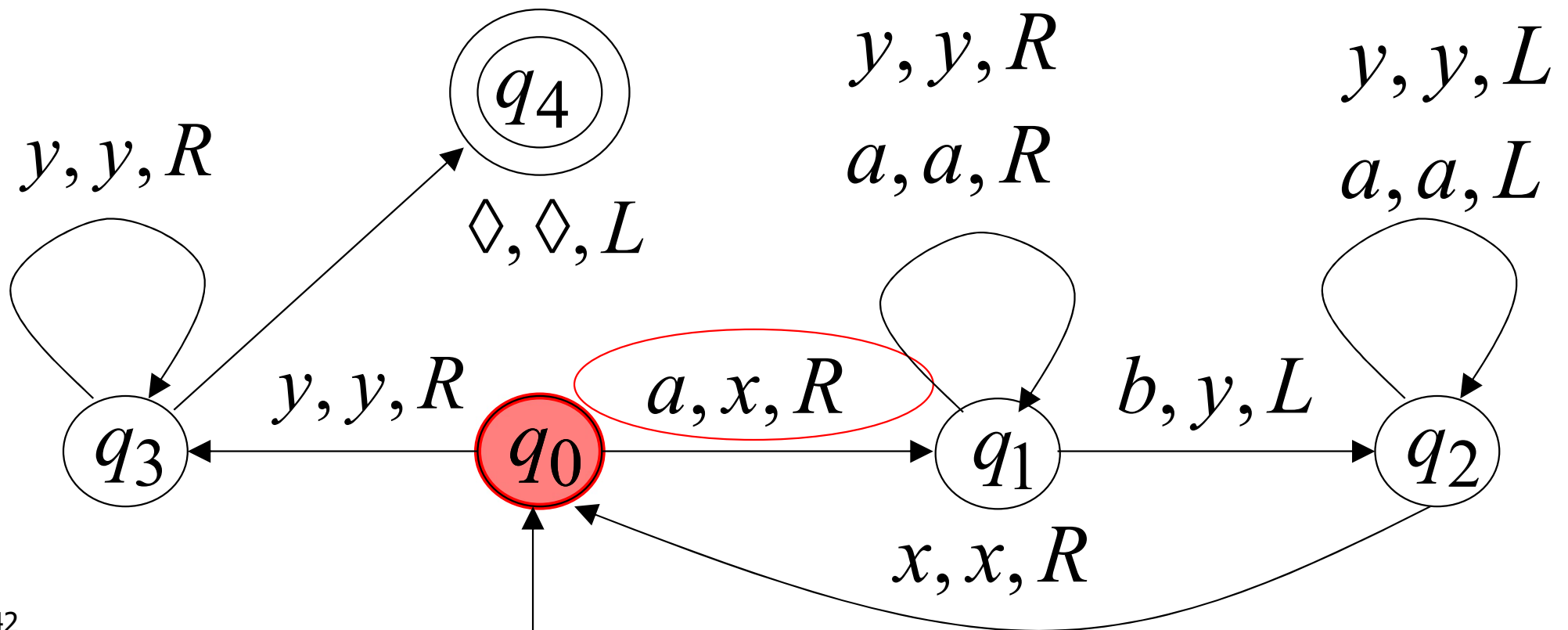
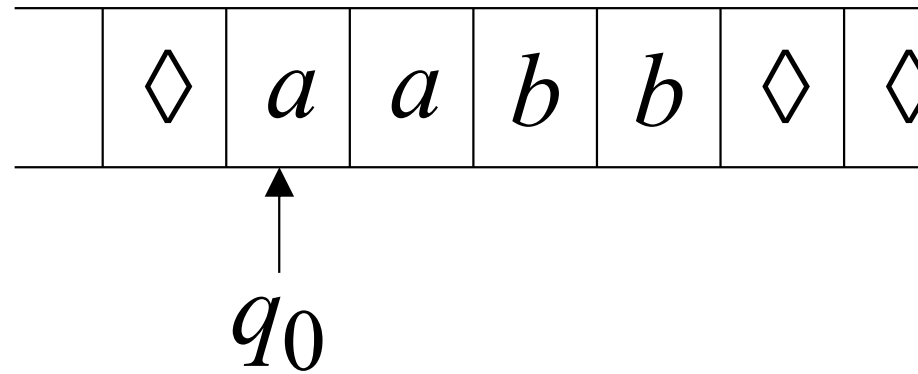
Remark: the input string is never empty

What is Computational Thinking

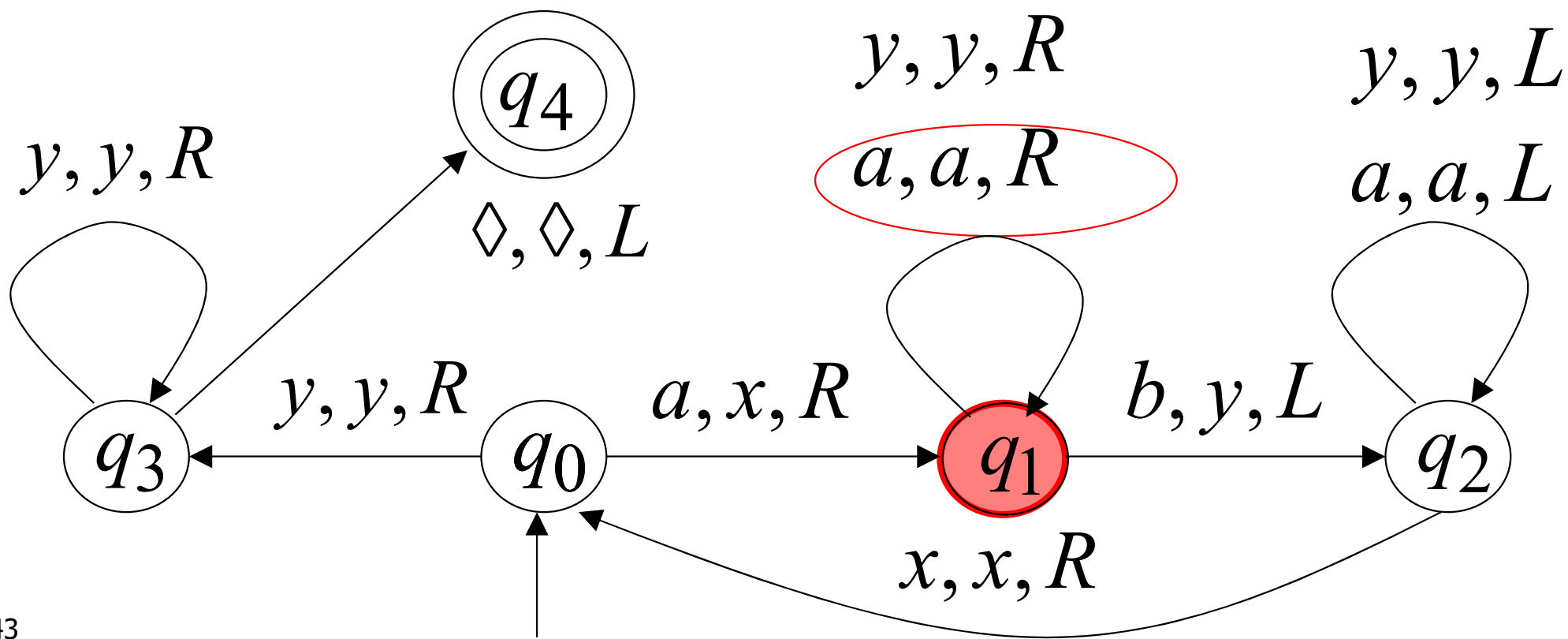
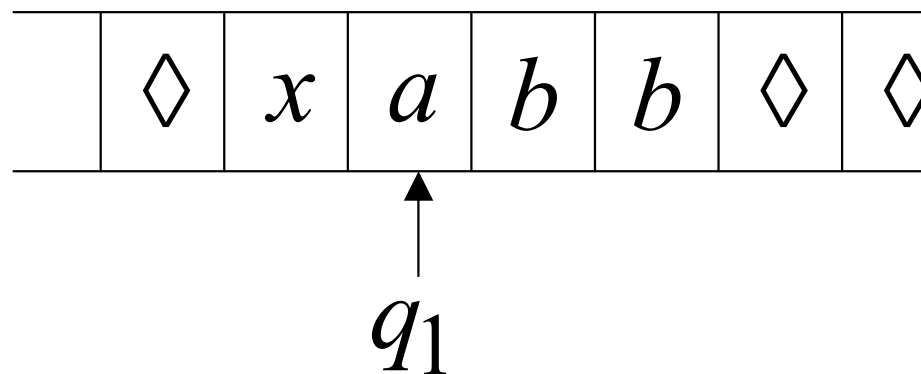
Build a Turing Machine to check $\{a^n b^n\}$



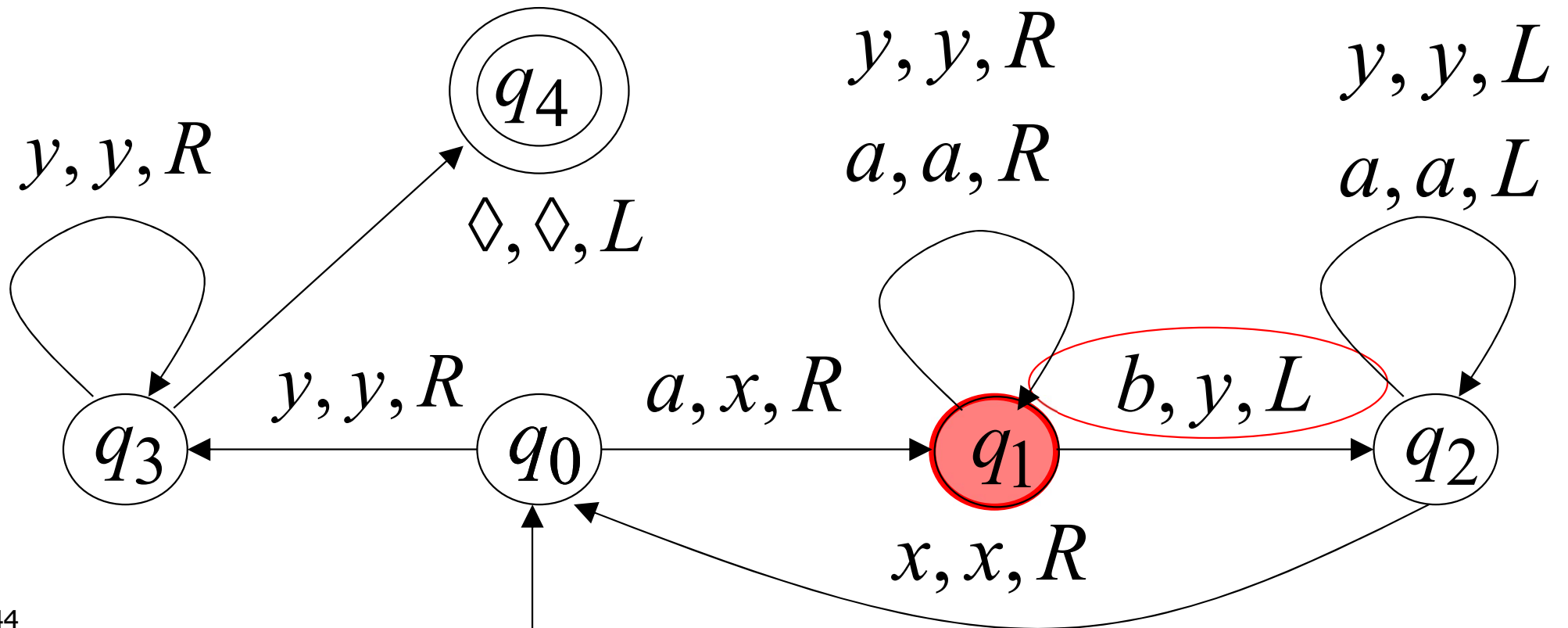
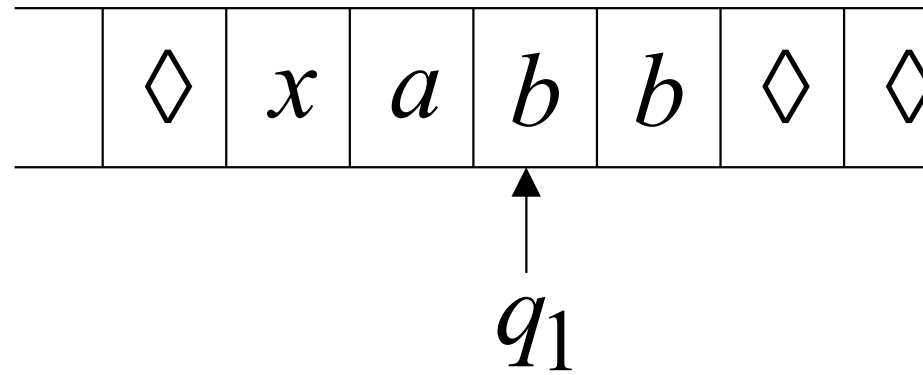
Time 0



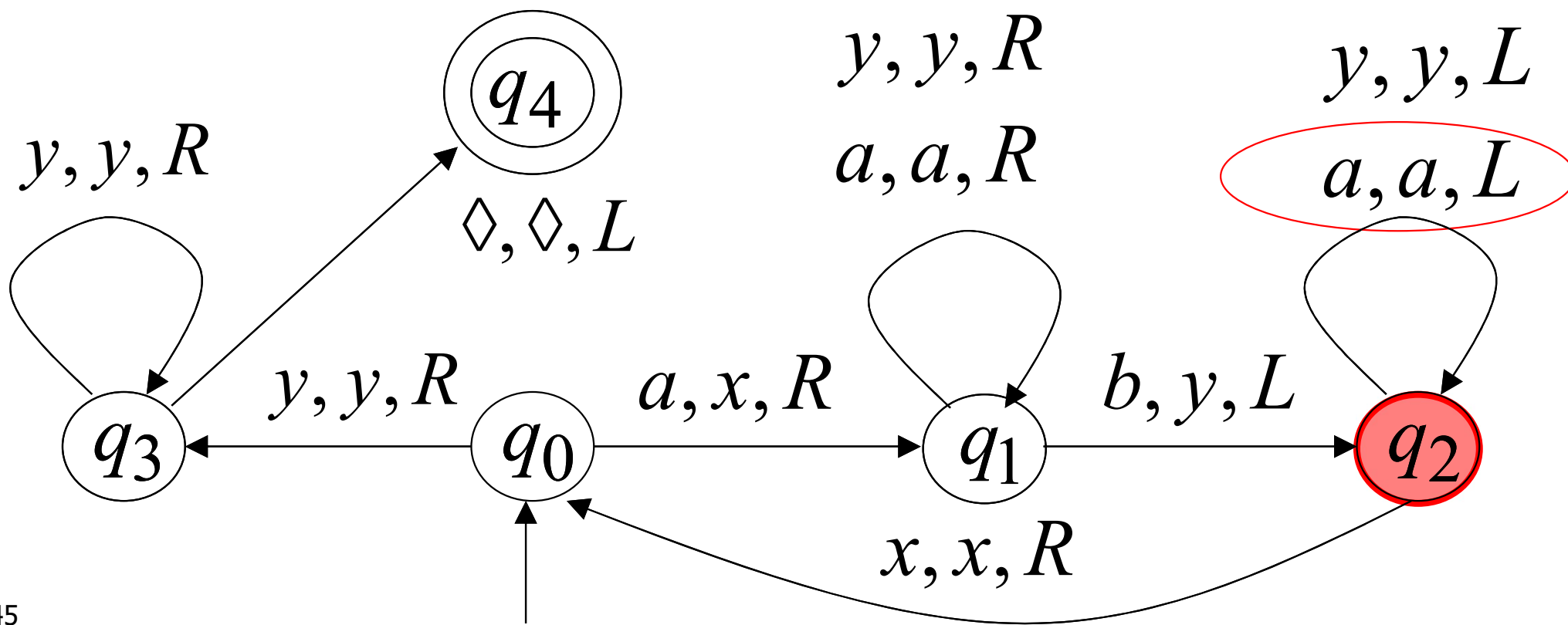
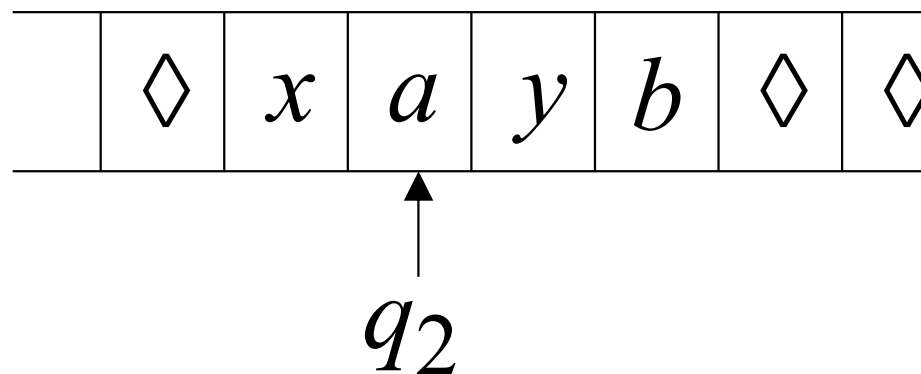
Time 1



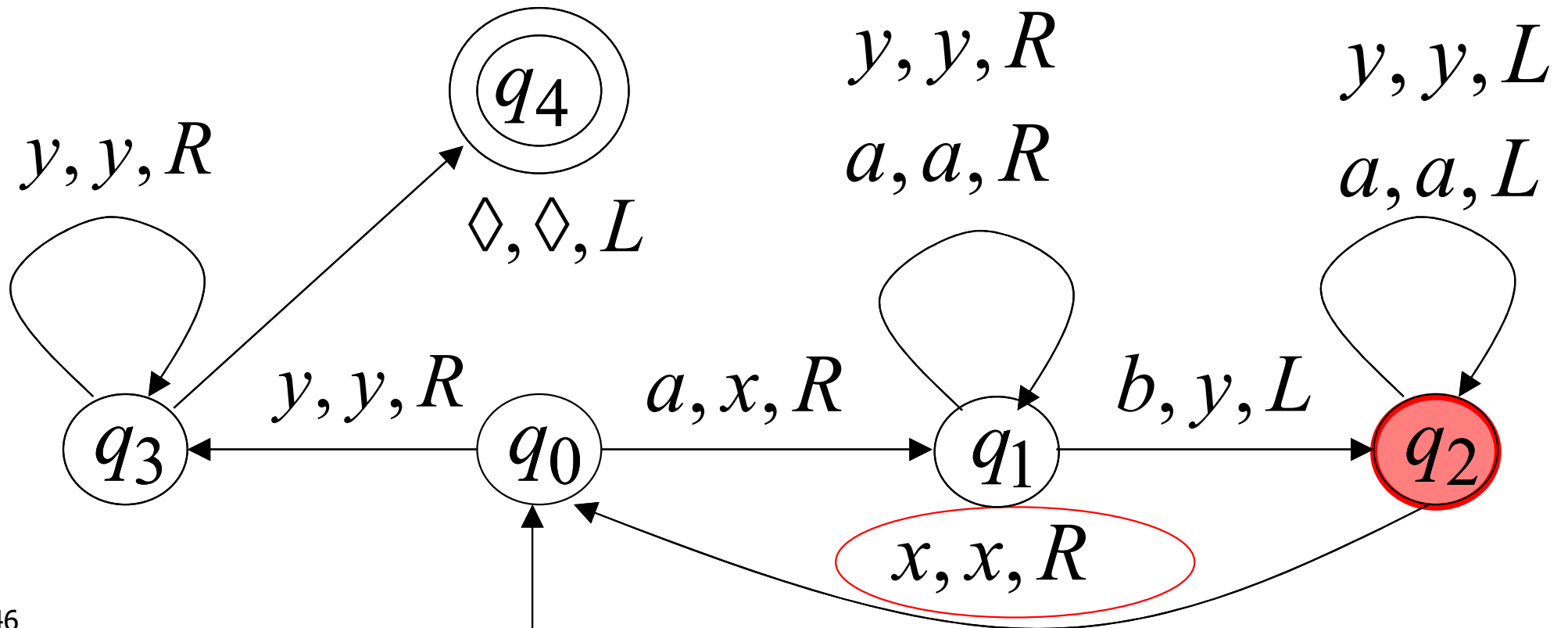
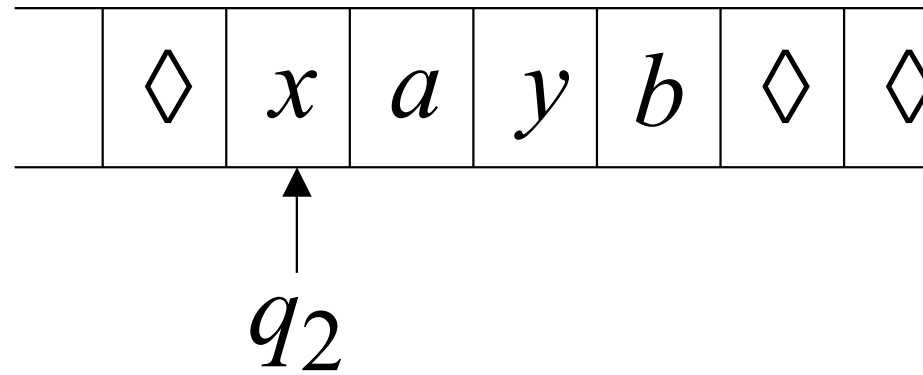
Time 2



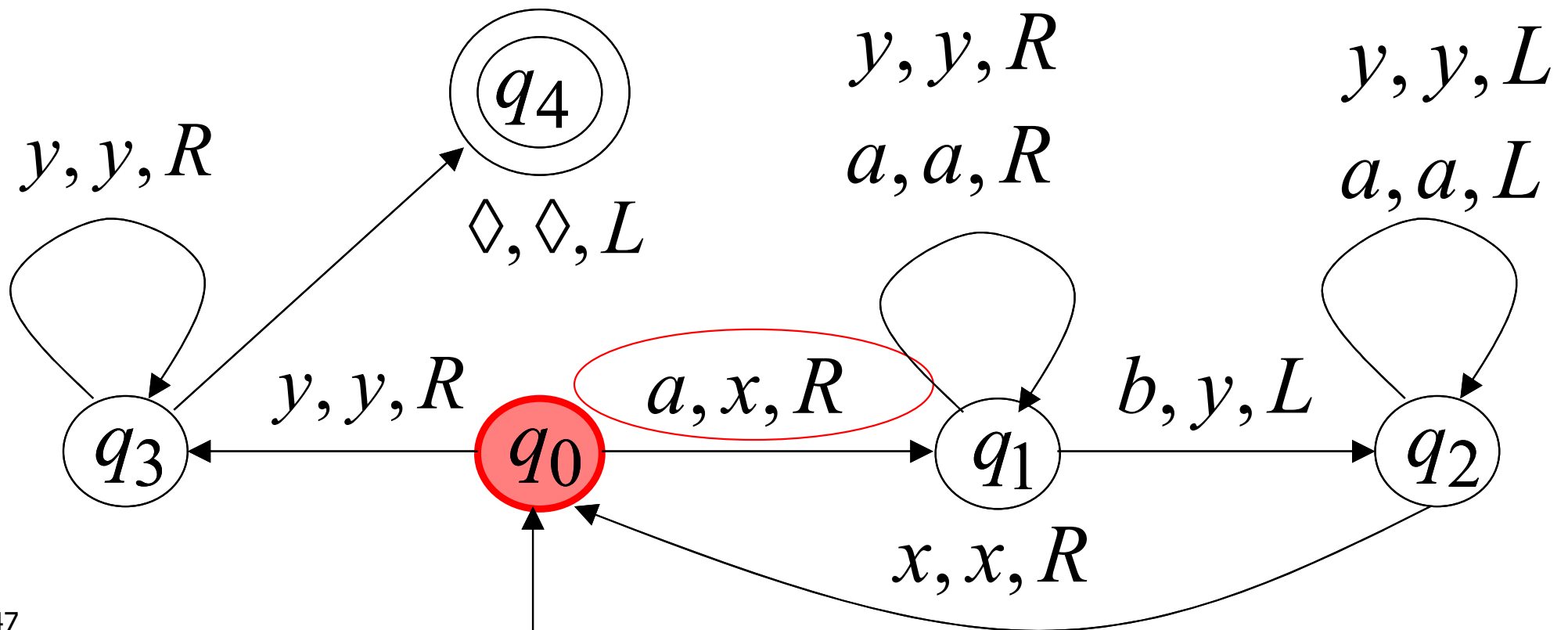
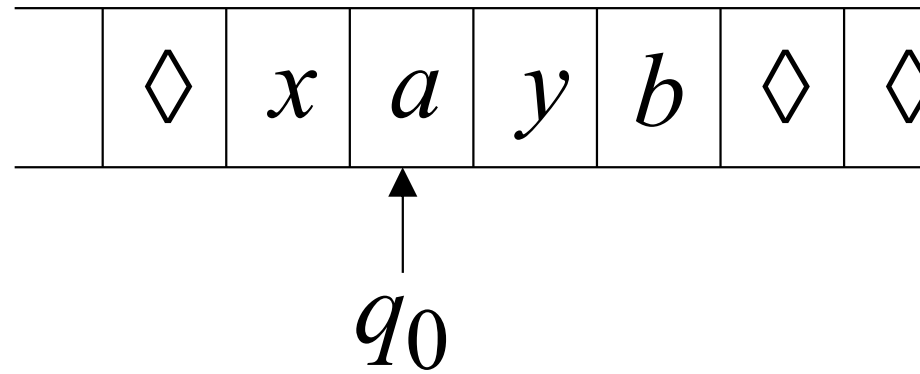
Time 3



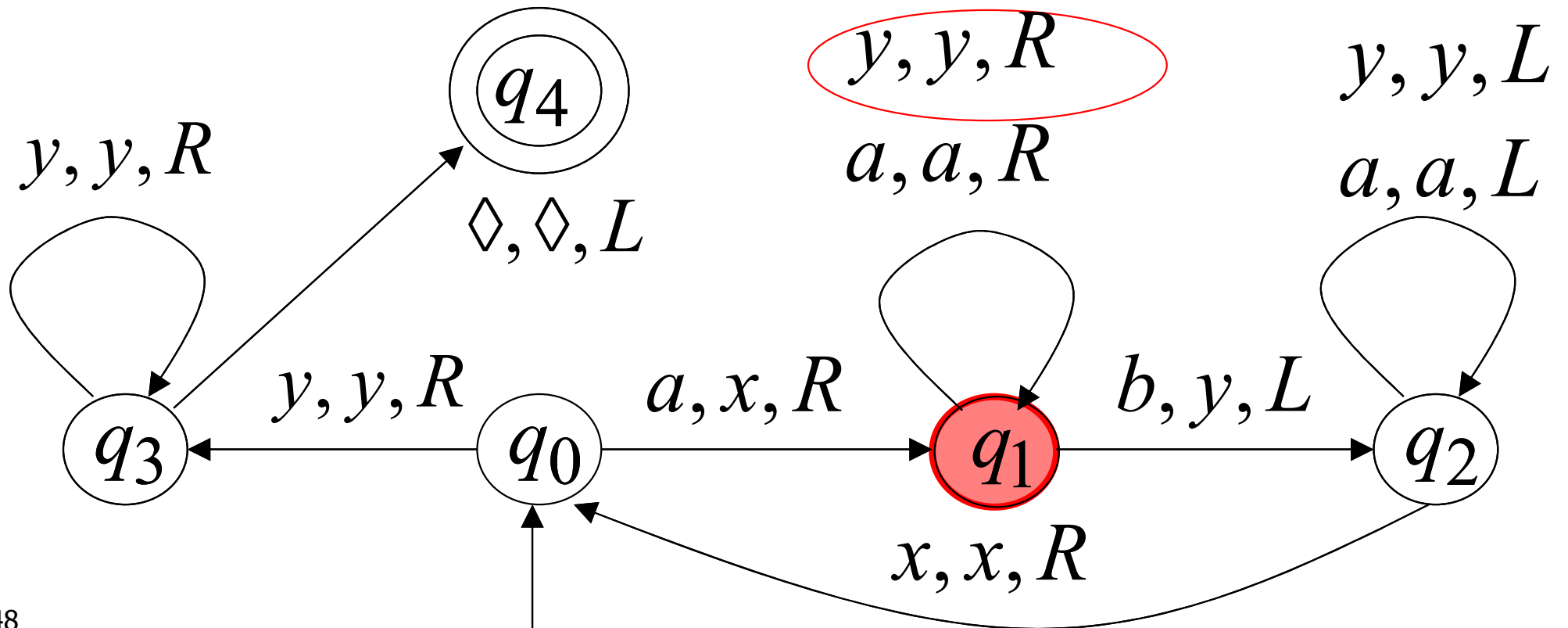
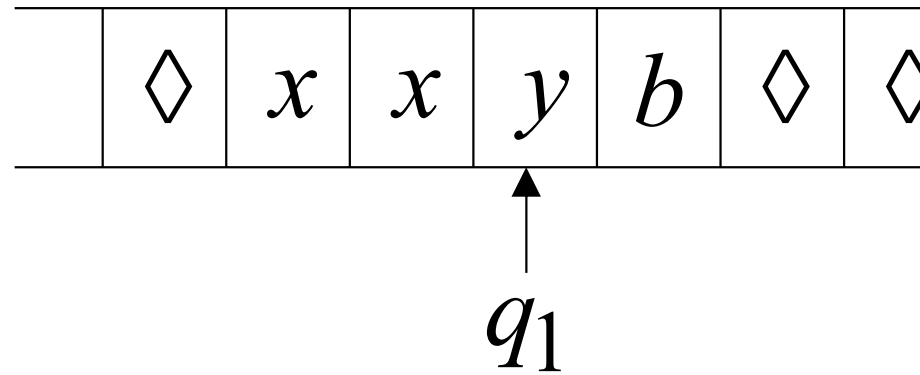
Time 4



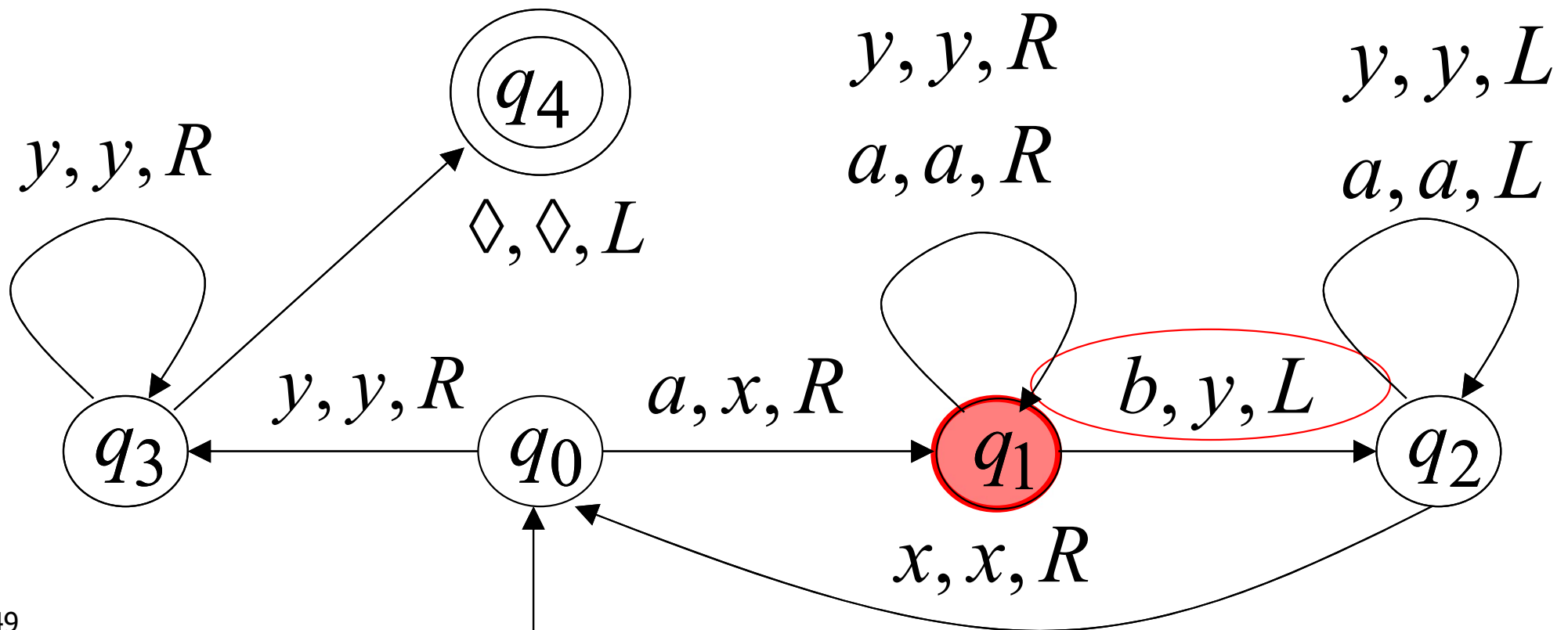
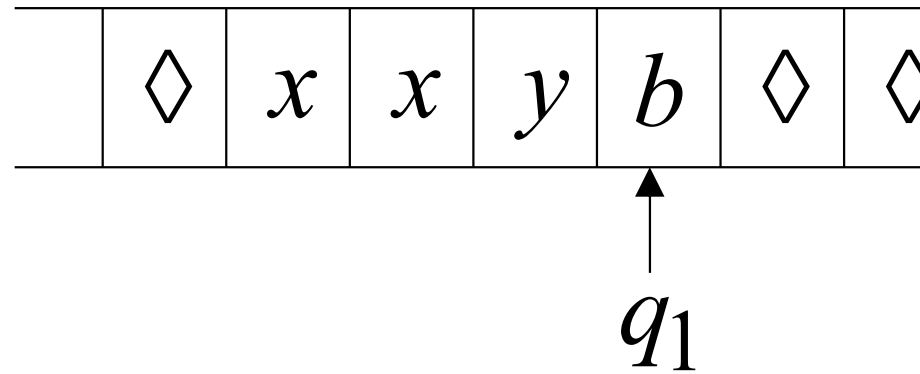
Time 5



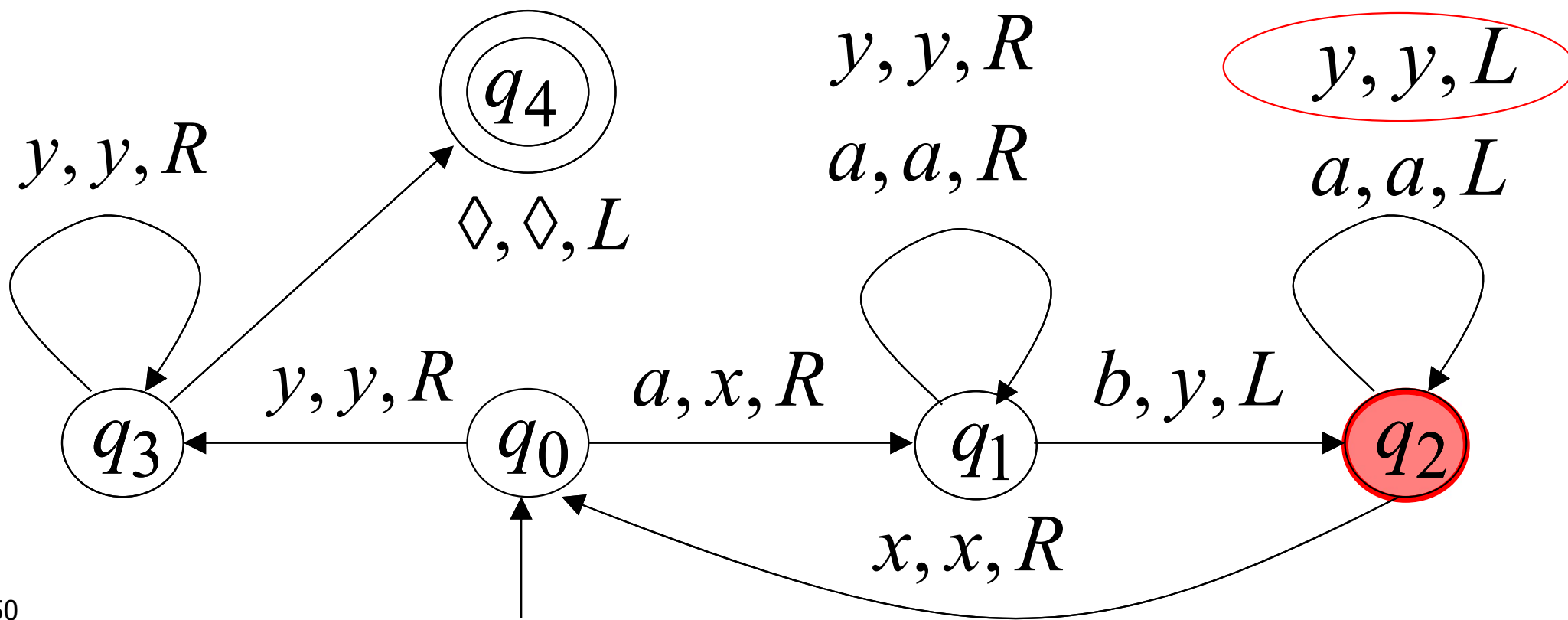
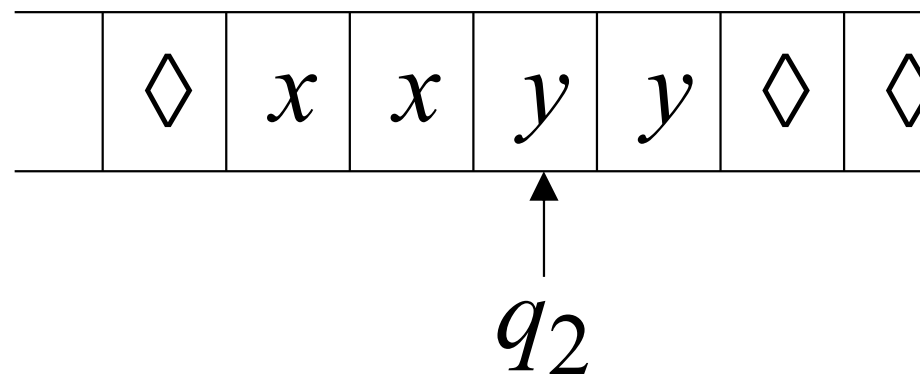
Time 6



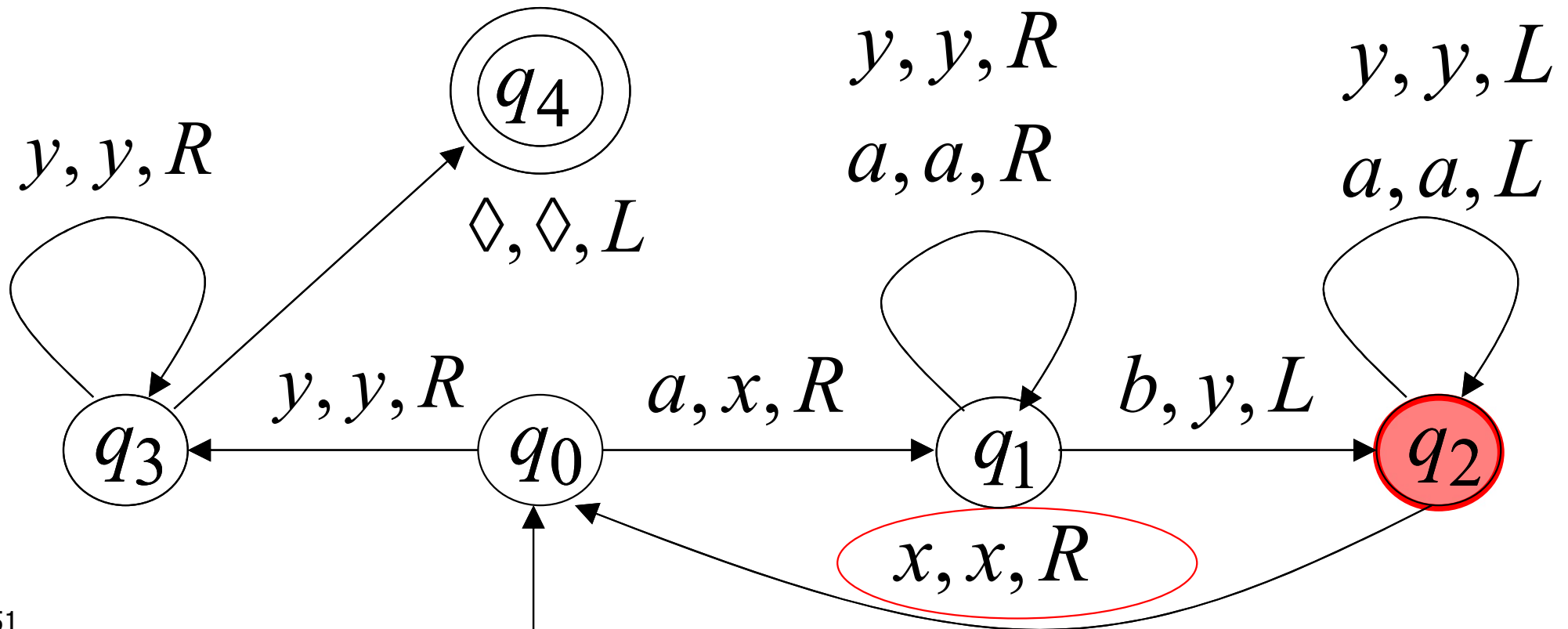
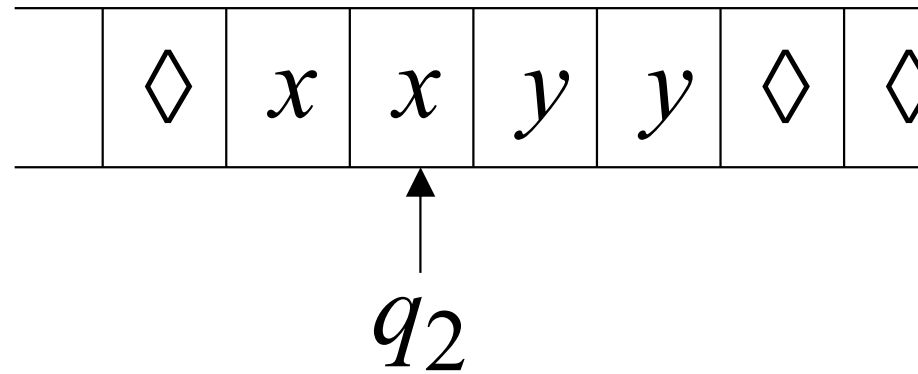
Time 7



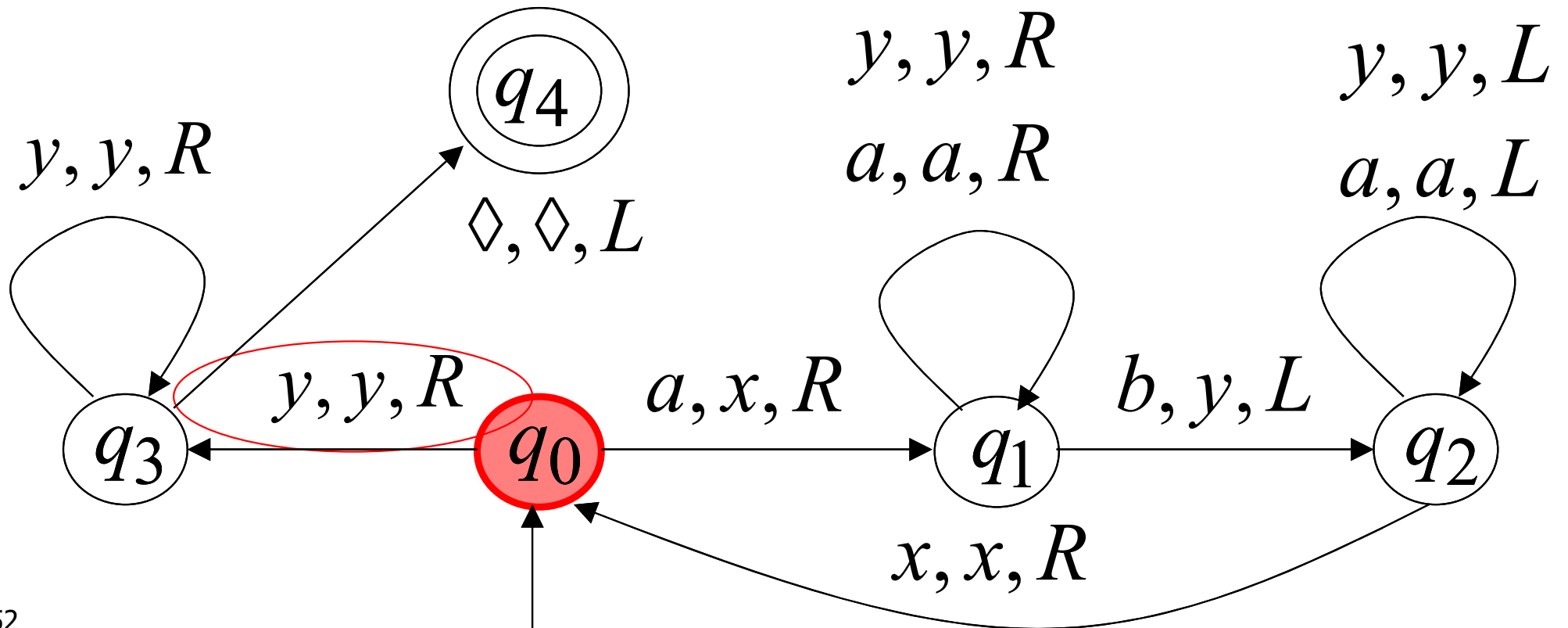
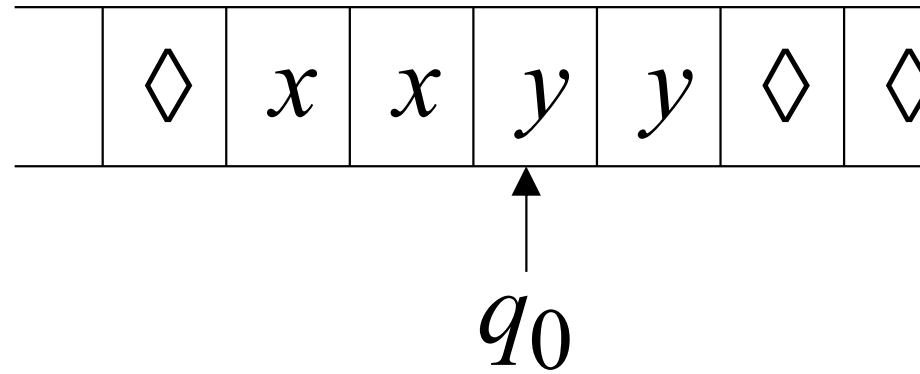
Time 8



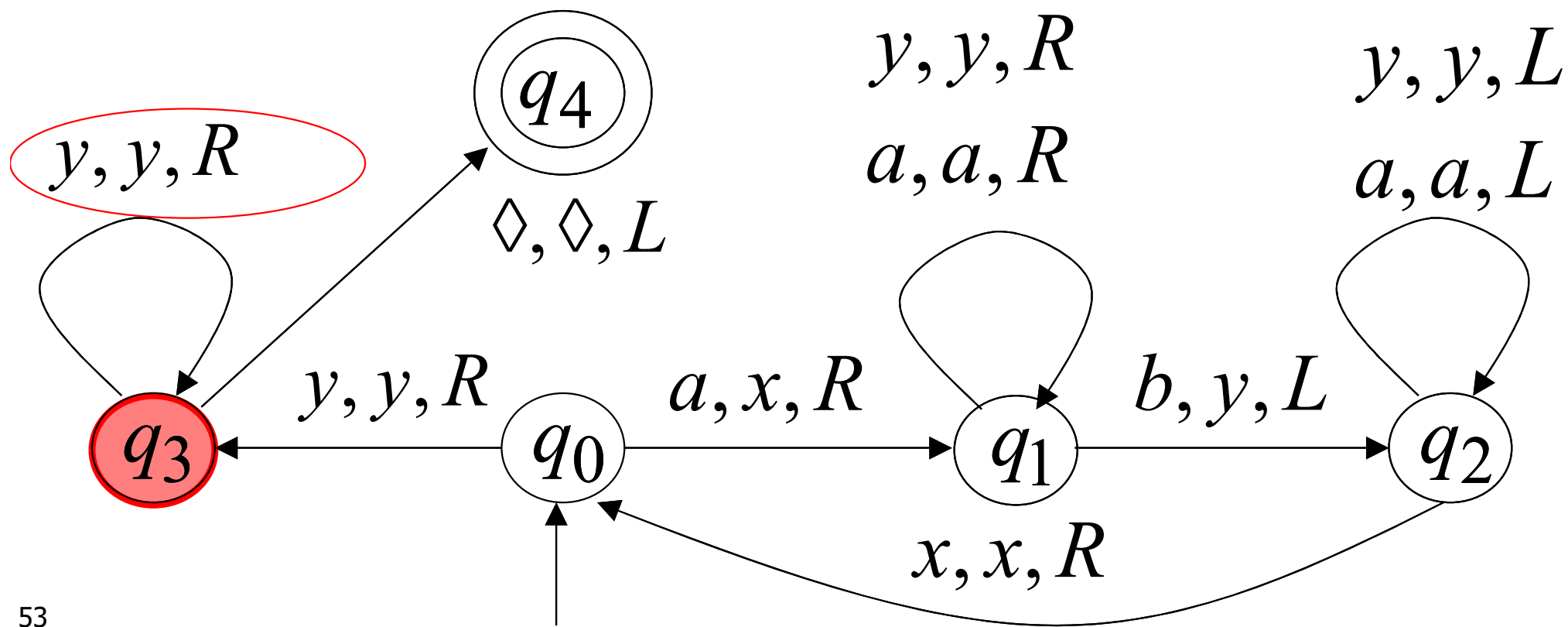
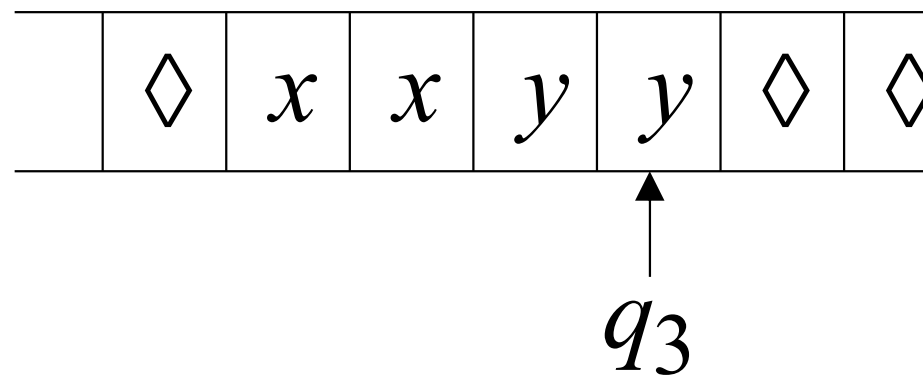
Time 9



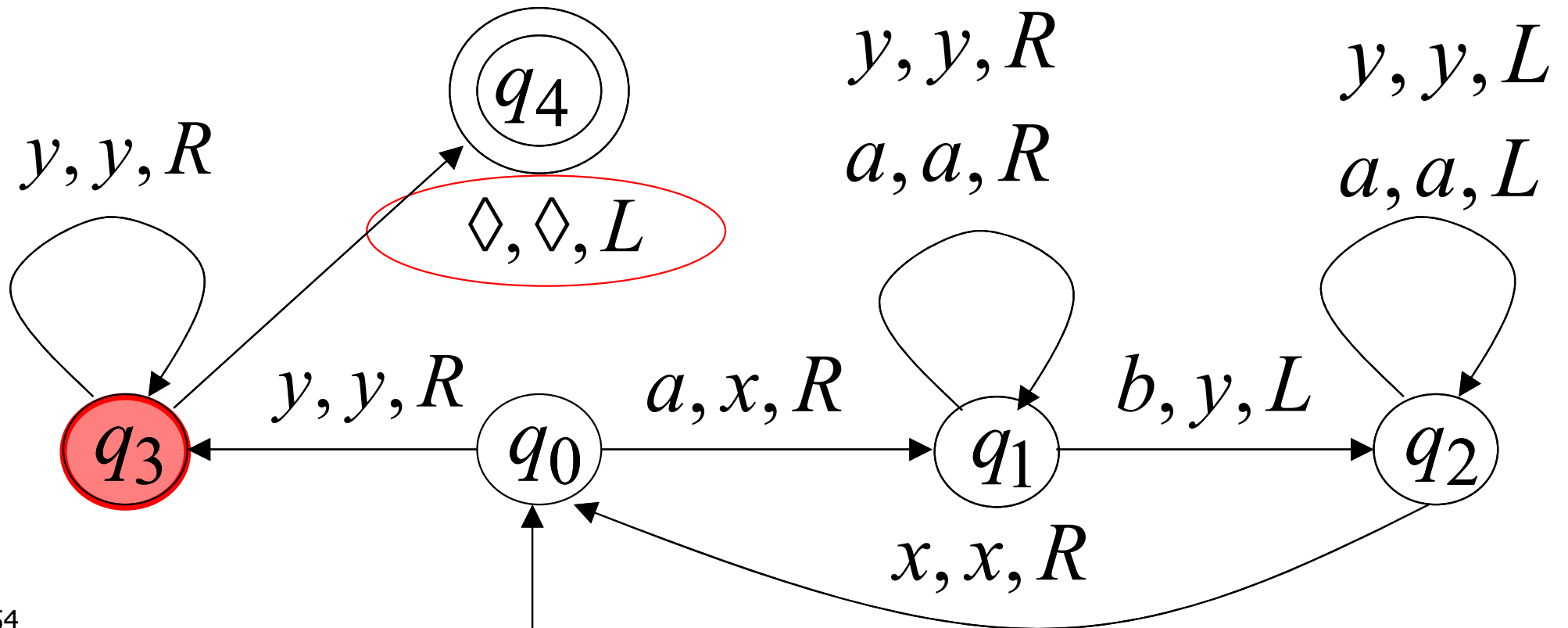
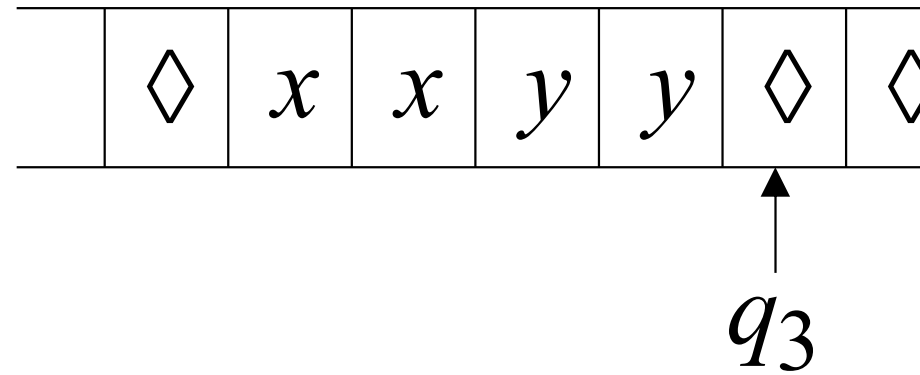
Time 10



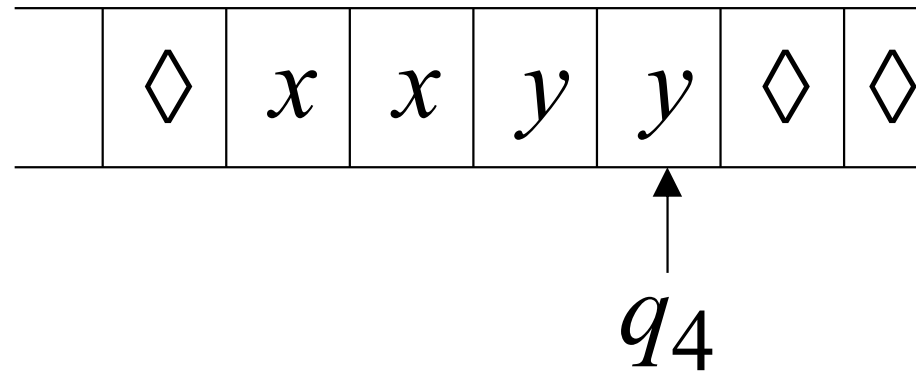
Time 11



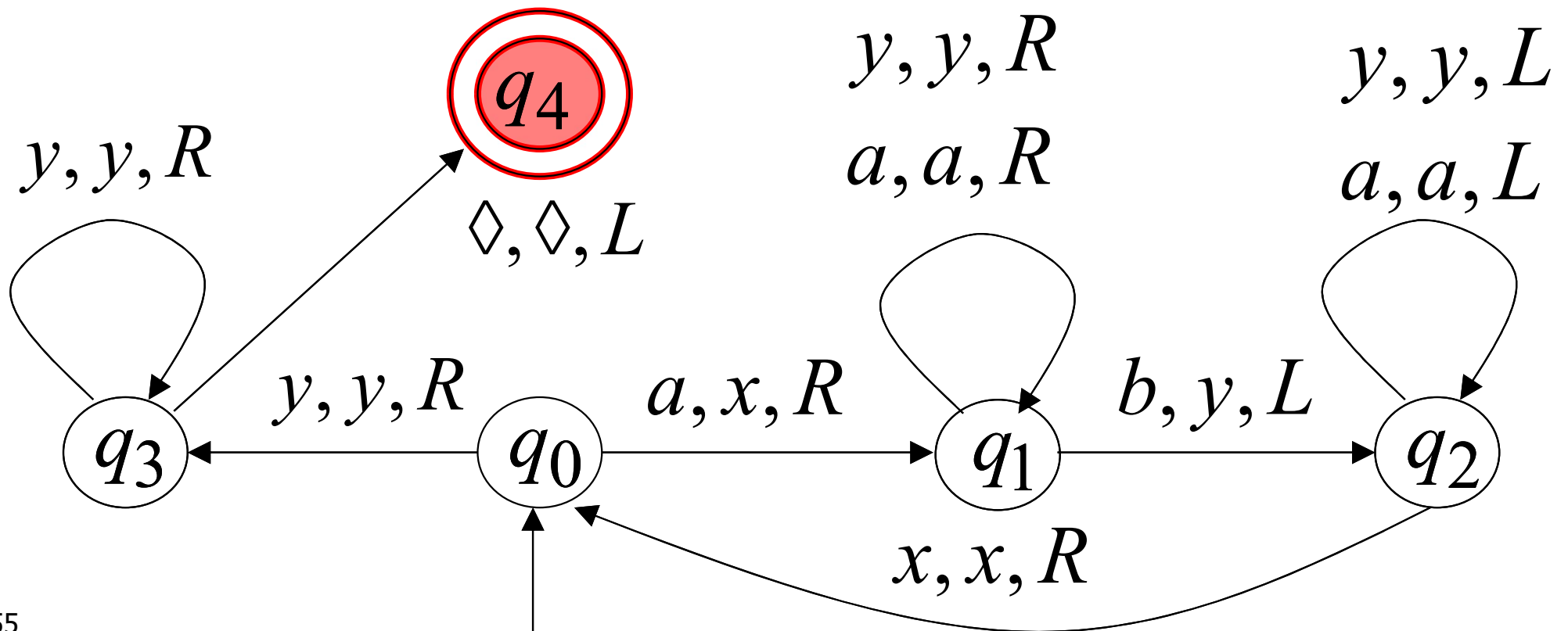
Time 12



Time 13



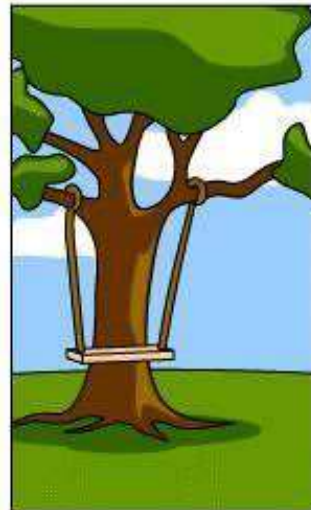
Halt & Accept



你想像中的軟體開發是怎樣？



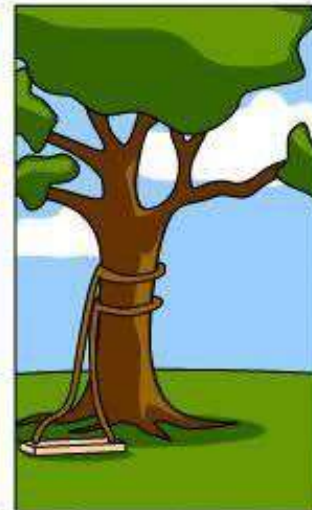
客戶解釋他們想要的



專案主持人對客戶需求的認知



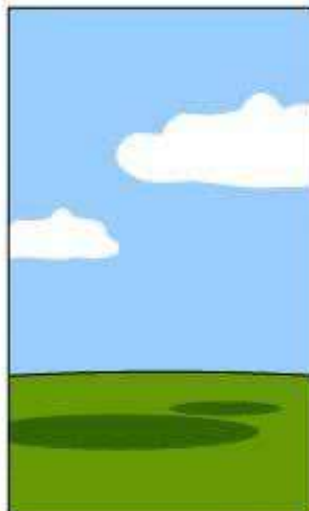
系統分析師所設計的



程式設計師所寫出來的



顧問所描繪的願景



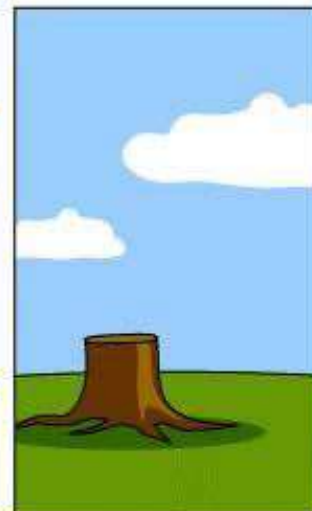
專案的文件



最後交付給客戶的軟體



客戶所付的錢



上線後的技術支援



客戶真正需要的

QA Engineer

Devs watching QA test the product



QA Engineer



History of C

- Created by Dennis Ritchie, AT&T Bell Labs in 1970s
- International standard ISO/IEC9899
- Available on wide range of platforms, from microcontrollers to supercomputers; very few platforms for which C compiler not available
- Procedural, provides language constructs that map efficiently to machine instructions
- Does not directly support object-oriented or generic programming
- Application domains: system software, device drivers, embedded applications, application software
- Greatly influenced development of C++
- When something lasts in computer industry for more than 40years (outliving its creator), must be good

History of C++

- Created by Bjarne Stroustrup, Bell Labs
- Originally C with Classes, renamed as C++ in 1983
- International standard ISO/IEC14882
- Procedural
- Loosely speaking is superset of C
- Directly supports object-oriented and generic programming
- Maintains efficiency of C
- Application domains: systems software, application software, device drivers, embedded software, high-performance server and client applications, entertainment software such as video games, native code for Android applications
- Greatly influenced development of Python, C# and Java

A Sample C++ Program (2/2)

Display 1.1 A Sample C++ Program

```
1  #include <iostream>
2  using namespace std;

3  int main( )
4  {
5      int numberOfLanguages;

6      cout << "Hello reader.\n"
7           << "Welcome to C++.\n";

8      cout << "How many programming languages have you used? ";
9      cin >> numberOfLanguages;

10     if (numberOfLanguages < 1)
11         cout << "Read the preface. You may prefer\n"
12              << "a more elementary book by the same author.\n";
13     else
14         cout << "Enjoy the book.\n";

15     return 0;
16 }
```

A Sample C++ Program (2/2)

SAMPLE DIALOGUE 1

Hello reader.

Welcome to C++.

How many programming languages have you used? 0 ← *User types in 0 on the keyboard.*

Read the preface. You may prefer
a more elementary book by the same author.

SAMPLE DIALOGUE 2

Hello reader.

Welcome to C++.

How many programming languages have you used? 1 ← *User types in 1 on the keyboard.*

Enjoy the book

Days before OpenAI



Days after OpenAI



Any Question?