

อบรมการใช้ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ สำหรับการสร้างเอกสาร วิชาการ

จิตยา หวานวารี

17-18 กันยายน 2565

สารบัญ

สารบัญ	ii
สารบัญรูป	iv
สารบัญตาราง	v
1 การติดตั้ง L^AT_EX	3
1.1 MikT _E X บน Windows	3
1.2 T _E XLive บน Windows	4
1.3 โปรแกรมสำหรับพิมพ์เอกสาร	4
2 L^AT_EX เบื้องต้น	7
2.1 พื้นฐานที่ควรรู้	7
2.2 ช่องว่าง	8
2.3 อักขระพิเศษ	8
2.4 กลุ่ม	9
2.5 environment	9
2.6 คำสั่งเพื่อการเรียงพิมพ์	10
2.7 Comment	11
2.8 ไฟล์ต่างๆ	11
3 โครงสร้างเอกสาร	13
3.1 โครงสร้างของแฟ้มต้นฉบับ	13
3.2 การจัดหน้าทั่วไป	14
3.3 โครงสร้างเนื้อหาในเอกสาร	14
3.4 โครงสร้างเนื้อหาในเอกสาร (ต่อ)	15

4	การจัดรูปแบบ	17
4.1	ขนาดและรูปแบบตัวอักษร	17
4.2	การจัดเรียงข้อความ	19
4.3	ช่องว่างแบบต่างๆ	20
4.4	ข้อย่อย	21
5	การแทรกรูปภาพและตาราง	23
5.1	การแทรกรูปภาพ	23
5.2	การสร้างตาราง	25
6	การอ้างอิง	27
6.1	การอ้างอิงภายในเอกสาร	27
6.2	บรรณานุกรม	27
6.3	เอกสารสำหรับอ่านเพิ่มเติม	29
7	การแทรกสมการและสัญลักษณ์คณิตศาสตร์	31
	บรรณานุกรม	33

สารบัญรูป

1.1	ตัวอย่างหน้าจอของ TeXstudio	4
2.1	โครงสร้างพื้นฐานของแฟ้มต้นฉบับเอกสาร L ^A T _E X	7
3.1	โครงสร้างแฟ้มต้นฉบับเอกสาร L ^A T _E X	13
5.1	ตัวอย่างการแทรกรูปภาพ	23
6.1	ตัวอย่างข้อมูลสำหรับ BibTeX	28

สารบัญตาราง

4.1	การกำหนดขนาดของตัวอักษร	17
4.2	การจัดรูปแบบตัวอักษร	18

เกี่ยวกับเอกสาร

เอกสารนี้ร่างขึ้นโดยอาศัยเนื้อหาจาก <https://en.wikibooks.org/wiki/LaTeX> รหัสต้นฉบับของเอกสารนี้อยู่ที่ <https://www.overleaf.com/read/cgrvbrzpxrtm>

ผู้ที่สนใจสามารถนำไปดัดแปลงใช้ประโยชน์ได้โดยไม่ต้องแสดงที่มา และอนุญาตให้ใช้สิทธิในแบบเดียวกัน

This work is licensed under a Creative Commons
“Attribution-NonCommercial-ShareAlike 3.0 Un-
ported” license.



บทที่ 1

การติดตั้ง L^AT_EX

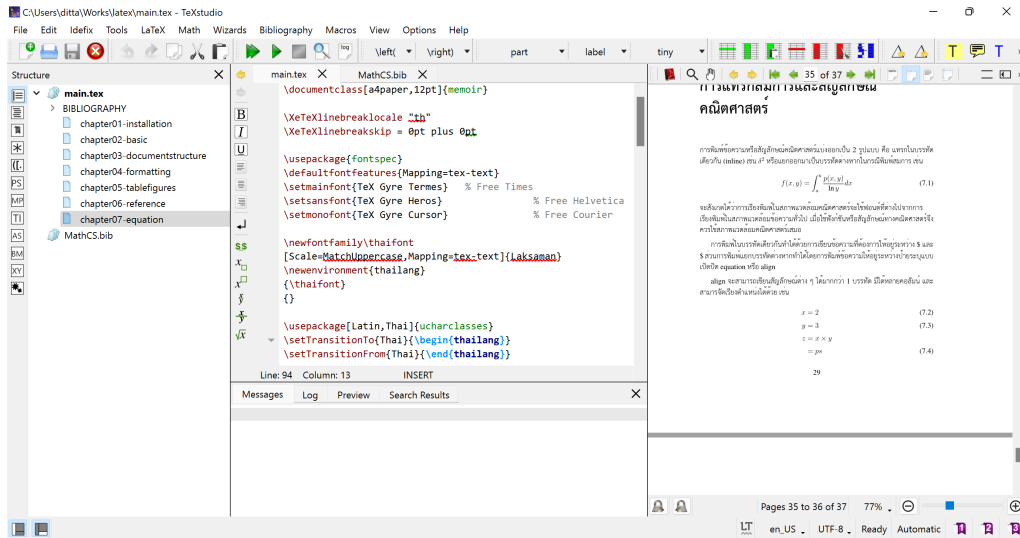
มีแพ็คเกจสำหรับ L^AT_EX หลายค่ายให้เลือกใช้ ทุกค่ายทำงานหลัก ๆ ได้เหมือนกัน แยกแพ็คเกจตามระบบปฏิบัติการหลัก ๆ เรียงลำดับตามความนิยมได้ดังนี้

- Windows: MikT_EX และ T_EXLive
- Linux: T_EXLive
- OS X: MacT_EX, T_EXLive

ในเอกสารนี้จะแนะนำการติดตั้ง MikT_EX และ T_EXLive บนระบบปฏิบัติการ Windows

1.1 MikT_EX บน Windows

1. ดาวน์โหลด MikT_EX จาก <http://miktex.org/download>
 - เลือกแบบ Basic หรือ Net Installer ก็ได้
 - หากความเร็วอินเทอร์เน็ตไม่สูง แนะนำให้เลือกแบบ Basic
 - หากความเร็วอินเทอร์เน็ตสูง แนะนำให้เลือกแบบ Net Installer และลงแบบ Full
2. เมื่อโปรแกรมให้เลือกลงแพ็คเกจเพิ่มเติม แนะนำให้เลือกลงเพิ่มแบบ on-the-fly (ลง package เสริมตามที่จำเป็นเมื่อถูกเรียกใช้)



รูปที่ 1.1: ตัวอย่างหน้าจอของ TeXstudio

1.2 \TeX Live บน Windows

1. ดาวน์โหลดตัวติดตั้ง \TeX Live จาก <http://mirror.ctan.org/systems/texlive/tlnet/install-tl-windows.exe>
2. โปรแกรมจะดาวน์โหลดแพ็คเกจเพื่อการใช้งานลงมาเก็บในเครื่องเพื่อดำเนินการติดตั้ง

1.3 โปรแกรมสำหรับพิมพ์เอกสาร

\LaTeX นั้นเป็นเพียงโปรแกรมเพื่อการเรียงพิมพ์เท่านั้น การเขียนบทความหรือเอกสารต่างๆ สามารถใช้โปรแกรมสำหรับพิมพ์ข้อความใด ๆ ก็ได้ เช่น Notepad จัดเก็บไฟล์เป็นข้อความธรรมดา (plain text) จากนั้นจึงนำเอกสารที่พิมพ์แล้วมาส่งให้โปรแกรม \LaTeX เรียงพิมพ์ให้ แต่หากใช้โปรแกรมสำหรับพิมพ์ที่มีความสามารถในการเน้นคำเฉพาะที่ใช้ในเอกสารแบบ \LaTeX รวมถึงมีตัวเรียงพิมพ์ และตัวแสดงผลในตัวโปรแกรมด้วย ก็จะช่วยให้การทำงานสะดวกขึ้น

โปรแกรมเหล่านี้เทียบได้กับ IDE (Integrated Development Environment) ในการเขียนโปรแกรมสำหรับ \LaTeX IDE ที่แนะนำในการเขียน \LaTeX คือ TeXstudio ซึ่งมีให้ใช้ทั้งใน Windows, OS X และ Linux สามารถดาวน์โหลดได้จาก <http://>

1.3. โปรแกรมสำหรับพิมพ์เอกสาร

5

texstudio.sourceforge.net/ ตัวอย่างหน้าจอของ TeXstudio เป็นดังรูปที่

1.1

โปรแกรมอื่นที่สะดวกและใช้งานง่ายเช่น TeXWorks, LEd เป็นต้น

บทที่ 2

L^AT_EX เบื้องต้น

เนื้อหาต่อไปนี้นำมาจาก <https://en.wikibooks.org/wiki/LaTeX> ซึ่งเป็นหนังสือสอนการใช้ L^AT_EX และสามารถค้นเพื่อใช้อ้างอิงได้

2.1 พื้นฐานที่ควรรู้

เนื่องจากการใช้ L^AT_EX สร้างเอกสาร คือการเขียนคำสั่งเพื่อการเรียงพิมพ์ การสร้างเอกสารก็จะคล้ายๆ กับการเขียนโปรแกรม ซึ่งจำเป็นต้องรู้โครงสร้างไวยากรณ์ต่างๆ ที่ควบคุมให้เกิดการแสดงผลตามต้องการ

โครงสร้างเบื้องต้นที่สุดของ L^AT_EX เป็นดังรูปที่ 2.1 การอธิบายโครงสร้างของรหัสต้นแบบ จะอยู่ในบทที่ 3

```
1 \documentclass{article}
2 \begin{document}
3 My text
4 \end{document}
```

รูปที่ 2.1: โครงสร้างพื้นฐานของแฟ้มต้นฉบับเอกสาร L^AT_EX

2.2 ช่องว่าง

ช่องว่างในรหัสต้นฉบับไม่ว่าจะเป็นการย่อหน้า tab เคาะวรรคไม่ว่าจะกี่ครั้งก็ตาม จะให้ผลเหมือนกันทั้งสิ้น ในทำนองเดียวกับการเขียนโปรแกรมอื่นๆ

การขึ้นบรรทัดใหม่เพียงครั้งเดียว จะให้ผลเหมือนกับการเคาะวรรค ซึ่งมีประโยชน์ในการทำให้อ่านต้นฉบับง่ายขึ้น โดยเฉพาะอย่างยิ่ง เมื่อเขียนคำสั่งยาวๆ หรือบางครั้ง อาจเขียนเนื้อหาบรรทัดละประโยค เพื่อให้สะดวกในการปรับแก้ข้อความก็ได้เช่นกัน

แต่การขึ้นบรรทัดใหม่สองครั้งในรหัสต้นฉบับจะเป็นการสั่งให้ขึ้นย่อหน้าใหม่ในการเรียงพิมพ์

2.3 อักขระพิเศษ

อักขระต่อไปนี้ใช้สำหรับไวยากรณ์ต่างๆ

```
# $ % ^ & _ { } ~ \
```

ความหมายของอักขระพิเศษต่างๆ มีดังนี้

- วงเล็บปีกกา ({ , }) ใช้ระบุขอบเขตของสภาพแวดล้อมต่าง ๆ เช่นเดียวกับโครงสร้างบล็อกในการเขียนโปรแกรม
- เครื่องหมายดอลลาร์ (\$) ใช้ล้อมสภาพแวดล้อมแบบคณิตศาสตร์สำหรับการเขียนในบรรทัดเดียวกับข้อความ เช่น $2x + 1$
- เครื่องหมายชาร์ป (#) ใช้ระบุลำดับที่ของตัวแปรสำหรับมาโคร
- เครื่องหมายและ (&) ใช้เป็นตัวระบุขอบเขตคอลัมน์ในตาราง

หากต้องการให้แสดงอักขระเหล่านี้ในฐานะอักขระปกติ จะต้องตัดความเป็นอักขระพิเศษ (escape) ด้วยการใช้นำหน้า ดังนี้

```
\# \$ \% \^{} \& \_ \{ \} \~{} \textbackslash{}
```

บางครั้ง การใช้นำหน้า ก็อาจจะไม่ได้เป็นการตัดความเป็นอักขระพิเศษเสมอไป กรณีที่พบบ่อย เช่น กรณีดังต่อไปนี้

- \\ ไม่ได้เป็นการตัดความหมายพิเศษของอักขระ \ ออก แต่หมายถึงการขึ้นบรรทัดใหม่เมื่อเรียงพิมพ์

- `\textasciitilde` ใช้เมื่อต้องการสัญลักษณ์ `~` แต่ไม่ใช่ `\~` ตรงๆ เนื่องจาก `\~a` นั้นใช้สำหรับการใส่ `tilda` ลงเหนืออักขระ เช่น `ã` ส่วน `~` เปร่าๆ นั้นหมายถึงการสร้างช่องไฟขนาด 1 ตัวอักษร
- `\textasciicircum` ใช้เมื่อต้องการสัญลักษณ์ `^` แต่ไม่ใช่ `\^` ตรงๆ เนื่องจาก `\^a` นั้นใช้สำหรับการใส่ `circumflex` ลงเหนืออักขระ เช่น `â`

การจดจำวิธีพิมพ์ทั้งหมดอาจเป็นไปได้ยาก การทดลองเรียงพิมพ์แล้วตรวจสอบผลลัพธ์ดู หากผิดปกติก้อยไปหาวิธีในคู่มือหรืออินเทอร์เน็ตดูจะเหมาะสมกับการใช้งานจริงมากกว่า

2.4 กลุ่ม

คำสั่งบางประเภท เช่น การสั่งให้เปลี่ยนแบบอักษรเป็นตัวหนา จะมีผลตั้งแต่เริ่มสั่งเป็นต้นไป แต่เราอาจจะไม่ได้ต้องการให้มีผลยาวไปจนจบขอบเขตของเอกสาร ดังนั้น เราสามารถสร้างกลุ่มของข้อความขึ้นมาได้ โดยใช้วงเล็บปีกการครอบข้อความที่ต้องการ แล้วใช้คำสั่งต่างๆ ภายในวงเล็บปีกการนั้น ผลของคำสั่งจะมีผลเฉพาะในกลุ่มเท่านั้น เช่น

ทดสอบ **ตัวหนา** *ตัวเอียงหนา*

และบางคำสั่งที่มีผลเฉพาะอักขระตัวถัดไป ก็สามารถให้มีผลเป็นกลุ่มได้ด้วยการรวมอักขระไว้ในวงเล็บปีกกาให้เป็นกลุ่ม เช่น

อันนี้มีผล แค่ตัวเดียว แต่ถ้า **แบบนี้มีผลทั้งกลุ่ม** *ตัวเอียงแบบนี้* ก็เช่นกัน

2.5 environment

การใช้วงเล็บปีกกานั้นเหมาะกับข้อความสั้นๆ และเป็นการกำหนดขอบเขตแบบง่ายๆ การกำหนดขอบเขตที่กว้างขึ้น และมีรายละเอียดภายในมากขึ้น จะเรียกว่าการสร้าง `environment` โดยมีคำสั่งหลายประเภทที่ทำงานภายใน `environment` เช่น การสร้าง `quote`

ข้อความนี้อยู่ใน `quote`

การสร้าง `quote` ในตัวอย่างคือการเรียกใช้ชุดคำสั่งหลายอย่างประกอบกัน ได้แก่ การเว้นระยะบนล่างจากย่อหน้าปกติ การเว้นย่อหน้าลึกเข้ามากว่าข้อความปกติ เป็นต้น

เมื่อจบ `environment` แล้ว การเรียงพิมพ์ต่างๆ จะกลับมาสู่สถานะเดิมก่อนที่จะเข้าไปใน `environment`

2.6 คำสั่งเพื่อการเรียงพิมพ์

คำสั่งพิเศษเพื่อการเรียงพิมพ์ของ *TEX* จะอยู่ในป้ายระบุ (`tag`) โดยโปรแกรมจะอ่านป้ายระบุแล้วเรียงพิมพ์ตามคำสั่งของป้ายระบุนั้นๆ และเป็นสาเหตุให้ *TEX* ไม่สามารถแสดงผลได้ทันทีหากไม่สั่งเรียงพิมพ์ (`typeset`) เสียก่อน

ป้ายระบุเป็นคำสั่งพิเศษซึ่งอยู่หลังเครื่องหมาย `backslash` (`\`) เราจะเรียกชื่อป้ายระบุตามคำสั่งที่อยู่หลังเครื่องหมาย `backslash` ป้ายระบุแบ่งคร่าว ๆ ได้ 3 รูปแบบ ดังนี้

1. **ป้ายระบุแบบใช้ตัวแปร:** โครงสร้างของป้ายระบุแบบกำหนดขอบเขตคือ

```
\tagname [options] {text}
```

ความหมายของโครงสร้างป้ายระบุชนิดนี้คือใช้คำสั่ง (`tagname`) โดยให้อ่านค่าตัวแปรที่อยู่ภายในขอบเขตวงเล็บปีกกา (`text`) หากมีตัวแปรมากกว่าหนึ่งตัว จะใส่ค่าตัวแปรในวงเล็บปีกกาเรียงต่อไปเรื่อย ๆ

2. **ป้ายระบุแบบไม่ใช้ตัวแปร:** โครงสร้างของป้ายระบุชนิดนี้คือ

```
\tagname
```

ป้ายระบุชนิดนี้มีการใช้งานสองรูปแบบ แบบหนึ่งคือเป็นคำสั่งในการเรียงพิมพ์และใช้คำสั่งนี้ต่อไปจนกว่าจะสิ้นสุดขอบเขตของเอกสารหรือสิ้นสุดบล็อกปัจจุบัน เช่น คำสั่งเพิ่มขนาดอักษร (`\Large`) การใช้งานอีกรูปแบบคือเป็นคำสั่งเฉพาะ เช่น คำสั่งสร้างหัวเรื่อง (`\maketitle`)

3. **ป้ายระบุเปิด/ปิด:** โครงสร้างของป้ายระบุชนิดนี้คือ

```
\begin{tagname} [options] text \end{tagname}
```

ป้ายระบุชนิดนี้มีการใช้งานสองรูปแบบเช่นกัน แบบหนึ่งคือเรียงพิมพ์ข้อความที่อยู่ระหว่าง `begin` และ `end` ด้วยคำสั่ง (`tagname`) นั้นๆ เช่น การจัดข้อความ

ให้อยู่กึ่งกลางหน้ากระดาษ อีกแบบเป็นการกำหนดขอบเขตของสภาพแวดล้อม เช่นในรูปที่ 3.1 บรรทัดที่ 9-13 หมายถึงข้อความทั้งหมดที่อยู่ระหว่าง `begin` และ `end` เป็นเนื้อหาของเอกสาร (document)

ป้ายระบุแบบที่ 1 และแบบที่ 3 สามารถตั้งค่าเพิ่มเติม (option) ได้ โดยกำหนดค่าต่างๆ ลงในวงเล็บสี่เหลี่ยม และคั่นด้วย , หากมีค่าเพิ่มเติมมากกว่าหนึ่งตัว

2.7 Comment

ในการเขียนต้นฉบับ บางครั้งเราอาจต้องการทดแนวกคิด หรือข้อมูลบางอย่างเอาไว้ หรือบางครั้งก็อาจจะต้องการปรับข้อความให้สั้นลงโดยตัดบางประโยคออก แต่ยังไม่อยากลบทิ้ง เพราะอาจจะพิจารณาเก็บมาใช้ใหม่อีกครั้ง เราสามารถใช้การ `comment` ข้อความหรือแม้แต่คำสั่งเหล่านี้ออกได้

เมื่อโปรแกรมพบ `%` ในรหัสต้นฉบับ ข้อความหลังจาก `%` ทั้งหมดจะถูกตัดเป็น `comment` ทั้งหมด และไม่นำมาเรียงพิมพ์

ในการ `comment` หลายๆ บรรทัดนั้น การค่อยๆ พิมพ์ `%` ที่ละบรรทัดอาจจะไม่ค่อยสะดวก `text editor` ส่วนใหญ่จึงอำนวยความสะดวกให้โดยมี `shortcut` ในการ `comment` เมื่อ `mark` บรรทัดทั้งหมดที่ต้องการ และสามารถกดซ้ำอีกครั้งเพื่อ `uncomment` หรือนำข้อความกลับมาได้ การใช้ `comment/uncomment` แบบนี้จะสะดวกในการพิมพ์ และทดสองเรียงพิมพ์เพื่อดูผลลัพธ์อย่างรวดเร็ว

นอกจากการใช้ `%` แล้ว ยังมี `environment comment` ที่ใช้ครอบข้อความทั้งหมดที่ไม่ต้องการให้เรียงพิมพ์ได้ด้วย

2.8 ไฟล์ต่างๆ

การเรียงพิมพ์ใน `LATEX` มักจะไม่จบในขั้นตอนเดียว โดยทั่วไปแล้ว เราจะต้องสั่งเรียงพิมพ์ 2-3 ครั้ง จึงจะได้เอกสารที่สมบูรณ์ ไฟล์นามสกุลต่างๆ ที่สำคัญในการเรียงพิมพ์มีดังนี้

- `.tex` เป็นไฟล์ต้นฉบับของเอกสาร
- `.bib` เป็นไฟล์เอกสารอ้างอิง ซึ่งอยู่ในรูปแบบ `BibTEX`

- `.cls` เป็นไฟล์คลาสของเอกสาร กำหนดรูปแบบ และคำสั่งต่างๆ ที่ใช้ในเอกสาร หากใช้คลาสมาตรฐานก็ไม่จำเป็นต้องมีไฟล์นี้ แต่วารสาร หรือการประชุมที่สังกัดองค์กรใหญ่ๆ อาจจะมีคลาสของเอกสารเฉพาะให้ใช้สำหรับเรียงพิมพ์
- `.sty` เป็นไฟล์กำหนดรูปแบบของเอกสาร มักรวมคำสั่งต่างๆ เอาไว้ `template` ของการประชุมวิชาการมักจะใช้ `style file` ของการประชุมนั้นๆ มา เพื่อให้สะดวกในการจัดหน้าเพื่อรวมเป็นรูปเล่ม
- `.bst` เป็นไฟล์กำหนดรูปแบบของเอกสารอ้างอิง ซึ่งไม่จำเป็นต้องใช้หากใช้รูปแบบมาตรฐานทั่วไป
- `.dvi` เป็นเอกสารผลลัพธ์ของการเรียงพิมพ์ด้วย *TEX* สามารถแปลงเป็น `postscript` ที่ใช้กับเครื่องพิมพ์ได้ อย่างไรก็ตาม การใช้ *XeTeX* เพื่อเรียงพิมพ์นั้นจะไม่ได้ผลลัพธ์ออกมาเป็น `dvi` แต่เป็น `pdf` แทน

ไฟล์อื่นๆ เช่น `.aux` `.bbl` `.lot` `.log` `.toc` `.out` นั้นเป็นผลระหว่างการเรียงพิมพ์ เก็บข้อมูลต่างๆ ที่จำเป็นในการเรียงพิมพ์ให้สวยงาม โดยปกติแล้ว การเรียงพิมพ์จะเริ่มจากการสร้างไฟล์ประกอบเหล่านี้ขึ้นมาก่อน แล้วจึงเรียงพิมพ์เอกสารผลลัพธ์โดยอาศัยข้อมูลจากไฟล์ประกอบเหล่านี้ เป็นเหตุให้เราต้องสั่งเรียงพิมพ์ 2-3 รอบจึงจะได้เอกสารที่สมบูรณ์

บทที่ 3

โครงสร้างเอกสาร

3.1 โครงสร้างของแฟ้มต้นฉบับ

ในเอกสารต้นฉบับสำหรับการเรียงพิมพ์ด้วย $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ นั้น นอกจากจะมีข้อความที่ต้องการให้แสดงผลแล้ว ยังมีคำสั่งพิเศษต่าง ๆ ที่ใช้ในการเรียงพิมพ์ด้วย

```
1 \documentclass[a4paper,twocolumn]{article}
2 % preamble
3 \usepackage{fontspec}
4
5 \title{My first article}
6 \author{My Name}
7
8 %document
9 \begin{document}
10     \maketitle
11
12     My text
13 \end{document}
```

รูปที่ 3.1: โครงสร้างแฟ้มต้นฉบับเอกสาร $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

โดยทั่วไป แฟ้มต้นฉบับจะประกอบด้วยสองส่วนหลัก คือ ส่วนก่อนเริ่มต้นเอกสาร (preamble) และส่วนที่เป็นเอกสาร (document) ดังรูปที่ 3.1 โดย

- ส่วนที่เป็นเอกสารคือทั้งหมดที่อยู่ระหว่าง `\begin{document}` และ `\end{`

document } ซึ่งคือบรรทัดที่ 8-13

- และสิ่งที่อยู่ก่อนหน้านั้นทั้งหมด (บรรทัดที่ 1-7) เรียกว่า ส่วนก่อนเริ่มต้นเอกสาร

ส่วนก่อนเริ่มต้นเอกสารจะเป็นตัวกำหนดการตั้งค่าเอกสารที่ไม่ขึ้นกับเนื้อหาเฉพาะจุด เช่น แบบอักษร ตำแหน่งของหัวกระดาษและท้ายกระดาษ ขนาดกระดาษ ซึ่งค่าเหล่านี้ใช้ร่วมกันทั้งเอกสาร และเป็นส่วนสำหรับประกาศการใช้แพ็คเกจต่าง ๆ ที่จะถูกเรียกใช้ในเอกสาร เพื่อให้โปรแกรมเรียงพิมพ์สามารถเรียกใช้ได้อย่างถูกต้องเมื่อทำงาน

3.2 การจัดหน้าทั่วไป

การใช้รูปแบบเอกสารต่าง ๆ ที่กำหนดไว้แล้ว

ป้ายระบุ `\documentclass` ที่บรรทัดแรกของเอกสารจะเป็นตัวกำหนดคลาส (class) หรือรูปแบบสำหรับการเรียงพิมพ์ คลาสมาตรฐานที่มีมาให้เมื่อลงโปรแกรมได้แก่ `article`, `report`, `plain` แต่วารสารและการประชุมวิชาการต่าง ๆ รวมถึงวิทยานิพนธ์ของมหาวิทยาลัยจะกำหนดรูปแบบเฉพาะในการเรียงพิมพ์เอาไว้ เช่น `ieeetran`, `amsart`

รูปแบบการเรียงพิมพ์ เช่น การตั้งกั้นหน้า การขึ้นย่อหน้า ขนาดตัวอักษร รูปแบบตาราง การใส่ชื่อรูปภาพและตาราง จะกำหนดไว้ในแฟ้มที่มีนามสกุล `cls` (class) ซึ่งสิ่งตีพิมพ์ต่าง ๆ จะสร้างมาให้แล้วและให้ดาวน์โหลดไปใช้ได้ วิธีการใช้งานรูปแบบเหล่านี้ทำได้โดยกำหนดป้ายระบุ `documentclass` ให้เป็นชื่อรูปแบบแฟ้มที่มีนามสกุล `cls` นั้นๆ

3.3 โครงสร้างเนื้อหาในเอกสาร

ตอนต้นของส่วนเนื้อหา (document) มักจะเป็นข้อมูลชื่อเรื่อง (title) ชื่อผู้แต่ง (author) วันที่ (date) ของเอกสารนั้นๆ `template` ส่วนมากจะให้ผู้ใช้เขียนเอกสารระบุข้อมูลเหล่านี้ แล้วนำไปเรียงพิมพ์ให้ด้วยคำสั่ง `\maketitle`

ถัดจากส่วนหัวข้อ มักจะเป็นส่วนบทคัดย่อ (abstract) ซึ่งระบุด้วย environment `abstract`

และหลังจากนั้น จึงเป็นเนื้อหาจริงๆ หากเป็นหนังสือหรือรายงานจะแบ่งเป็นบท (chapter) หากเป็นบทความวิจัย มักจะเริ่มต้นที่ตอน (section) และแยกย่อยลงไปเป็นตอนย่อย (subsection) และอาจมีตอนย่อยลงไปอีก (subsubsection) เราสามารถ

กำหนดข้อความต่างๆให้เป็นชื่อเรื่อง ชื่อผู้แต่ง ชื่อบท ชื่อตอน ได้ โดยใช้ป้ายระบุแบบกำหนดขอบเขต ดังนี้

- `\title{...}`
- `\author{...}`
- `\chapter{...}`
- `\section{...}`
- `\subsection{...}`
- `\subsubsection{...}`

ลำดับเลขของส่วนต่าง ๆ จะถูกกำหนดขณะเรียงพิมพ์เรียงลำดับตามเนื้อหา ไม่จำเป็นต้องกำหนดตัวเลขเอง

สำหรับการทำสารบัญนั้น หากใช้โครงสร้างเนื้อหาด้วยป้ายระบุต่าง ๆ แล้ว การสร้างสารบัญทำได้โดยป้ายระบุ `\tableofcontents` รูปแบบของหน้าสารบัญถูกกำหนดโดยคลาส ส่วนสารบัญภาพและตารางทำได้โดยเติมป้ายกำกับภาพและตารางทั้งหมด แล้วสร้างโดยใช้ป้ายระบุ `\listoftables` และ `\listoffigures` ตามลำดับ

หัวข้อไม่นับเลข

ทั้งนี้ เราสามารถระบุโครงสร้างแต่ไม่นับเลขได้ด้วยการใส่ * หลังป้ายกำกับคำสั่งระบุระดับหัวข้อระดับนี้จะไม่ปรากฏในสารบัญ

จะเห็นว่า หัวข้อนี้มีรูปแบบเดียวกับ section เว้นแต่ที่ไม่มีเลขลำดับ และหัวข้อถัดไป จะนับเลขต่อจากหัวข้อก่อนหน้า โดยนับข้ามหัวข้อนี้ไป

3.4 โครงสร้างเนื้อหาในเอกสาร (ต่อ)

โครงสร้างหนังสือ

หนังสือมักจะแบ่งส่วนเอกสารออกเป็นส่วนหน้า (frontmatter) ส่วนเนื้อหา (mainmatter) และส่วนหลัง (backmatter) การแบ่งเหล่านี้มักจะอำนวยความสะดวกในการนับเลขหน้าเพื่อไปใส่ในสารบัญด้วย โดยส่วนหน้าและส่วนหลังมักจะไม่นับเลขหน้า หรือหากนับ ก็จะไม่แยกจากส่วนเนื้อหา และใช้เลขโรมัน

ข้อมูลที่มีมักจะอยู่ในส่วนหน้าของหนังสือ ได้แก่

1. หน้าปก

2. คำนำ (preface)
3. คำอุทิศ (acknowledgement)
4. สารบัญ (tableofcontents)
5. สารบัญรูป (listoffigures)
6. สารบัญตาราง (listoftables)

การเรียงลำดับข้อมูลนี้สามารถปรับได้ตามความต้องการ เช่นในหนังสือหรือตำราเรียน มักจะใส่คำนำไว้หลังจากสารบัญ เป็นต้น

ถัดจากส่วนเนื้อหา สามารถมีส่วนภาคผนวก ก่อนจะเป็นส่วนท้าย ซึ่งเป็นบรรณานุกรม วรรณคดี และข้อมูลเกี่ยวกับผู้เขียน

บทที่ 4

การจัดรูปแบบ

4.1 ขนาดและรูปแบบตัวอักษร

ตารางที่ 4.1: การกำหนดขนาดของตัวอักษร

คำสั่ง	ผลลัพธ์
<code>\tiny</code>	text ขอความ
<code>\scriptsize</code>	text ขอความ
<code>\footnotesize</code>	text ขอความ
<code>\small</code>	text ขอความ
<code>\normalsize</code>	text ขอความ ขนาดปกติ
<code>\large</code>	text ขอความ
<code>\Large</code>	text ขอความ
<code>\huge</code>	text ขอความ
<code>\Huge</code>	text ขอความ

ในการพิมพ์ข้อความอาจมีความจำเป็นต้อง **เพิ่ม** หรือ **ลด** ขนาดของตัวอักษรในบางจุด การสร้างเอกสารโดย **L^AT_EX** จะมีป้ายระบุเพื่อใช้กำหนดขนาดสัมพันธ์กับขนาดปกติ โดยใช้ป้ายระบุดังแสดงในตารางที่ 4.1 โดยที่ป้ายระบุเพื่อกำหนดขนาดตัวอักษรเหล่านี้เป็นป้ายระบุแบบไม่ใช่ตัวแปร ซึ่งจะเรียงพิมพ์ด้วยคำสั่งนี้ไปจนกว่าจะหมดขอบเขตของข้อความ หากเราต้องการจำกัดขอบเขตของป้ายระบุสามารถทำได้โดยใส่วงเล็บปีกกา

ตารางที่ 4.2: การจัดรูปแบบตัวอักษร

คำสั่ง	ผลลัพธ์
<code>\textit{...}</code>	ตัวเอียง
<code>\textbf{...}</code>	ตัวหนา
<code>\underline{...}</code>	ขีดเส้นใต้
<code>\sout{...}</code>	ขีดกลาง

ครอบทั้งป้ายระบุและข้อความเพื่อกำหนดขอบเขตบล็อกที่จะให้ป้ายระบุนั้นมีผลในการเรียงพิมพ์

การทำตัวหนา *ตัวเอียง* ขีดเส้นใต้ ~~ขีดกลาง~~ (สำหรับขีดกลางต้องเรียกใช้แพ็คเกจ `ulem`) ใช้ป้ายระบุดังที่สรุปอยู่ในตาราง 4.2 การจัดรูปแบบตัวอักษร *ผสม* กันหลายรูปแบบสามารถทำได้ด้วยการใช้ป้ายระบุซ้อนกัน

โดยปกติเอกสาร `LaTeX` จะใช้ฟอนต์เดียวกันทั้งเอกสาร หากไม่ได้กำหนดฟอนต์เป็นพิเศษ ตัวอักษรทั้งหมดจะใช้ฟอนต์ `Computer Modern` การเปลี่ยนฟอนต์เฉพาะบางส่วนของเอกสารทำได้หลายวิธี หากใช้ `XeTeX/XeLaTeX` ซึ่งสนับสนุน `Unicode` การกำหนดฟอนต์จะใช้แพ็คเกจ `fontspec` แล้วระบุตัวแปรเป็น `font` ที่ต้องการ เช่น `TeX Gyre Heros` หรือ `TeX Gyre Heros` ฟอนต์ที่สามารถใช้ได้คือ `True Type Font` หรือ `OpenType font` ใด ๆ ที่ลงไว้ในเครื่อง สำหรับฟอนต์มาตรฐานของเอกสารราชการไทยในปัจจุบัน (พ.ศ. 2557) คือ `TH Sarabun New` การกำหนดฟอนต์หลักจึงทำได้ดังนี้


```
\usepackage{fontspec}
\setmainfont{TH Sarabun New}
```

เอกสารนี้ใช้คลาส memoir และกำหนดฟอนต์หลักเป็น TeX Gyre Therme เทียบเท่ากับ Times New Roman ใน Windows ขนาดของฟอนต์ปกติ 12 pt และกำหนดขนาดฟอนต์ภาษาไทยให้ขนาดอักษรปกติเท่ากับอักษรตัวพิมพ์ใหญ่ในภาษาอังกฤษ

4.2 การจัดเรียงข้อความ

หากไม่ได้ตั้งค่าใดเป็นพิเศษ ข้อความจะถูกจัดให้ชิดทั้งซ้ายและขวาเสมอ (fully justified)

การจัดข้อความให้อยู่กึ่งกลางพื้นที่

ชิดขวา

หรือบังคับให้ชิดซ้าย

ทำได้โดยคำสั่งต่อไปนี้

- กึ่งกลาง
 - `\centering` หรือ
 - `\begin{center} ... \end{center}`
- ชิดขวา
 - `\raggedleft` หรือ
 - `\begin{flushright} ... \end{flushright}`
- ชิดซ้าย
 - `\raggedright` หรือ
 - `\begin{flushleft} ... \end{flushleft}`

ปัญหาหนึ่งที่พบกับการใช้ $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ในภาษาไทยคือการตัดคำเพื่อจัดหน้าทำได้ไม่ดีนัก บางกรณีอาจมีคำที่ล้นขอบขวาเนื่องจากโปรแกรมไม่สามารถตัดคำได้ ปัญหานี้ยังไม่มีวิธีการแก้ไขที่ถาวร ผู้ใช้จำเป็นต้องเลือกเว้นวรรคข้อความ หรือปรับเนื้อหาข้อความเองเพื่อให้โปรแกรมพยายามเรียงพิมพ์ได้อย่างเหมาะสม

การจัดชิดซ้ายและขวาในภาษาอังกฤษนั้น $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ สามารถตัดกึ่งกลางคำโดยใช้ - (hyphen) ได้โดยอัตโนมัติ และทำได้ค่อนข้างดี ซึ่งปกติแล้วการเว้นช่องไฟต่าง ๆ เพื่อจัดหน้าในภาษาอังกฤษด้วย $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ จะสวยงามกว่าการใช้ Microsoft Word

4.3 ช่องว่างแบบต่างๆ

การเคาะวรรค ย่อหน้า หรือขึ้นบรรทัดใหม่หนึ่งบรรทัดใน $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ นั้นมีผลเหมือนกันในการเรียงพิมพ์ คือ อนุญาตให้จัดช่องไฟกว้างแคบได้ตามความเหมาะสมเมื่อต้องการตัดคำเพื่อขึ้นบรรทัดใหม่ การเคาะวรรคหรือย่อหน้าในการพิมพ์เอกสารจึงมีไว้เพื่อให้ผู้พิมพ์อ่านข้อความได้ง่ายเท่านั้น นอกจากนี้ การขึ้นบรรทัดใหม่บรรทัดเดียวยังมีประโยชน์ในการเขียนย่อหน้ายาวๆ หากเป็นข้อความภาษาอังกฤษ บางครั้งจะนิยมเขียนข้อความบรรทัดละประโยค เพื่อให้สะดวกในการกลับมาหาข้อความเพื่อแก้ไขในอนาคต

แต่การจัดช่องไฟอัตโนมัติอาจทำให้การแสดงผลบางรูปแบบไม่ถูกต้อง เช่น หลังคำว่า รูปที่ ควรเคาะเพียงวรรคเดียวแล้วตามด้วยตัวเลข การบังคับให้เครื่องเว้นวรรคเดียวและห้ามตัดระหว่างคำทำได้โดยใช้เครื่องหมาย \sim แทนการเคาะวรรค เช่น รูปที่ 1 ส่วนการบังคับให้ขึ้นบรรทัดใหม่ ใช้คำสั่ง \backslash

หรือใช้ป้ายระบุ \backslash newline

ก็ได้ ทุกครั้งที่ขึ้นบรรทัดใหม่โดยใช้คำสั่งหรือป้ายระบุนี้จะไม่มีการย่อหน้า หากต้องการให้ย่อหน้าใหม่ให้ใช้การขึ้นบรรทัดใหม่สองครั้ง

หรือใช้ป้ายระบุ \backslash indent เพื่อบังคับให้ขึ้นย่อหน้าใหม่

บางครั้งอาจต้องบังคับเว้นระยะระหว่างย่อหน้า ระหว่างข้อความ รูปภาพ มากกว่าค่าโดยปริยายที่รูปแบบเอกสารจัดให้ คำสั่งที่ใช้ในการเว้นระยะแนวนอนคือ

```
\hspace{<length>}
```

เช่น ส่วนคำสั่งที่ใช้ในการเว้นระยะแนวนอนตั้งคือ อยู่ห่างจากคำ
ก่อนหน้า 10 ซม. การกำหนดระยะสามารถระบุเป็น ซม. (cm) นิ้ว (in) พอยท์ (pt) ได้

จะเห็นว่าระยะที่ใช้จะเป็นลบหรือบวกก็ได้เช่นกัน

4.4 ข้อย่อย

โครงสร้างเนื้อหาอีกรูปแบบซึ่งใช้บ่อยคือข้อย่อย ใน $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ มีโครงสร้างข้อย่อยให้เลือก 2 รูปแบบ แบบนับเลขข้อ (enumerate) และแบบไม่นับเลขข้อ (itemize) ตัวข้อย่อยแต่ละตัวจะอยู่หลังป้ายระบุ `\item` หากต้องการซ่อนข้อย่อยก็แทรกโครงสร้างข้อย่อยลงไปในป้ายระบุ `item` ได้ เช่น

1. ข้อแรกไม่มีข้อย่อย
2. ข้อสองแยกเป็นข้อย่อยแบบไม่นับเลขสองข้อ ได้แก่

- bullet แรก
- bullet ที่สอง

และแบบนับเลขข้ออีกสามข้อ ดังนี้

- a) ข้อสอง ข้อย่อยที่ 1
- b) ข้อสอง ข้อย่อยที่ 2
- c) ข้อสอง ข้อย่อยที่ 3

3. ข้อสาม

ทั้งนี้ เราสามารถกำหนดวิธีการนับเลข (เช่น เลขไทย เลขโรมัน อักษร) และรูปสัญลักษณ์ของ `bullet` (วงกลมทึบ วงกลมโปร่ง สีเหลี่ยม หรือสัญลักษณ์อื่นๆ) ได้เองผ่านตัวเลือกต่างๆ ผู้ที่สนใจสามารถไปค้นหาวิธีเพิ่มเติมได้ในอินเทอร์เน็ต

บทที่ 5

การแทรกรูปภาพและตาราง

5.1 การแทรกรูปภาพ

การแทรกไฟล์รูปภาพทำได้โดยใช้แพ็คเกจ `graphicx` และป้ายระบุดังนี้

```
\includegraphics [options] {filename}
```



```
\includegraphics {Figures/  
Prakeaw-Pink}
```

a รูปเต็ม



```
\includegraphics [width=1cm  
{Figures/Prakeaw-Pink}
```

c รูปขนาดกว้าง 1 ซม. ปรับความสูงอัตโนมัติ



```
\includegraphics [scale  
=.5] {Figures/Prakeaw-Pink}
```

b รูปขนาดครึ่งของรูปเต็ม

d รูปขนาดกว้าง 2 ซม. ความสูง 1 ซม.



```
\includegraphics [width=2cm  
, height=1cm] {Figures/  
Prakeaw-Pink}
```

รูปที่ 5.1: ตัวอย่างการแทรกรูปภาพ


คำสั่ง `includegraphics` มีตัวเลือกเพื่อปรับขนาดของภาพที่จะแทรกได้ด้วย ดังตัวอย่างในรูป 5.1 โดยปกติเรามักใช้ป้ายระบุ `figure` แบบเปิด/ปิดเพื่อกำหนดขอบเขตของรูป ภายในเรียกใช้คำสั่ง `includegraphics` แล้วตามด้วยป้ายกำกับชื่อรูปภาพเหล่านั้น เพื่อให้โปรแกรมเรียงพิมพ์สามารถแยกแยะได้ว่าป้ายกำกับเหล่านี้เป็นป้ายกำกับของรูปภาพ และสามารถนับลำดับเฉพาะรูปภาพได้

ส่วนคำบรรยายภาพจะถูกกำหนดโดยป้ายระบุ `\caption{..}` โดยมากแล้ววารสารต่างๆ มักให้พิมพ์ข้อความบรรยายภาพไว้ใต้ภาพ แต่หากต้องการให้ข้อความบรรยายภาพอยู่เหนือภาพก็ทำได้โดยย้ายตำแหน่งของป้ายระบุ `caption` ไปไว้ก่อนภาพ ดังรูป 5.1d

หากรูปมีขนาดเล็กกว่าพื้นที่หน้ากระดาษ โดยปกติรูปจะถูกวางไว้ชิดซ้าย หากต้องการจัดให้รูปอยู่กึ่งกลางหรือชิดขวาของพื้นที่ก็ใช้วิธีเช่นเดียวกับการจัดหน้าปกตินอกจากนี้ หากต้องการใส่รูปย่อยหลายรูปในภาพเดียวกัน เราสามารถวางรูปด้วยป้ายระบุ `includegraphics` ต่อ ๆ กันไปได้เลย แต่หากต้องการใส่คำอธิบายภาพแยกตามรูปย่อยสามารถทำได้ผ่านการแทรกรูปย่อยด้วยแพ็คเกจ `subcaption` ซึ่งใช้ป้ายระบุ `subfigure` ที่มีโครงสร้างเหมือนป้ายระบุ `figure` ทุกประการ การอ้างอิงถึงรูปย่อยก็ทำเช่นเดียวกับการอ้างอิงถึงรูปปกติ เช่น รูปที่ 5.1c อ้างอิงถึงรูปขนาดกว้าง 1 ซม. ปรับความสูงอัตโนมัติ เป็นต้น

โปรแกรมเรียงพิมพ์จะจัดตำแหน่งของรูปภาพในเอกสารให้เอง ค่าเริ่มต้นที่ตั้งมามักเป็นการจัดให้รูปภาพอยู่บนสุดของหน้ากระดาษ แต่หากต้องการกำหนดตำแหน่งของรูปภาพให้อยู่ท้ายกระดาษ หรือ ณ บริเวณที่กำหนด เราสามารถระบุได้ผ่านตัวเลือกของป้ายระบุ `figure` ได้ เช่น

<code>h</code>	ณ ตำแหน่งใกล้ ๆ นี้
<code>H</code>	ณ ตำแหน่งนี้ (ต้องใช้แพ็คเกจ <code>float</code>)
<code>t</code>	ด้านบนของกระดาษ
<code>b</code>	ด้านล่างของกระดาษ

หากไม่ใช้ป้ายระบุ `figure` ครอบรูปภาพ โปรแกรมจะเรียงพิมพ์รูปนั้นเสมือนเป็นอักขระหนึ่ง เช่น 

5.2 การสร้างตาราง

โครงสร้างตารางอย่างง่ายใช้ป้ายระบุ `tabular` ดังตัวอย่างต่อไปนี้

```
\begin{tabular}{|c||lr|}
t1      & t2      & t3 \\ \hline
abcd    & defghij & klmnop \\ \hline
\end{tabular}
```

เมื่อเรียงพิมพ์แล้ว จะให้ผลออกมาเป็น

t1	t2	t3
abcd	defghij	klmnop

วงเล็บปีกกาแรกถัดจาก `tabular` คือจำนวนและการจัดหน้าของแต่ละคอลัมน์ จากตัวอย่างข้างต้นนี้มีทั้งหมด 3 คอลัมน์ โดย `c l r` แทนกึ่งกลาง ขัดซ้าย และขัดขวา ตามลำดับ ตัวอักษร `|` แทนการขีดเส้นตั้งระหว่างคอลัมน์ ข้อความที่อยู่ระหว่าง `begin` และ `end` คือข้อมูลในตาราง ข้อความแต่ละบรรทัดคือข้อความในแต่ละแถวของตาราง คั่นระหว่างคอลัมน์ในแถวด้วยสัญลักษณ์ `&` และจบแถวด้วยการสั่งขึ้นบรรทัดใหม่ การขีดเส้นนอนในตารางทำได้โดยคำสั่ง `\hline`

ในการทำงานเกี่ยวกับรูปภาพ เรามักใช้ป้ายระบุ `table` แบบกำหนดขอบเขต เพื่อระบุขอบเขตของตาราง และให้โปรแกรมจัดตำแหน่งที่เหมาะสมสำหรับตารางให้เอง (หรือระบุตำแหน่งเองแบบเดียวกับรูปภาพ) รวมถึงนับลำดับการอ้างอิงแยกเป็นตารางด้วย

บทที่ 6

การอ้างอิง

6.1 การอ้างอิงภายในเอกสาร

ป้ายกำกับ (label) มีโครงสร้างดังนี้

```
\label{<label name>}
```

โดย <label name> คือชื่อของป้ายกำกับ การใส่ป้ายกำกับลงไปในเอกสารเพื่อใช้อ้างถึงตำแหน่งนั้น ๆ การอ้างอิงถึงบทและตอน (และตอนย่อยหากมี) ของตำแหน่งป้ายกำกับนั้นใช้ป้ายระบุ `\ref{<label name>}` เช่น หากกำหนดป้ายกำกับที่นี่ `\label{sample}` การอ้างอิงถึง `\ref{sample}` จะได้เป็น 6.1 ส่วนการอ้างอิงถึงหน้าที่ตำแหน่งป้ายกำกับนั้นใช้คำสั่ง `\pageref{<label name>}` เช่น `\pageref{sample}` จะได้ 27 และหากป้ายกำกับอยู่ในขอบเขตของรูปภาพและตาราง การอ้างอิงถึงก็จะได้เลขลำดับของรูปภาพและตารางนั้นมา เช่น รูปที่ 6.1 อยู่ที่หน้า 28 เป็นต้น

6.2 บรรณานุกรม

BibTeX เป็นโปรแกรมการจัดการรูปแบบเอกสารอ้างอิงที่ทำให้ L^AT_EX สามารถเรียงพิมพ์ได้สะดวก หากดาวน์โหลดข้อมูลบรรณานุกรมจากเว็บของสำนักพิมพ์ ทางสำนักพิมพ์มักจะอำนวยความสะดวกด้วยการทำรายละเอียดสำหรับ BibTeX สำหรับบทความแต่ละชิ้นไว้ให้ สามารถคัดลอกไปใช้ได้ทันที เช่น รูปที่ 6.1

ทั้งนี้ ควรตรวจแก้ไขสอบข้อมูลต่าง ๆ ให้ถูกต้องตามรูปแบบที่สิ่งพิมพ์แต่ละชนิดกำหนดด้วย เช่น ชื่อหนังสือ บางวารสารอาจบังคับให้ใช้ชื่อเต็มสำหรับการประชุมวิชาการ

```

@inproceedings{DBLP:conf/pricai/WanvarieTO10,
author    = {Dittaya Wanvarie and
Hiroya Takamura and
Manabu Okumura},
title     = {Active Learning for Sequence Labelling
with Probability Re-estimation},
booktitle = {PRICAI},
year      = {2010},
pages     = {681-686},
ee        = {http://dx.doi.org/10.1007/978-3-642-15246-7_69},
crossref  = {DBLP:conf/pricai/2010},
bibsource = {DBLP, http://dblp.uni-trier.de}
}

@book{thaibib,
author = {{ชิตยา หวานวารี}},
title = {การใช้ \LaTeX สำหรับเรียงพิมพ์วิทยานิพนธ์ภาษาไทยและภาษา
อังกฤษ โดยใช้รูปแบบของจุฬาลงกรณ์มหาวิทยาลัย},
publisher = {{ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์}},
month = {กรกฎาคม},
year = {2557},
}

```

รูปที่ 6.1: ตัวอย่างข้อมูลสำหรับ BibTeX

ในขณะที่ `booktitle` ในข้อมูลที่ได้อาจเป็นชื่อย่อ ก็ต้องแก้เองให้ถูกต้องด้วย เป็นต้น การอ้างอิงถึงเอกสารชิ้นต่าง ๆ ในแฟ้มต้นฉบับทำได้โดยคำสั่ง

```
\cite{<cite name>}
```

และเขียนข้อมูลบรรณานุกรมของแต่ละเอกสารอ้างอิงให้ถูกต้องรวมกับไว้ในแฟ้มรายการบรรณานุกรม ซึ่งมักให้มีนามสกุล `bib` การระบุรูปแบบของบรรณานุกรมให้ใช้รูปแบบที่วารสารกำหนดให้ใช้แฟ้มรูปแบบบรรณานุกรม (นามสกุล `bst`) การระบุรูปแบบและแฟ้มรายการบรรณานุกรมอ้างอิงทำได้ดังนี้

```

\bibliographystyle{<bib style name>}
\bibliography{<bib name>}

```

การใช้ BibTeX นั้น หากใช้โปรแกรมจำพวก IDE ในการสร้างเอกสารต้นฉบับ มัก

จะมีการเรียก BibTeX ให้โดยอัตโนมัติ อย่างไรก็ตาม หากสั่งเรียงพิมพ์โดยตรง ต้องสั่งเรียงพิมพ์ก่อนหนึ่งครั้งเพื่อสร้างรายการบรรณานุกรม แล้วใช้ BibTeX เพื่อโยงรายการอ้างอิงเหล่านั้นกับแฟ้มข้อมูลรายการบรรณานุกรม แล้วจึงสั่งเรียงพิมพ์อีกรอบ เพื่อให้แทนค่าแต่ละตำแหน่งของคำสั่ง \cite ด้วยข้อมูลบรรณานุกรมในเอกสารผลลัพธ์ให้ถูกต้อง เช่นจากรูปที่ 6.1 สามารถอ้างอิงถึงได้ดังนี้

```
\cite{DBLP:conf/pricai/WanvarieT010}
```

และผลลัพธ์ที่ได้เป็นดังนี้ [WTO10] และ \citep{thaibib} จะได้ผลลัพธ์เป็น [ท57]

6.3 เอกสารสำหรับอ่านเพิ่มเติม

รายละเอียดการใช้ LaTeX สามารถอ่านเพิ่มเติมได้ที่

- LaTeX User's Guide and Reference Manual
<http://latex-project.org/guides/usrguide.pdf>
- <http://www.ctan.org/tex-archive/info/lshort/thai>
- <http://en.wikibooks.org/wiki/LaTeX>
- สัญลักษณ์ต่าง ๆ
<http://www.tex.ac.uk/tex-archive/info/symbols/comprehensive/symbols-a4.pdf>
- ตัวอย่างบทความเมื่อใช้คลาสของ AMS
<http://www.ams.org/publications/authors/tex/amslatex>

หรือค้นหาเพิ่มเติมได้จากอินเทอร์เน็ต

บทที่ 7

การแทรกสมการและสัญลักษณ์คณิตศาสตร์

การพิมพ์ข้อความหรือสัญลักษณ์คณิตศาสตร์แบ่งออกเป็น 2 รูปแบบ คือ แทรกในบรรทัดเดียวกัน (inline) เช่น δ^2 หรือแยกออกมาเป็นบรรทัดต่างหากในกรณีพิมพ์สมการ เช่น

$$f(x, y) = \int_a^b \frac{p(x, y)}{\ln y} dx \quad (7.1)$$

จะสังเกตได้ว่าการเรียงพิมพ์ในสภาพแวดล้อมคณิตศาสตร์จะใช้ฟอนต์ที่ต่างไปจากการเรียงพิมพ์ในสภาพแวดล้อมข้อความทั่วไป เมื่อใช้ฟังก์ชันหรือสัญลักษณ์ทางคณิตศาสตร์จึงควรใช้สภาพแวดล้อมคณิตศาสตร์เสมอ

การพิมพ์ในบรรทัดเดียวกันทำได้ด้วยการเขียนข้อความที่ต้องการให้อยู่ระหว่าง `$` และ `$` ส่วนการพิมพ์แยกบรรทัดต่างหากทำได้โดยการพิมพ์ข้อความให้อยู่ระหว่างป้ายระบุแบบเปิดปิด `equation` หรือ `align`

`align` จะสามารถเขียนสัญลักษณ์ต่าง ๆ ได้มากกว่า 1 บรรทัด มีได้หลายคอลัมน์ และสามารถจัดเรียงตำแหน่งได้ด้วย เช่น

$$x = 2 \quad (7.2)$$

$$y = 3 \quad (7.3)$$

$$\begin{aligned} z &= x \times y \\ &= ps \end{aligned} \quad (7.4)$$

การเรียงพิมพ์สมการแยกบรรทัดนั้น โปรแกรมจะสร้างหมายเลขกำกับสมการให้โดยอัตโนมัติ หากต้องการอ้างอิงถึงเลขสมการ ทำได้โดยเติมป้ายกำกับ (label) ลงไป ตั้งชื่อได้ตามต้องการ เมื่อต้องการเรียกใช้ก็เรียกใช้ป้ายกำกับนั้นๆ โดยใช้ป้ายระบุ `\ref{<label name>}` เช่น เราอ้างอิงถึงสมการข้างต้นได้ด้วย (7.1) คำแนะนำในการตั้งชื่อป้ายระบุคือควรตั้งให้สื่อถึงข้อความส่วนนั้น หากไม่ต้องการให้มีเลขสมการ ให้เติม * ลงไปข้างหลัง equation เช่น

$$\mathbf{x}_k = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N}, k \in \mathbb{Z}$$

การเขียนเมตริกซ์ การกำหนดนิยาม ทฤษฎีบท และอื่นๆ สามารถค้นคว้าเพิ่มเติมได้จากอินเทอร์เน็ต

บรรณานุกรม

- [WTO10] Dittaya Wanvarie, Hiroya Takamura, and Manabu Okumura. Active learning for sequence labelling with probability re-estimation. In Byoung-Tak Zhang and Mehmet A. Orgun, editors, PRICAI, volume 6230 of Lecture Notes in Computer Science, pages 681–686, Heidelberg, Germany, 2010. Springer.
- [ท57] จิตยา หวานวารี. การใช้ L^AT_EX สำหรับเรียงพิมพ์วิทยานิพนธ์ภาษาไทยและภาษาอังกฤษ โดยใช้รูปแบบของจุฬาลงกรณ์มหาวิทยาลัย. ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย, กรกฎาคม 2557.