



ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Kiến thức máy tính - Phần 2: Latex

Bài giảng sử dụng tài liệu của Dr. Nicola L. C. Talbot
tại

<http://theoval.cmp.uea.ac.uk/~nlct>



Kiến thức máy tính - Phần II: \LaTeX

Tài liệu giảng dạy tại Viện CNTT-TT

Bài giảng sử dụng tài liệu của Dr Nicola L. C. Talbot tại

<http://theoval.cmp.uea.ac.uk/~nlct>

Chú ý

- Cuối mỗi chủ đề có phần bài tập. Cần thực hiện bài tập để có thể nắm vững nội dung của chủ đề.
- Cần đọc kỹ hướng dẫn trong tài liệu handout.pdf đi kèm, và chú ý đến các **nhắc nhở**.
- Nếu cảm thấy mệt mỏi, có thể thực hiện phần bài tập đánh dấu . Nếu cảm thấy bài tập quá dễ, hãy làm thêm các phần đánh dấu .
- Ghi lại các câu hỏi để có thể hỏi lại về sau.
- \LaTeX đều phân biệt chữ hoa chữ thường, nên cần đánh các lệnh đúng như trong handout đi kèm.

\LaTeX là gì?

- \LaTeX là một *ngôn ngữ* biểu diễn văn bản
- \LaTeX thường được sử dụng để soạn các tài liệu khoa học kỹ thuật kích thước trung bình hoặc dài.
- \LaTeX dựa trên ý tưởng, người viết chỉ cần tập trung vào nội dung của văn bản mà không cần quan tâm đến trình bày của văn bản (layout/presentation)
 - Người viết đặc tả cấu trúc logic: chương, mục, bảng, hình ...
 - Hệ thống \LaTeX xử lý việc trình bày các cấu trúc này trên trang giấy: đặt ở đâu, màu gì, đánh số thế nào, kích thước ra sao ... căn cứ theo một template có sẵn.
- Người viết sử dụng các công cụ soạn thảo văn bản thuần text để soạn các tài liệu \LaTeX . Điều này hoàn toàn khác với các bộ soạn thảo văn bản như MS Word.

Ngôn ngữ lập trình

C

1. Viết/soạn thảo mã chương trình trong một file text (v.d. `HelloWorld.c`)
2. Dịch mã nguồn. (v.d. `►gcc HelloWorld.c`)
 - Nếu có lỗi trở về bước 1.
 - Nếu thành công, tạo ra một file chạy(v.d. `HelloWorld.exe`)

Thực hiện file chạy (v.d. `►HelloWorld`).

Java

1. Viết/soạn thảo mã nguồn trong text file (e.g. `HelloWorld.java`)
2. Dịch mã nguồn. (v.d. `►javac HelloWorld.java`)
 - Nếu có lỗi, trở về bước 1.
 - Nếu thành công, tạo ra Java bytecode file (v.d. `HelloWorld`)
3. Nạp Java bytecode vào máy ảo Java (v.d. `►java HelloWorld`).

\LaTeX

1. Soạn thảo/viết *mã nguồn* trong text file (v.d. HelloWorld.tex)
2. \LaTeX mã nguồn. (v.d. \blacktriangleright latex HelloWorld.tex)
 - Nếu có lỗi trở về bước 1.
 - Nếu thành công tạo ra một file không phụ thuộc vào các hệ thống nền(DVI) (v.d. HelloWorld.dvi)
3. Nạp DVI file vào một trình hiển thị DVI viewer (v.d. \blacktriangleright yap HelloWorld.dvi).

Ưu điểm và nhược điểm của \LaTeX

- ☞ Chỉ có thể hiển thị tài liệu khi đã \LaTeX *mã nguồn*.
- ☹ Không nhìn thấy văn bản hiển thị khi đang gõ.
- 😊 *M*ã nguồn của các tài liệu lớn có kích thước rất khiêm tốn. Soạn thảo một văn bản lớn không ảnh hưởng đến tốc độ gõ. Với Word hoặc OpenOffice, cả tài liệu sẽ bị định dạng lã khi ta gõ.
- 😊 Tài liệu \LaTeX thường có vẻ chuyên nghiệp hơn các tài liệu khác.
- 😊 Miễn phí! (mặc dù một vài front-end như WinEdt, là phần mềm chia sẻ.)

Công cụ soạn tài liệu bằng \LaTeX

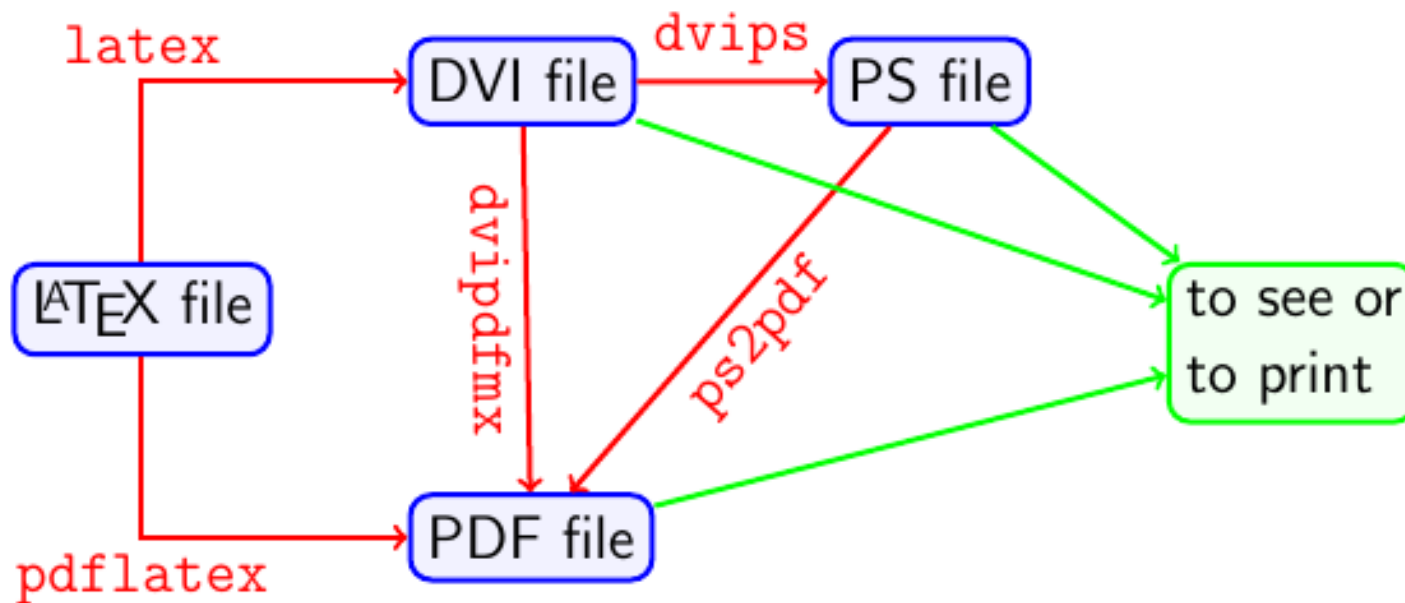
- Editor/Front-end
 - Công cụ để viết một tài liệu dưới dạng văn bản không mã hóa.
 - Có giao diện tương tác với hệ thống xử lý tài liệu (typesetting)
 - Texnic Center: công cụ soạn thảo miễn phí trên Windows
 - TexWorks, đi kèm với bộ dịch Latex (Latex typesetting)
 - TexMaker: chạy được trên nhiều hệ điều hành.
 - Overleaf: Online (trên nền web) hoàn toàn miễn phí.
- Hệ thống xử lý - typesetting system (trình dịch Latex)
 - Xử lý tài liệu Latex được soạn: áp các định dạng để tạo ra sản phẩm cuối là các file với định dạng dvi, pdf, ps
 - teTeX, TexLive: \TeX typesetting cho Linux
 - MiKTeX: \TeX typesetting cho Windows
 - MacTex: \TeX typesetting cho MacOS

Công cụ soạn tài liệu bằng \LaTeX

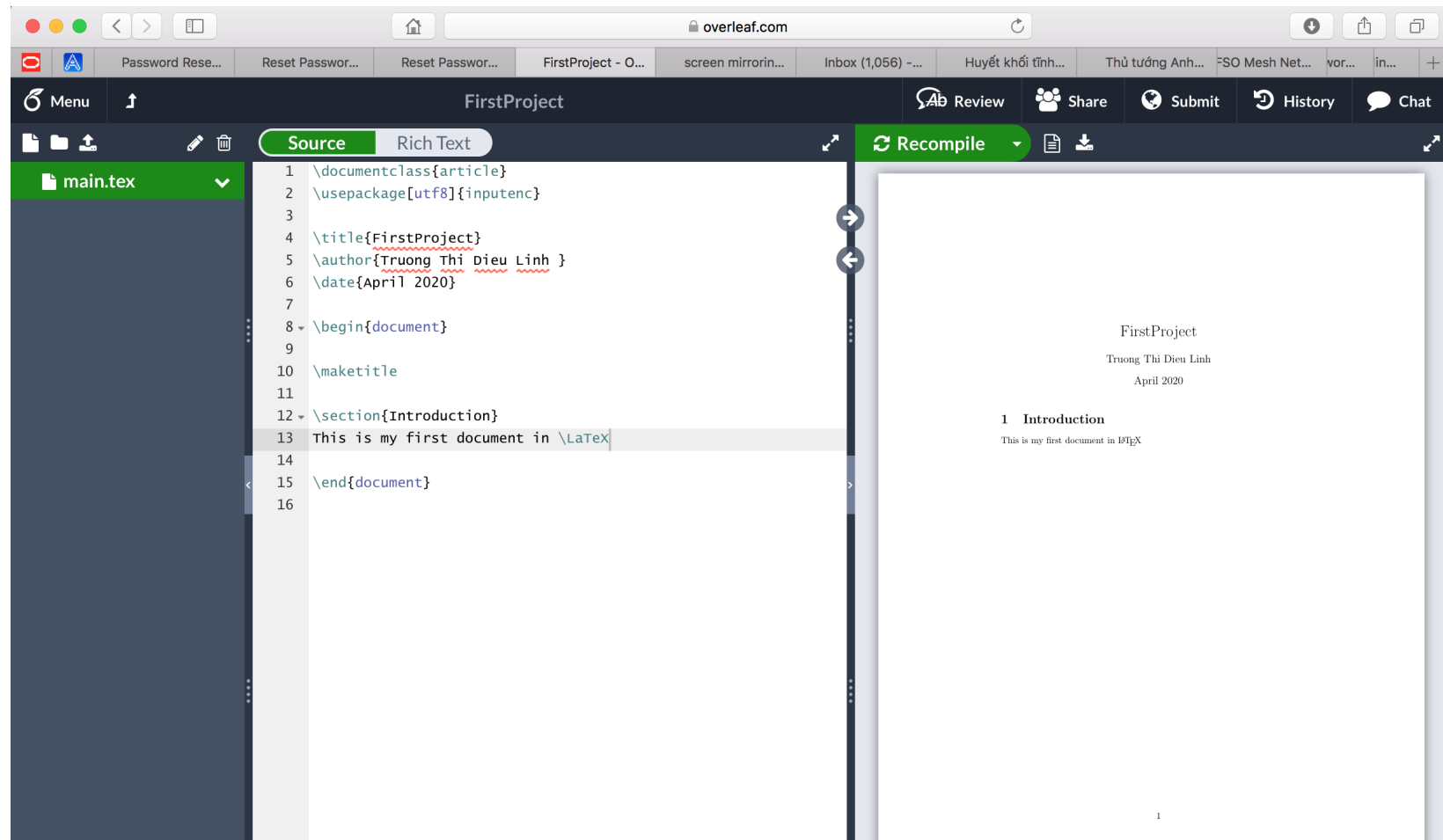
- Công cụ xem các tài liệu được tạo ra từ Latex.
 - Hiển thị tài liệu tạo ra
 - Hiển thị các file .dvi: Kdvi (Linux), Yap (Windows)
 - Hiển thị các file .ps: KGhostview (Linux), GSview (Windows)
 - Hiển thị các file .pdfviewer: Kpdf, xpdf (Linux), Adobe Reader (Windows)

Công cụ soạn tài liệu bằng \LaTeX

- Trình dịch Latex cung cấp một số câu lệnh để dịch các file Latex sang các loại đầu ra khác nhau.
- Nói chung, các Latex Editor có các nút bấm/giao diện để kích hoạt các lệnh này.



Overleaf Editor



Overleaf Editor

- Với Overleaf, trình dịch Latex được cài đặt sẵn trên server trên Internet. Người dùng không cần phải cài đặt trình dịch.
- Editor Overleaf là một trang web, người dùng cũng không cần cài đặt.
- Trên Overleaf, file đầu ra luôn là PDF.

Các ký hiệu đặc biệt

- 10 ký hiệu sau có ý nghĩa đặc biệt và cần được sử dụng cẩn thận:

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

- Dấu phần trăm `%` bắt đầu một chú thích cho mã nguồn. Tất cả những gì sau ký hiệu `%` trên một dòng đều bị \LaTeX bỏ qua.

Một vài từ % chú thích.

INPUT

Một vài từ

OUTPUT

- Các ký hiệu đặc biệt khác sẽ được đề cập đến sau.

Các lệnh(Macros)

- *Các lệnh* cho phép định dạng các phần của tài liệu.
- Có thể được biểu diễn bằng **ký hiệu đặc biệt** (v.d. \$) hoặc một backslash sau đó là một ký hiệu đặc biệt (v.d. \=) hoặc một backslash sau đó là một từ (v.d. \today).
- Tất cả các dấu cách sau tên lệnh đều bị bỏ qua. Chú ý sự khác nhau giữa

\oe

INPUT

œ

OUTPUT

và

\o e

INPUT

øe

OUTPUT

Nhóm

- Một đoạn mã có thể được *nhóm lại* bằng cách đặt trong { and }
- Hầu hết các lệnh đặt trong một nhóm chỉ có tác dụng tổng nhóm đó.

Một đoạn văn bản.
 {Đoạn văn bản này nằm
 trong nhóm.} Một đoạn
 văn bản khác.

INPUT

Một đoạn văn bản.
 Đoạn văn bản này
 nằm trong nhóm.
 Một đoạn văn bản
 khác.

OUTPUT

- Lệnh có thể được nhóm để loại bỏ dấu cách trước lệnh:

man{\oe}uvre

INPUT

manœuvre

OUTPUT

Thông số lệnh

- Một vài lệnh có thể có một hoặc nhiều *tham số*. Ví dụ câu lệnh `\textbf` có một tham số là văn bản sẽ được làm béo.

```
\textbf{Một đoạn văn  
bản chữ đậm}
```

INPUT

**Một đoạn văn bản
chữ đậm**

OUTPUT

- Nếu tham số có nhiều hơn một ký hiệu, các ký hiệu này cần được gộp lại bằng dấu ngoặc nhọn `{ }`. Nếu không, chỉ có ký hiệu đầu tiên được tính là tham số:

```
\textbf Ký hiệu này  
béo
```

INPUT

Ký hiệu này béo

OUTPUT

Các tham số phụ

- Một vài lệnh có các *thông số bổ sung*.
- Các tham số bổ sung được đặt trong các dấu ngoặc vuông []:

Đây là \\một
dòng mới.

INPUT

Đây là
một dòng mới.

OUTPUT

Đây là\\[5mm]
một dòng mới.

INPUT

Đây là
một dòng mới.

OUTPUT

- Tham số bổ sung *nói chung* khai báo trước tham số chính (có một vài trường hợp đặc biệt.)

Môi trường

- Môi trường *environment* khác với các lệnh.
- `\begin{name}` khai báo điểm bắt đầu của một môi trường.
- `\end{name}` khai báo điểm kết thúc của môi trường.

- ```
\begin{sffamily}
Văn bản dạng
sans-serif
\end{sffamily}
```

INPUT

Văn bản dạng sans-serif.

OUTPUT

- Môi trường cũng thực hiện việc nhóm các đoạn văn bản. Các thay đổi trong một môi trường thường là cục bộ trong môi trường.
- Môi trường cũng có thể có các *tham số* chính và phụ.

## Dấu cách

- Các dấu cách liên tiếp được xử lý như một dấu cách duy nhất.

Một đoạn văn bản.

INPUT

Một đoạn văn bản.

OUTPUT

- Các ký hiệu xuống dòng và các ký hiệu tab đều được hiển thị thành các dấu cách.
- Hiển thị một dấu cách sau một lệnh, sử dụng `\_`:

`\LaTeX\` thật tuyệt  
vời!

INPUT

L<sup>A</sup>T<sub>E</sub>X thật tuyệt  
vời!

OUTPUT

- Có thể sử dụng `\_` dấu cách sau các từ viết tắt:

v.d.\thế này.

INPUT

v.d. thế này.

OUTPUT

## Đoạn văn-Paragraphs

- Một dòng trắng định nghĩa cuối một đoạn văn.

Đây là đoạn  
văn đầu tiên.

Đây là đầu  
đoạn văn thứ hai.

INPUT

Đây là đoạn văn  
đầu tiên.

Đây là đầu đoạn  
văn thứ hai.

OUTPUT

- Ngắt đoạn văn cũng có thể được khai báo bằng câu lệnh `\par`

## Tạo tài liệu mới

- Tất cả các file  $\text{\LaTeX} 2_{\epsilon}$  đều bắt đầu bằng câu lệnh *command*

```
\documentclass[options]{class}
```

Câu lệnh này định nghĩa *Loại văn bản*. Ví dụ:

```
\documentclass[a4paper,12pt]{article}
```

- Tất cả các đoạn văn bản của tài liệu cần được nằm trong môi trường *document environment*:

```
\begin{document}
```

định nghĩa điểm đầu của một tài liệu

```
\end{document}
```

định nghĩa điểm cuối của tài liệu.

## Tạo một tài liệu

Tất cả tài liệu  $\text{\LaTeX}$  đều có 3 dòng lệnh sau:

```
\documentclass[a4paper,12pt]{article}
```

```
\begin{document}
```

```
\end{document}
```

Dòng đầu tiên có thể thay đổi với các tài liệu khác nhau, nhưng hai dòng còn lại luôn luôn giống như ở trên.

## Bài tập 1 : Tạo một tài liệu đơn giản (Trang 1)

- Mở editor, và soạn thảo nội dung giống trong Hình 1 ở trang 2 của handouts.
- Ấn vào phím build .
- Nếu có lỗi, chọn Build → Next Error để xem lỗi đã xảy ra.
- Để xem file đầu ra, ấn phím view output.

## Các lệnh thay đổi Font và các khai báo Font

**Một lệnh thay đổi font** là cấu trúc không làm ảnh hưởng đến phần còn lại của tài liệu. Có thể hiểu các lệnh là: *làm thao tác X với đối tượng sau*. Ví dụ, `\textbf{A}` có nghĩa là: “làm cho đoạn văn bản sau có font béo”, trong đó đối tượng là chữ cái A.

**Một lệnh khai báo Font** là cấu trúc ảnh hưởng đến toàn bộ văn bản từ vị trí câu lệnh. Ví dụ, `\bfseries` thay đổi sang chế độ font béo cho tất cả các đoạn văn sau lệnh này.



## Các lệnh thay đổi Font

| Command                               | Sample Input                      | Sample Output     |        |
|---------------------------------------|-----------------------------------|-------------------|--------|
| <code>\textrm{<i>text</i>}</code>     | <code>\textrm{Roman}</code>       | Roman             | Family |
| <code>\textsf{<i>text</i>}</code>     | <code>\textsf{Sans serif}</code>  | Sans serif        |        |
| <code>\texttt{<i>text</i>}</code>     | <code>\texttt{typewriter}</code>  | typewriter        |        |
| <code>\textmd{<i>text</i>}</code>     | <code>\textmd{medium}</code>      | medium            | Weight |
| <code>\textbf{<i>text</i>}</code>     | <code>\textbf{bold}</code>        | <b>bold</b>       |        |
| <code>\textup{<i>text</i>}</code>     | <code>\textup{upright}</code>     | upright           | Shape  |
| <code>\textit{<i>text</i>}</code>     | <code>\textit{italic}</code>      | <i>italic</i>     |        |
| <code>\textsl{<i>text</i>}</code>     | <code>\textsl{slanted}</code>     | <i>slanted</i>    |        |
| <code>\textsc{<i>text</i>}</code>     | <code>\textsc{Small Caps}</code>  | SMALL CAPS        |        |
| <code>\emph{<i>text</i>}</code>       | <code>\emph{emphasized}</code>    | <i>emphasized</i> |        |
| <code>\textnormal{<i>text</i>}</code> | <code>\textnormal{default}</code> | default           |        |

## Các lệnh khai báo Font

| Declaration              | Sample Input                      | Sample Output     |        |
|--------------------------|-----------------------------------|-------------------|--------|
| <code>\rmfamily</code>   | <code>\rmfamily Roman</code>      | Roman             | Family |
| <code>\sffamily</code>   | <code>\sffamily Sans serif</code> | Sans serif        |        |
| <code>\ttfamily</code>   | <code>\ttfamily typewriter</code> | typewriter        |        |
| <code>\mdseries</code>   | <code>\mdseries medium</code>     | medium            | Weight |
| <code>\bfseries</code>   | <code>\bfseries bold</code>       | <b>bold</b>       |        |
| <code>\upshape</code>    | <code>\upshape upright</code>     | upright           | Shape  |
| <code>\itshape</code>    | <code>\itshape italic</code>      | <i>italic</i>     |        |
| <code>\slshape</code>    | <code>\slshape slanted</code>     | <i>slanted</i>    |        |
| <code>\scshape</code>    | <code>\scshape Small Caps</code>  | SMALL CAPS        |        |
| <code>\em</code>         | <code>\em emphasized</code>       | <i>emphasized</i> |        |
| <code>\normalfont</code> | <code>\normalfont default</code>  | default           |        |

## Ví dụ thay đổi Font

1.

`\em` Đoạn văn nhấn mạnh.

INPUT

*Đoạn văn nhấn mạnh.*

OUTPUT

2.

Đoạn văn `\emph{nhấn mạnh}`.

INPUT

Đoạn văn *nhấn mạnh*.

OUTPUT

3.

`\sffamily` Some  
`\textsl{slanted}` text.

INPUT

Some *slanted* text.

OUTPUT

## Ví dụ thay đổi font

4. `\scshape` Some more  
`\upshape` text.

INPUT

SOME MORE text.

OUTPUT

5. `\itshape` Some  
`\emph{emphasized}` text.

INPUT

*Some emphasized text.*

OUTPUT

6. `{\bfseries` Some  
`bold}` text.

INPUT

**Some bold text.**

OUTPUT

## Các môi trường thay đổi Font

- Một *môi trường* cũng có thể được sử dụng để thay đổi cục bộ các tính chất của font.
- Tên của môi trường giống như tên của khai báo font *declaration*, không có \ đi trước.
- Ví dụ:

```
Some normal text.
\begin{bfseries}
Some bold text.
\end{bfseries}
Back to normal text.
```

INPUT

```
Some normal text. Some
bold text. Back to
normal text.
```

OUTPUT

## Thay đổi kích cỡ font

| Khai báo                   | Môi trường                | Ví dụ               |
|----------------------------|---------------------------|---------------------|
| <code>\tiny</code>         | <code>tiny</code>         | tiny text           |
| <code>\scriptsize</code>   | <code>scriptsize</code>   | script sized text   |
| <code>\footnotesize</code> | <code>footnotesize</code> | footnote sized text |
| <code>\small</code>        | <code>small</code>        | small text          |
| <code>\normalsize</code>   | <code>normalsize</code>   | normal sized text   |
| <code>\large</code>        | <code>large</code>        | large text          |
| <code>\Large</code>        | <code>Large</code>        | even larger         |
| <code>\LARGE</code>        | <code>LARGE</code>        | larger still        |
| <code>\huge</code>         | <code>huge</code>         | huge                |
| <code>\Huge</code>         | <code>Huge</code>         | really huge         |

## Ví dụ về thay đổi kích cỡ font

1. Some normal sized text.  
{\small Some small  
text.} Normal again.

INPUT

Some normal sized  
text. Some small text.  
Normal again.

OUTPUT

2. Some \textbf{\large  
large bold} text.

INPUT

Some **large bold**  
text.

OUTPUT

## Ví dụ về thay đổi kích cỡ font

4. `\begin{footnotesize}`  
Some text.  
`\end{footnotesize}`

INPUT

Some text.

OUTPUT

5. Some normal text.  
`\begin{tiny}`  
`\itshape`  
Some tiny italic text.  
`\end{tiny}`  
Back to normal.

INPUT

Some normal text.  
*Some tiny italic text.* Back  
to normal.

OUTPUT



## Bài tập 2 : Fonts (Trang 2)

- Trở lại bài tập 1.
- Đoạn văn đầu tiên đổi sang font sans serif lớn, đoạn văn thứ 2 giữ nguyên font Roman thường.
- Nhấn mạnh từ “Footnotes” trong đoạn 2, và thay đổi cả đoạn văn sang chữ nghiêng.

## Ký hiệu

Chúng ta đã nói đến các ký hiệu đặc biệt **các ký hiệu đặc biệt**? Làm thế nào có thể hiển thị chúng? Làm thế nào để hiển thị các ký hiệu không có trên bàn phím?

|                     |    |                                |   |                    |   |                           |     |
|---------------------|----|--------------------------------|---|--------------------|---|---------------------------|-----|
| <code>\%</code>     | %  | <code>\textasciicircum</code>  | ^ | <code>\P</code>    | ¶ | <code>\ldots</code>       | ... |
| <code>\\$</code>    | \$ | <code>\textasciitilde</code>   | ~ | <code>\S</code>    | § | <code>\textbar</code>     |     |
| <code>\#</code>     | #  | <code>\textbackslash</code>    | \ | <code>\yen</code>  | ¥ | <code>\textgreater</code> | >   |
| <code>\&amp;</code> | &  | <code>\copyright</code>        | © | <code>\i</code>    | ı | <code>\textless</code>    | <   |
| <code>\{</code>     | {  | <code>\textregistered</code>   | ® | <code>\j</code>    | Ƶ | <code>\textbullet</code>  | •   |
| <code>\}</code>     | }  | <code>\texttrademark</code>    | ™ | <code>\ddag</code> | ‡ | <code>\pounds</code>      | £   |
| <code>\_</code>     | _  | <code>\textvisiblespace</code> | ␣ | <code>\dag</code>  | † |                           |     |

## Ví dụ về các ký hiệu

1.

`\pounds 43.50`

INPUT

£43.50

OUTPUT

2.

`A, B \& C`

INPUT

A, B & C

OUTPUT

3.

`As she opened the  
door, she saw \ldots`

INPUT

As she opened the  
door, she saw ...

OUTPUT

## Các ký hiệu phân cách

- Các ký hiệu phân cách thông thường có thể được nhập bình thường: , . ; : ? !
- Một vài ký hiệu cần đến các tổ hợp đặc biệt của các ký hiệu khác.

| Quotes |   |       | Dashes |         | Foreign Punctuation |    |
|--------|---|-------|--------|---------|---------------------|----|
| ‘      | ’ | —     | -      | hyphen  | ?’                  | ?’ |
| ‚      | › | --    | —      | en dash | !‘                  | !‘ |
| “      | ” | ---   | —      | em dash |                     |    |
| „      | ” | \$-\$ | —      | minus   |                     |    |

- Khoảng cách nhỏ \, có thể được sử dụng để phân cách nháy đơn và nháy kép. Ví dụ, ’\,’ hiển thị thành ’’

## Ví dụ

1.

See pages 23--30

INPUT

See pages 23–30

OUTPUT

2.

She opened the box ---  
the twenty-second  
one --- and  
fainted in surprise.

INPUT

She opened the  
box — the twenty-  
second one — and  
fainted in surprise.

OUTPUT

3.

‘‘She said to me:  
‘is that it?’\,’’

INPUT

“She said to me: ‘is  
that it?’”

OUTPUT

## Các dấu trọng âm

Các lệnh trọng âm cho phép đặt các dấu lên trên các ký hiệu.

| Ví dụ                    |                      |        | Ví dụ                    |                     |        |
|--------------------------|----------------------|--------|--------------------------|---------------------|--------|
| Definition               | Input                | Output | Definition               | Input               | Output |
| <code>\' {object}</code> | <code>\' {c}</code>  | ć      | <code>\={object}</code>  | <code>\={c}</code>  | ċ      |
| <code>\' {object}</code> | <code>\' {c}</code>  | ċ      | <code>\. {object}</code> | <code>\. {c}</code> | ċ      |
| <code>\^ {object}</code> | <code>\^ {c}</code>  | ĉ      | <code>\~ {object}</code> | <code>\~ {c}</code> | č      |
| <code>\" {object}</code> | <code>\" {c}</code>  | č      | <code>\v {object}</code> | <code>\v {c}</code> | č      |
| <code>\u {object}</code> | <code>\u {c}</code>  | č      | <code>\H {object}</code> | <code>\H {c}</code> | č      |
| <code>\t {object}</code> | <code>\t {cc}</code> | ĉĉ     | <code>\c {object}</code> | <code>\c {c}</code> | ç      |
| <code>\d {object}</code> | <code>\d {c}</code>  | ç      | <code>\b {object}</code> | <code>\b {c}</code> | ċ      |

## Ví dụ về các từ với trọng âm

1.

Caf\'e

INPUT

Café

OUTPUT

2.

Cr\'eche

INPUT

Crèche

OUTPUT

3.

P\^at\'e

INPUT

Pâté

OUTPUT

4.

Na\"{\i}ve

INPUT

Naïve

OUTPUT

Chú ý việc sử dụng ký hiệu **dotless i**.

## Các chữ ghép

|                  |    |                  |     |                  |    |                  |     |
|------------------|----|------------------|-----|------------------|----|------------------|-----|
| <code>\AE</code> | Æ  | <code>\ae</code> | æ   | <code>\OE</code> | Œ  | <code>\oe</code> | œ   |
| <code>fi</code>  | fi | <code>ffi</code> | ffi | <code>fl</code>  | fl | <code>ffl</code> | ffl |

## Đơn vị tiền tệ

|                  |   |                  |   |                  |   |                  |   |
|------------------|---|------------------|---|------------------|---|------------------|---|
| <code>\AA</code> | Å | <code>\aa</code> | å | <code>\L</code>  | Ł | <code>\l</code>  | ł |
| <code>\O</code>  | Ø | <code>\o</code>  | ø | <code>\SS</code> | Œ | <code>\ss</code> | ß |



## Ví dụ từ có các chữ ghép

|    |                                           |                                        |
|----|-------------------------------------------|----------------------------------------|
| 1. | <div>Man{\oe}uvre</div> <div>INPUT</div>  | <div>Manœuvre</div> <div>OUTPUT</div>  |
| 2. | <div>{\AE}olian</div> <div>INPUT</div>    | <div>Æolian</div> <div>OUTPUT</div>    |
| 3. | <div>{\OE}sophagus</div> <div>INPUT</div> | <div>Œsophagus</div> <div>OUTPUT</div> |
| 4. | <div>fluffier</div> <div>INPUT</div>      | <div>fluffier</div> <div>OUTPUT</div>  |

## Bài tập 3 : Đấu chấm phẩy, trọng âm, ký hiệu (trang 4)

Tạo tài liệu  $\text{\LaTeX}$  như trong hình vẽ 2 trang 5 của handouts.

**Chú ý** điểm sau đây:

- Các lệnh trọng âm chỉ có một tham số là ký hiệu cần đánh trọng âm.
- Trọng âm trên  $i$  được đánh bằng **dotless  $i$** .
- Chú ý **nhóm** các lệnh ký hiệu hoặc phải đặt một dấu cách sau các lệnh đó.

## Các lớp tài liệu

- Lớp tài liệu (document class) được định nghĩa sử dụng câu lệnh:  
`\documentclass[option-list]{class}`

- Có nhiều lớp có sẵn. Các lớp chuẩn là:

`article`   `report`   `book`   `slides`   `letter`

- Cần dùng lớp nào?

- ☞ Cần viết tài liệu như thế nào?

- ☞ Có cần tóm tắt không? (`article` hoặc `report` class)

- ☞ Chia thành các mục (`article` class), hay cần chia thành các chương (`report` hoặc `book` class)?

## Các lớp tài liệu chuẩn

Một vài tùy biến thường dùng:

|                          |                               |
|--------------------------|-------------------------------|
| <code>onecolumn</code>   | Tài liệu một cột              |
| <code>twocolumn</code>   | Tài liệu hai cột              |
| <code>titlepage</code>   | Tiêu đề trên một trang riêng  |
| <code>notitlepage</code> | Tiêu đề trên đầu trang nhất   |
| <code>oneside</code>     | Định dạng để in một mặt       |
| <code>twoside</code>     | Định dạng để in 2 mặt         |
| <code>portrait</code>    | Tài liệu nằm ngang            |
| <code>landscape</code>   | Tài liệu đứng dọc             |
| <code>10pt</code>        | Font mặc định kích thước 10pt |
| <code>11pt</code>        | Font mặc định 11pt            |
| <code>12pt</code>        | Font mặc định 12pt            |

## Các lệnh phân đoạn chuẩn

`\part [Tiêu đề] {Tiêu đề}`

`\subsubsection [Tiêu đề] {Tiêu đề}`

`\chapter [Tiêu đề] {Tiêu đề}`

`\paragraph [Tiêu đề] {Tiêu đề}`

`\section [Tiêu đề] {Tiêu đề}`

`\subparagraph [Tiêu đề] {Tiêu đề}`

`\subsection [Tiêu đề] {Tiêu đề}`

- Có một số lệnh chỉ được sử dụng trong các loại tài liệu nhất định.
- Các phần, chương, mục, mục con, mục nhỏ đều được đánh số tự động.
- `\paragraph` và `\subparagraph` thường được dùng để đánh dấu các phân đoạn không đánh dấu. Có thể thay đổi cấu hình mặc định này

## Ví dụ về các phần của văn bản

1.

```
\section{Giới thiệu}
Tài liệu \LaTeX\ là \ldots
```

INPUT

# 1 Giới thiệu

Tài liệu L<sup>A</sup>T<sub>E</sub>X là ...

OUTPUT

2.

```
\subsection{Macros}
Macros\ldots
```

INPUT

## 1.1 Macros

Macros...

OUTPUT

3.

```
\section*{Unnumbered
Sections}
```

INPUT

## Unnumbered Sections

OUTPUT

## Tóm tắt - Abstract

- Một vài *lớp tài liệu* như *article* và *report* định nghĩa *môi trường abstract* để biểu diễn phần tóm tắt của văn bản.
- Ví dụ:

```
\begin{abstract}
Đây là nội dung của
tóm tắt. Định dạng
phụ thuộc vào loại
văn bản sử dụng.
\end{abstract}
```

INPUT

### Tóm tắt nội dung

Đây là nội dung  
của tóm tắt. Định  
dạng phụ thuộc  
vào loại văn bản sử  
dụng.

OUTPUT

## Trang tiêu đề

- Để có thể tạo ra trang bìa, cần phải đưa một số thông tin bằng các lệnh:

`\author{Author Names}`

`\title{Document Title}`

`\date{Date}`

- Các thông tin đó sau đó được hiển thị bằng lệnh:

`\maketitle`



## Ví dụ về trang bìa

```
\author{Hà Quốc Trung}
\title{Nhập môn
 LaTeX}
\maketitle
```

INPUT

Nhập môn L<sup>A</sup>T<sub>E</sub>X

Hà Quốc Trung

November 2006

OUTPUT

## Phụ lục

- Để viết phụ lục, sử dụng lệnh:  
`\appendix`  
Để bắt đầu một phụ lục.
- Có thể tiếp tục dùng các lệnh `\chapter` hoặc `\section`, phụ thuộc vào *loại tài liệu*.

## Ví dụ

```
% This is the 4th section
\section{Kết luận}
Here are the conclusions.
```

```
\appendix
\section{Tables}
Đây là phụ lục số 1.
\section{Proofs}
Đây là phụ lục số 2.
```

INPUT

## 4 Kết luận

Here are the conclusions.

### A Tables

Đây là phụ lục số 1.

### B Proofs

Đây là phụ lục số 2.

OUTPUT

## Mục lục

- Mục lục có thể được hiển thị sử dụng lệnh:  
`\tableofcontents`
- Các lệnh phân cấp tài liệu ghi các thông tin vào một file mục lục(`.toc`).
- Lần  $\text{\LaTeX}$  tài liệu tuêps theo, file mục lục sẽ được đọc, mục lục sẽ được sinh ra.
- Nếu tên của chương mục quá dài, có thể sử dụng tham số phụ của **các lệnh phân cấp văn bản** để chỉ ra một tiêu đề ngắn hơn.
- Lệnh `\tableofcontents` thường được đặt sau lệnh `\maketitle`.

## Kiểu định dạng (Style) của trang

- Số trang thường được đặt tự động
- Mặc định, trong loại văn bản **article** số trang xuất hiện ở giữa và dưới.
- Kiểu định dạng trang(cách thức hiển thị headers và footers)có thể được thay đổi sử dụng câu lệnh:

```
\pagestyle{style}
```

- Các style thường dùng là: **plain**, **empty** and **headings**.
- Style của trang hiện tại có thể được đặt sử dụng:

```
\thispagestyle{style}
```

## Kiểu đánh số trang

- Số trang được hiển thị bằng các số theo mặc định.
- Định dạng được thay đổi bằng: `\pagenumbering{style}`
- Các kiểu chuẩn là: `arabic`, `roman`, `Roman`, `alph` and `Alph`.
- `\pagenumbering` sẽ đặt số trang bắt đầu từ 1.
- Thông thường, các chữ số la mã thường được dùng cho các phần đầu (lời nói đầu, thay cho lời tựa, ...), còn số thông thường được dùng cho các phần còn lại của văn bản:

```
\maketitle
\pagenumbering{roman}
\tableofcontents % Front matter
\begin{abstract} ... \end{abstract}
\chapter{Introduction} % Start of main matter
\pagenumbering{arabic}
```

## Exercise 4 : Sectioning Commands etc (Page 6)

- Download the file `sectioning.tex`. Load the file `sectioning.tex` into TeXnicCenter, and find the line that says:  
`% CHAPTER : Introduction`  
On the following line, insert the line  
`\chapter{Introduction}`
- Go through the rest of the file, and insert the appropriate sectioning commands.
- Use `\maketitle` to make the title information appear.
- Make the table of contents appear, by placing the command `\tableofcontents` at the place where you want it to appear.

## Định dạng đoạn văn

Mặc định, các đoạn văn được canh lề đều hai bên, tuy nhiên định dạng này có thể được thay đổi sử dụng *khai báo*, hoặc *môi trường*.

|             |                           |                          |                         |
|-------------|---------------------------|--------------------------|-------------------------|
| Khai báo    | <code>\raggedright</code> | <code>\raggedleft</code> | <code>\centering</code> |
| Environment | <code>flushleft</code>    | <code>flushright</code>  | <code>center</code>     |



## Ví dụ (môi trường)

1.

```
\begin{flushright}
Some right
justified text.
\end{flushright}
```

INPUT

Some right justified  
text.

OUTPUT

2.

```
\begin{center}
Some more text,
this time it is centred.
\end{center}
```

INPUT

Some more text,  
this time it is  
centred.

OUTPUT

## Ví dụ (lệnh khai báo)

Các khai báo về canh lề được thực hiện cho *cả* đoạn văn.

1. Canh lề được áp dụng cho cả đoạn văn:

```
{\raggedleft Canh lề
phải đây này.\par}
```

INPUT

Canh lề  
phải đây này.

OUTPUT

2. Các dấu cách đoạn văn được bỏ qua, đoạn văn được định dạng theo đầu đoạn văn:

```
{\raggedleft Canh lề
phải đây này.}
```

INPUT

Canh lề  
phải đây này.

OUTPUT

## Canh giữa một dòng

Có cả những *lệnh* để canh giữa một dòng:

```
\centerline{text}
```

Ví dụ:

```
\centerline{Some centred text}
```

Some centred text

## Các dòng mới

- Để xuống dòng: `\\[length]` hoặc `\newline`

- ```
Line one\\
Line two\\[20pt]
Line three
```

INPUT

```
Line one
Line two

Line three
```

OUTPUT

- ```
\begin{flushright}
Line one\\
Line two\\[20pt]
Line three
\end{flushright}
```

INPUT

```
Line one
Line two

Line three
```

OUTPUT

## Ngắt dòng

- Để ngắt một dòng và giữ nguyên canh lề: `\linebreak[n]`

- A short fully justified paragraph.

INPUT

A short fully justified paragraph.

OUTPUT

- A short \linebreak fully justified paragraph.

INPUT

A short  
fully justified  
paragraph.

OUTPUT

## Chống ngắt dòng

- Để chống ngắt dòng theo từ dùng: `\nolinebreak[n]`

- |                                       |
|---------------------------------------|
| A short fully<br>justified paragraph. |
|---------------------------------------|

INPUT

|                                       |
|---------------------------------------|
| A short fully justified<br>paragraph. |
|---------------------------------------|

OUTPUT
- |                                                       |
|-------------------------------------------------------|
| A short fully<br>justified\nolinebreak\<br>paragraph. |
|-------------------------------------------------------|

INPUT

|                                         |
|-----------------------------------------|
| A short fully justi-<br>fied paragraph. |
|-----------------------------------------|

OUTPUT

## Khoảng trống đi kèm

Có thể dùng ký hiệu ~ để có một dấu cách không nằm riêng một dòng. Ví dụ:

Numbers such as the 3  
in Example 3, should  
never occur at the  
start of a new line.

INPUT

Numbers such as the 3 in Example  
3, should never occur at the start  
of a new line.

OUTPUT

Numbers such as the 3  
in Example~3, should  
never occur at the  
start of a new line.

INPUT

Numbers such as the 3 in Exam-  
ple 3, should never occur at the  
start of a new line.

OUTPUT

## Cách trang

- Cách trang bắt buộc

`\newpage`

- Để tạo một cách trang căn lề đứng, dùng:

`\pagebreak[n]`

- Để chống ngắt trang dùng:

`\nopagebreak[n]`

- Cách trang và hiển thị tất cả các ảnh treo, dùng:

`\clearpage`



## Bài tập 5 : Định dạng đoạn văn (Trang 6)

- Tạo đầu ra giống hình 3 ở trang 7 của handouts.
- Dùng thử các lệnh *khai báo* hoặc *môi trường* hoặc lệnh `\centerline`.

## Định nghĩa câu lệnh mới

- Để định nghĩa câu lệnh mới dùng:

`\newcommand{cmd-name [nargs] [default] {text}`

- *cmd-name* là tên của lệnh mới (chú ý dấu gạch)
- *nargs* là số các *thông số* cần cho câu lệnh (mặc định 0)
- *default* là giá trị mặc định của tham số đầu tiên
- *text* là kết quả của câu lệnh.
- Các câu lệnh đã có được định nghĩa lại bằng `\renewcommand`, nhưng **không bao giờ** định nghĩa lại một lệnh chưa rõ chức năng.

## Vì sao định nghĩa câu lệnh mới?

- Viết ngắn hơn:

```
\newcommand{\introLaTeX}{%
\emph{Introductory \LaTeX}}
```

The \introLaTeX\ course  
is run by \ldots

INPUT

The *Introductory*  $\LaTeX$   
course is run by ...

OUTPUT

## Vì sao định nghĩa câu lệnh mới?

- Đảm bảo tính thống nhất:

```
\newcommand{\envname}[1]{%
\textsf{#1}}
```

```
The \envname{abstract}
environment\ldots
```

INPUT

```
The abstract envi-
ronment...
```

OUTPUT

## Ví dụ

```
% First define the new command
\newcommand{\price}[2]{\pounds #1.#2}
% Now you can use it
The price is \price{2}{50}.
```

INPUT

The price is £2.50.

OUTPUT

```
\newcommand{\cost}[2][17.5]{%
The cost is \pounds #2 excl.\ VAT
@ #1\%}
%
\cost{100}.\
\cost[0.0]{50}
```

INPUT

The cost is £100 excl.  
VAT @ 17.5%.  
The cost is £50 excl.  
VAT @ 0.0%

OUTPUT

## Bài tập 6 : Định nghĩa câu lệnh mới ( trang 7)

- Tạo một tài liệu mới tên là `exercise6.tex`.
- Định nghĩa lệnh `\timeofday` (in the *preamble*). Lệnh này có 2 tham số giờ và phút. Ví dụ câu lệnh `\timeofday{10}{25}` cần được hiển thị: 10:25.
- Tạo ra đầu ra như hình 4 trang 8 hiển thị thời gian bằng lệnh `\timeofday` .
- Sau đó, thay đổi định nghĩa của lệnh để thời gian được hiện bằng chữ đậm và phân cách bằng gạch ngang.(v.d. **10-25**).

## Các môi trường tạo danh sách

*Môi trường* `itemize` tạo ra danh sách *không sắp xếp* .

```
\begin{itemize}
\item The first item
\item The second item
\item The third item
\end{itemize}
```

INPUT

- The first item
- The second item
- The third item

OUTPUT

## Các môi trường **itemize** lồng nhau

Tối đa 4 **itemize** có thể lồng nhau:

```
\begin{itemize}
\item The first item
 \begin{itemize}
 \item First item
 of nested list
 \item Second item
 of nested list
 \end{itemize}
\item The second item
\end{itemize}
```

INPUT

- The first item
  - First item of nested list
  - Second item of nested list
- The second item

OUTPUT



## Danh sách đánh số

*Môi trường* `enumerate` tạo ra một danh sách *đánh số* .

```
\begin{enumerate}
\item The first item
\item The second item
\item The third item
\end{enumerate}
```

INPUT

1. The first item
2. The second item
3. The third item

OUTPUT

## Các môi trường **enumerate** lồng nhau

Tối đa 4 môi trường **enumerate** có thể lồng nhau:

```
\begin{enumerate}
\item The first item
 \begin{enumerate}
 \item First item
 of nested list
 \item Second item
 of nested list
 \end{enumerate}
\item The second item
\end{enumerate}
```

INPUT

1. The first item
  - (a) First item of nested list
  - (b) Second item of nested list
2. The second item

OUTPUT

## Môi trường **itemize** và **enumerate** lồng nhau

**itemize** và **enumerate** có thể lồng nhau:

```
\begin{enumerate}
\item The first item
 \begin{itemize}
 \item First item
 of nested list
 \item Second item
 of nested list
 \end{itemize}
\item The second item
\end{enumerate}
```

INPUT

1. The first item
  - First item of nested list
  - Second item of nested list
2. The second item

OUTPUT

## Description

```
\begin{description}
\item[Cabbage] A large
round green vegetable
\item[Brussel sprout] A
small round
green vegetable
\end{description}
```

INPUT

**Cabbage** A large round  
green vegetable

**Brussel sprout** A small  
round green vegetable

OUTPUT

## Bài tập 7 : danh sách (Page 8)

- Tạo ra tài liệu giống trong hình 5 trang 8 của handouts.
  - Cần dùng các môi trường `enumerate` `environments` lồng nhau
- ④ Đầu tiên là một danh sách đơn giản:
1. Animal
  2. Vegetable
  3. Mineral
- sau đó bổ sung vào danh sách.
- ⚠ Chuyển đổi môi trường `enumerate` thành `itemize` environments, định nghĩa lại các nhãn.

## Tạo bảng

- Các đoạn văn bản có thể được sắp xếp thành hàng và cột sử dụng *môi trường* **tabular** :

`\begin{tabular}{column specifiers}`

Mỗi cột được biểu diễn bằng một trong các chữ cái: l (canh trái)  
c (canh giữa) or r (canh phải). Ví dụ:

1. `\begin{tabular}{ccc}`

3 cột canh giữa.

2. `\begin{tabular}{lr}`

Hai cột, cột đầu canh trái, cột 2 canh phải.

- Trong môi trường **tabular**:
  - Ký hiệu **đặc biệt** & tạo cột mới.
  - Dòng mới được tạo bởi `\\`

## Ví dụ tạo bảng

- Bảng có 2 cột
- Mỗi dòng chỉ có 1 &
- Cột 1 canh trái, cột 2 canh giữa.

```
\begin{tabular}{lc}
Item & Cost\\
CD & \pounds 11.75\\
Video & \pounds 14.10\\
Total & \pounds 25.85
\end{tabular}
```

INPUT

| Item  | Cost   |
|-------|--------|
| CD    | £11.75 |
| Video | £14.10 |
| Total | £25.85 |

OUTPUT

Đừng nhầm 1 (ell) với 1 (one)!

## Thêm các đường kẻ

- Các đường kẻ đứng được hiển thị trong môi trường `tabular` bằng ký hiệu số đứng `|`

```
\begin{tabular}{|l|c|}
```

- Kẻ ngang:
  - Trên tất cả các cột:

```
\hline
```

- Từ cột *n* đến *m*:

```
\cline{n-m}
```

- `\hline` `\cline` chỉ được dùng ở đầu dòng.



## Ví dụ

Ví dụ trên có thể được biến đổi thành:

```
\begin{tabular}{|l|c|}
\hline
Item & Cost\\
\hline\hline
CD & \pounds 11.75\\
Video & \pounds 14.10\\
\hline
Total & \pounds 25.85\\
\hline
\end{tabular}
```

INPUT

| Item  | Cost   |
|-------|--------|
| CD    | £11.75 |
| Video | £14.10 |
| Total | £25.85 |

OUTPUT

## Các ô nhiều cột

- Một ô có thể chiếm nhiều cột:

`\multicolumn{cols}{align}{text}`

- *cols* số lượng cột cần cho ô.
- *align* canh lề của ô. Chỉ có thể có một chữ cái(v.d. c), nhưng có thể có ký hiệu số đứng.
- *text* Nội dung của ô.
- Lệnh này có thể được sử dụng để thay thế các tính chất của một ô.

```

\begin{tabular}{|l|cc|}\hline
& \multicolumn{2}{c|}{Cost}\\
Item & ex VAT & inc VAT (@17.5\%)\\\hline
CD & \pounds 10.00 & \pounds 11.75\\
Video & \pounds 12.00 & \pounds 14.10\\\hline
\multicolumn{1}{l|}{Total} & \pounds 22.00 & \pounds 25.85\\\cline{2-3}
\end{tabular}

```

INPUT

| Item  | Cost   |                  |
|-------|--------|------------------|
|       | ex VAT | inc VAT (@17.5%) |
| CD    | £10.00 | £11.75           |
| Video | £12.00 | £14.10           |
| Total | £22.00 | £25.85           |

OUTPUT

## Bài tập 8 : Tạo bảng (trang 9)

- Khi tạo bảng, nên tạo từ một bảng đơn giản rồi thêm các ô vào bảng đó.
  - ④ Tạo văn bản giống hình 6 trang 9
  - ⚠ Tạo văn bản giống hình 7 trang 10.
- Tạo bảng sử dụng một *môi trường tabular*. Các dòng Equipment Expenditure và Travel Expenditure dùng cả 5 columns.
- Sau đó, canh lề giữa tất cả dùng lệnh \centerline.

## Công thức toán căn bản

- T<sub>E</sub>X có hai chế độ:
  1. Chế độ soạn thảo văn bản
  2. Chế độ gõ công thức toán
- Cho đến đây chúng ta chỉ dùng chế độ văn bản.
- Tất cả các công thức toán đều phải đặt trong chế độ gõ toán học. Các công thức toán học được định dạng khác với các đoạn văn bản.
  - Chế độ văn bản:  $y-c=mx$
  - Chế độ công thức:  $y - c = mx$
- Có hai loại công thức: *in-line* (công thức trong dòng) công thức đứng riêng

## Công thức trong dòng

- Công thức trong dòng hiển thị trong dòng của một đoạn văn.
- Các ký hiệu được hiển thị nhỏ hơn cho phù hợp với vị trí trong dòng. .
- Công thức có thể được hiển thị trên nhiều dòng khác nhau.
- Khai báo công thức trong dòng bằng môi trường `math`:

```
\begin{math}
```

```
...
```

```
\end{math}
```

- Để cho thuận tiện, môi trường chế độ toán được thay bằng: `\(`  
`... \)`
- Và gọn hơn nữa: `$ ... $`

Một ký hiệu đặc biệt khác **ký hiệu đặc biệt!**

## Ví dụ

I can refer to the variable  $x$ , or the formula  $y = mx + c$ .

INPUT

I can refer to the variable  $x$ , or the formula  $y = mx + c$ .

OUTPUT

The  $i$ th element of the vector  $\vec{a}$  has the value  $2i$  for  $i = 1 \dots m$ .

INPUT

The  $i$ th element of the vector  $\vec{a}$  has the value  $2i$  for  $i = 1 \dots m$ .

OUTPUT

## Các công thức riêng rẽ

- Công thức riêng rẽ được gửi trong một dòng riêng biệt.
- Các ký hiệu  $\sum$  được hiển thị lớn hơn.
- Rất nhiều môi trường khai báo chế độ toán Lots of different
- Một vài môi trường cơ bản:
  1. Môi trường `displaymath` hiển thị *một* công thức riêng không đánh số.  

$$\backslash\mathrm{begin}\{\mathrm{displaymath}\} \dots \backslash\mathrm{end}\{\mathrm{displaymath}\}$$

Cú pháp viết gọn: `\[ ... \]`
  2. Môi trường `equation` hiển thị *single* công thức riêng đánh số.  

$$\backslash\mathrm{begin}\{\mathrm{equation}\} \dots \backslash\mathrm{end}\{\mathrm{equation}\}$$



## Ví dụ

The function

```
\begin{displaymath}
```

$$f(x) = 4x + 1$$

```
\end{displaymath}
```

is linear

INPUT

The function

$$f(x) = 4x + 1$$

is linear

OUTPUT

The function

```
\begin{equation}
```

$$f(x) = 4x + 1$$

```
\end{equation}
```

is linear

INPUT

The function

$$f(x) = 4x + 1 \quad (1)$$

is linear

OUTPUT

## Chỉ số và số mũ

- Chỉ số được hiển thị bằng lệnh: `\sb{subscript}`
  - Viết gọn: `_{\i{subscript}}`
  - Ví dụ: `$a\sb{0}$` hoặc `$a_{0}$` hoặc `$a_0$` đều cho kết quả:  
 $a_0$
- Số mũ được tạo ra bằng: `\sp{superscript}`
  - Viết gọn: `^{\i{superscript}}`
  - Ví dụ: `$x\sp{2}$` or `$x^{\i{2}}$` or `$x^2$` all produce:  $x^2$
- Chỉ số và số mũ có thể được gộp lại.  
Ví dụ: `$a_0^2$` produces  $a_0^2$

## Ví dụ

A quadratic function:

$$\backslash f(x) = a_0 + a_1 x + a_2 x^2 \backslash$$

INPUT

A quadratic function:

$$f(x) = a_0 + a_1 x + a_2 x^2$$

OUTPUT

Compare  $a_b^c$  with  $a_{\{b^c\}}$ .

INPUT

Compare  $a_b^c$  with  $a_{b^c}$ .

OUTPUT

## Căn và phân số

- Phân số có thể được hiển thị sử dụng:

`\frac{numerator}{denominator}`

- Căn có thể được hiển thị sử dụng:

`\sqrt[n]{maths}`

- Ví dụ:

```
\begin{displaymath}
f(x_1, x_2) = x_1^2
+ e^{\{x_2\}} +
\frac{\sqrt[3]{a}}{1+\sqrt{x_2}}
\end{displaymath}
```

INPUT

$$f(x_1, x_2) = x_1^2 + e^{x_2} + \frac{\sqrt[3]{a}}{1 + \sqrt{x_2}}$$

OUTPUT

## Tên hàm

|                      |                      |                      |                   |                    |                    |
|----------------------|----------------------|----------------------|-------------------|--------------------|--------------------|
| <code>\arccos</code> | <code>\arcsin</code> | <code>\arctan</code> | <code>\arg</code> | <code>\cos</code>  | <code>\cosh</code> |
| <code>\cot</code>    | <code>\coth</code>   | <code>\csc</code>    | <code>\deg</code> | <code>\det</code>  | <code>\dim</code>  |
| <code>\exp</code>    | <code>\gcd</code>    | <code>\hom</code>    | <code>\inf</code> | <code>\ker</code>  | <code>\lg</code>   |
| <code>\lim</code>    | <code>\liminf</code> | <code>\limsup</code> | <code>\ln</code>  | <code>\log</code>  | <code>\max</code>  |
| <code>\min</code>    | <code>\Pr</code>     | <code>\sec</code>    | <code>\sin</code> | <code>\sinh</code> | <code>\sup</code>  |
| <code>\tan</code>    | <code>\tanh</code>   |                      |                   |                    |                    |

**Không đúng:**  $\$ \log x \$$        $\log x$

**Đúng:**  $\$ \backslash \log x \$$        $\log x$

Các lệnh **màu xanh** là các lệnh có cận trên và cận dưới, được hiển thị bằng `_`.

## Ví dụ

```
\begin{displaymath}
\exp(ix) = \sin(x) + i\cos(x)
\end{displaymath}
```

INPUT

$$\exp(ix) = \sin(x) + i \cos(x)$$

OUTPUT

```
\begin{displaymath}
\lim_{x \rightarrow 0} f(x)
\end{displaymath}
```

INPUT

$$\lim_{x \rightarrow 0} f(x)$$

OUTPUT

## Các chữ cái Hy Lạp thường

|                        |             |                       |            |                          |               |
|------------------------|-------------|-----------------------|------------|--------------------------|---------------|
| <code>\alpha</code>    | $\alpha$    | <code>\beta</code>    | $\beta$    | <code>\gamma</code>      | $\gamma$      |
| <code>\delta</code>    | $\delta$    | <code>\epsilon</code> | $\epsilon$ | <code>\varepsilon</code> | $\varepsilon$ |
| <code>\zeta</code>     | $\zeta$     | <code>\eta</code>     | $\eta$     | <code>\theta</code>      | $\theta$      |
| <code>\vartheta</code> | $\vartheta$ | <code>\iota</code>    | $\iota$    | <code>\kappa</code>      | $\kappa$      |
| <code>\lambda</code>   | $\lambda$   | <code>\mu</code>      | $\mu$      | <code>\nu</code>         | $\nu$         |
| <code>\xi</code>       | $\xi$       | <code>\pi</code>      | $\pi$      | <code>\varpi</code>      | $\varpi$      |
| <code>\rho</code>      | $\rho$      | <code>\varrho</code>  | $\varrho$  | <code>\sigma</code>      | $\sigma$      |
| <code>\varsigma</code> | $\varsigma$ | <code>\tau</code>     | $\tau$     | <code>\upsilon</code>    | $\upsilon$    |
| <code>\phi</code>      | $\phi$      | <code>\varphi</code>  | $\varphi$  | <code>\chi</code>        | $\chi$        |
| <code>\psi</code>      | $\psi$      | <code>\omega</code>   | $\omega$   |                          |               |

## Các chữ cái Hy Lạp hoa

|                      |           |                       |            |                     |          |
|----------------------|-----------|-----------------------|------------|---------------------|----------|
| <code>\Gamma</code>  | $\Gamma$  | <code>\Delta</code>   | $\Delta$   | <code>\Theta</code> | $\Theta$ |
| <code>\Lambda</code> | $\Lambda$ | <code>\Xi</code>      | $\Xi$      | <code>\Pi</code>    | $\Pi$    |
| <code>\Sigma</code>  | $\Sigma$  | <code>\Upsilon</code> | $\Upsilon$ | <code>\Phi</code>   | $\Phi$   |
| <code>\Psi</code>    | $\Psi$    | <code>\Omega</code>   | $\Omega$   |                     |          |

Ví dụ:

```
\begin{displaymath}
e^{i\theta} = \cos\theta
+ i\sin\theta
\end{displaymath}
```

INPUT

$$e^{i\theta} = \cos \theta + i \sin \theta$$

OUTPUT



## Các ký hiệu có hai cỡ

Các ký hiệu sau có hai cỡ phụ thuộc vào chế độ công thức:

|                         |           |              |                        |          |             |                        |          |             |
|-------------------------|-----------|--------------|------------------------|----------|-------------|------------------------|----------|-------------|
| <code>\bigcap</code>    | $\cap$    | $\bigcap$    | <code>\bigcup</code>   | $\cup$   | $\bigcup$   | <code>\bigodot</code>  | $\odot$  | $\bigodot$  |
| <code>\bigotimes</code> | $\otimes$ | $\bigotimes$ | <code>\bigoplus</code> | $\oplus$ | $\bigoplus$ | <code>\bigsqcup</code> | $\sqcup$ | $\bigsqcup$ |
| <code>\biguplus</code>  | $\uplus$  | $\biguplus$  | <code>\bigvee</code>   | $\vee$   | $\bigvee$   | <code>\bigwedge</code> | $\wedge$ | $\bigwedge$ |
| <code>\coprod</code>    | $\coprod$ | $\coprod$    | <code>\int</code>      | $\int$   | $\int$      | <code>\oint</code>     | $\oint$  | $\oint$     |
| <code>\prod</code>      | $\prod$   | $\prod$      | <code>\sum</code>      | $\sum$   | $\sum$      |                        |          |             |

## Ví dụ

```
\begin{displaymath}
f(x) = \sum_{i=0}^n
\alpha_i x^i
\end{displaymath}
```

INPUT

$$f(x) = \sum_{i=0}^n \alpha_i x^i$$

OUTPUT

In text :

```
\begin{math}
f(x) = \sum_{i=0}^n
\alpha_i x^i
\end{math}
```

INPUT

In text :  $f(x) = \sum_{i=0}^n \alpha_i x^i$

OUTPUT

## Bài tập 9 : Căn bản công thức toán (trang 10)

- ④ Tạo ra văn bản giống hình vẽ 8 trang 10 của handouts.
- ⚠ – Tạo ra văn bản giống hình vẽ 9 trang 11 của handouts.
  - Chú ý gõ  $f(x)$ ,  $f$  and  $x$  ở chế độ công thức.

Thử đánh số các công thức.

## Dấu phân cách

- Các dấu ngoặc khi đặt cạnh căn, phân số không thay đổi kích thước theo nội dung. Ví dụ:

```
\begin{displaymath}
(\frac{1}{1+x})
\end{displaymath}
```

INPUT

$$\left(\frac{1}{1+x}\right)$$

OUTPUT

- Khi đó, dùng các câu lệnh sau sẽ đẹp hơn:  
`\left`*delimiter* and `\right`*delimiter*
- Chú ý luôn luôn phải đảm bảo đủ cặp `\left` và `\right`, mặc dù các ngoặc đóng và ngoặc mở có thể khác nhau.

## Dấu phân cách

|              |   |              |   |          |         |            |         |
|--------------|---|--------------|---|----------|---------|------------|---------|
| (            | ( | )            | [ | [        | ]       | ]          |         |
| \{           | { | \}           | } |          |         | \          |         |
| /            | / | \backslash   | \ | \langle  | \rangle | \rangle    | \rangle |
| \lfloor      |   | \rfloor      |   | \lceil   |         | \rceil     |         |
| \uparrow     | ↑ | \downarrow   | ↓ | \Uparrow | ⇑       | \Downarrow | ⇓       |
| \updownarrow | ↕ | \Updownarrow | ↕ |          |         |            |         |

Nếu muốn ngoặc đóng hoặc ngoặc mở không hiển thị, sử dụng dấu phân cách là . (dấu chấm) .

## Ví dụ

```
\begin{displaymath}
\left(
\frac{1}{1+x}
\right)
\end{displaymath}
```

INPUT

$$\left( \frac{1}{1+x} \right)$$

OUTPUT

```
\begin{displaymath}
\left|
\frac{1}{1+x}
\right|
\end{displaymath}
```

INPUT

$$\left| \frac{1}{1+x} \right|$$

OUTPUT

## Mảng

- Mảng có thể được tạo ra sử dụng *môi trường array*.
- Môi trường này giống môi trường *tabular*, nhưng ở trong chế độ công thức toán.
- Các phần tử được sắp xếp theo dòng và theo cột để tạo thành các cấu trúc như mảng và vector.

```
\begin{displaymath}
\begin{array}{cc}
0 & 1 \\
2 & 3
\end{array}
\end{displaymath}
```

INPUT

$$\begin{array}{cc} 0 & 1 \\ 2 & 3 \end{array}$$

OUTPUT

```
\begin{displaymath}
\left (
\begin{array}{cc}
0 & 1 \\
2 & 3
\end{array}
\right)
\end{displaymath}
```

INPUT

$$\left ( \begin{array}{cc} 0 & 1 \\ 2 & 3 \end{array} \right )$$

OUTPUT



## Các dấu ngoặc không cần theo cặp

```
\begin{displaymath}
\left[
\begin{array}{cc}
0 & 1 \\
2 & 3
\end{array}
\right]
\end{displaymath}
```

INPUT

$$\left[ \begin{array}{cc} 0 & 1 \\ 2 & 3 \end{array} \right]$$

OUTPUT

## Ví dụ sử dụng các dấu ngoặc không hiển thị

```
\begin{displaymath}
f(x) =
\left \{
\begin{array}{cl}
0 & x \leq 0 \\
1 & x > 0
\end{array}
\right .
\end{displaymath}
```

INPUT

$$f(x) = \begin{cases} 0 & x \leq 0 \\ 1 & x > 0 \end{cases}$$

OUTPUT

## Bài tập 10 : Mảng (trang 12)

- ④ Tạo ra đầu ra như trong hình 10 on page 12 của handouts.
- ⚠ – Tạo ra văn bản giống như hình 11 on page 12 của handouts.
- Cần dùng các câu lệnh sau:

`\cdots`     ...

`\vdots`     ⋮

`\ddots`     ⋱

`\neq`      $\neq$

## Multiline Formulæ-Các công thức nhiều dòng

- Môi trường `displaymath` và môi trường `equation` chỉ cho phép một dòng.
- Môi trường `eqnarray` cho phép nhiều công thức có thể kết hợp với nhau.
- Môi trường `eqnarray` có 3 cột: cột đầu canh lề phải, cột 2 canh lề giữa, cột 3 canh lề trái.
- Mỗi dòng trong `eqnarray` được đánh số.
- Môi trường `eqnarray*` không được đánh số.
- Để xóa số công thức trong môi trường `eqnarray`, dùng `\nonumber` trên dòng tương ứng.

```
\begin{eqnarray}
\ln(f(x)) & = & x^2 + \frac{1}{x+3} \\
f(x) & = & \exp \left(x^2 + \frac{1}{x+3} \right)
\end{eqnarray}
```

INPUT

$$\ln(f(x)) = x^2 + \frac{1}{x+3} \quad (2)$$

$$f(x) = \exp \left( x^2 + \frac{1}{x+3} \right) \quad (3)$$

OUTPUT

```
\begin{eqnarray}
\ln(f(x)) & = & x^2 + \frac{1}{x+3} \quad \text{\nonumber} \\
f(x) & = & \exp \left(x^2 + \frac{1}{x+3} \right) \\
\end{eqnarray}
```

INPUT

$$\begin{aligned} \ln(f(x)) &= x^2 + \frac{1}{x+3} \\ f(x) &= \exp \left( x^2 + \frac{1}{x+3} \right) \end{aligned} \quad (4)$$

OUTPUT

## Bài tập 11 : Công thức nhiều dòng (trang 12)

④ Tạo ra văn bản theo hình 12 trang 13 trong handouts.

⚠ Tạo ra văn bản theo hình 13 trang 13 trong handouts.

– Cần sử dụng các lệnh sau:

|                          |               |                   |        |
|--------------------------|---------------|-------------------|--------|
| <code>\approx</code>     | $\approx$     | <code>\pm</code>  | $\pm$  |
| <code>\partial</code>    | $\partial$    | <code>\leq</code> | $\leq$ |
| <code>\varepsilon</code> | $\varepsilon$ |                   |        |

## Tham chiếu chéo

- Gán nhãn cho một vị trí trong văn bản bằng `\label{string}`

**Ví dụ:**

```
\section{Introduction}
\label{sec:intro}
```

**Ví dụ:**

```
\begin{equation}
E = mc^2
\label{eqn:einstein}
\end{equation}
```

- Tham chiếu tới vị trí đã gán nhãn sử dụng `\ref{string}`.
- Tham chiếu tới trang chứa nhãn bằng `\pageref{string}`.



## Ví dụ

```
\section{Introduction}
\label{sec:intro}
\ldots
See Section~\ref{sec:intro}
for a brief introduction.
```

INPUT

```
See subsection~\ref{sec:ex}
for examples.
```

```
\subsection{Examples}
\label{sec:ex}
```

INPUT

**1 Introduction**  
 ...See Section 1 for  
 a brief introduction.

OUTPUT

See subsection 2.3  
 for examples.

**2.3 Examples**

OUTPUT

## Ví dụ

```
See Appendix~\ref{apd:tables}
for tables\ldots
\appendix
\section{Tables}\label{apd:tables}
```

INPUT

See Appendix A for  
tables...

**A   Tables**

OUTPUT

```
\begin{equation}
\label{eqn:Emc}
E = mc^2
\end{equation}
\ldots
See Equation~\ref{eqn:Emc}
on page~\pageref{eqn:Emc}.
```

INPUT

$$E = mc^2 \quad (5)$$

...See Equation 5 on  
page 201.

OUTPUT

## Bài tập 12 (trang 13)

- Tạo ra văn bản như trong hình 14 trang 14 trong handouts sử dụng `\label` và `\ref`. Cần nhớ lại các kiến thức làm thế nào để:
  - Tạo ra **sections**
  - **emphasize** một đoạn văn
  - Tạo ra **các công thức đánh số**
  - Tạo ra **các công thức tại chỗ (trong dòng)**
- ⚠ Đồng thời thử chèn thêm một section mới giữa các phần giới thiệu và định luận Bayes, một công thức khác, để có thể thấy L<sup>A</sup>T<sub>E</sub>X cập nhật các tham chiếu chéo như thế nào.
- Thêm **tiêu đề** và **mục lục**

## Gói

- *Gói* là các tệp với phần mở rộng `.sty`
- Gói có thể định nghĩa mới/lại các lệnh.
- Sử dụng một gói:

```
\usepackage[options]{package-name}
```

- Lệnh này chỉ có thể được dùng trong phần *preamble-phần khai báo*.

## Ví dụ

- Gói `babel` định nghĩa các tên theo các tiếng khác nhau.
- `ukdate` định nghĩa lại lệnh `\today`, cũng như `datetime`.
- `xr` định nghĩa lệnh mới

`\externaldocument{filename}`

cho phép tham chiếu tới các nhãn nằm trong một file khác.

- `varioref` định nghĩa lệnh `\vref` giống `\ref` nhưng bổ sung thêm số trang. V.d. `Fig~\vref{fig:ex}` can sẽ cho: “Fig 2 on page 42”, hoặc “Fig 2 on the following page” hoặc “Fig 2”.

## Ví dụ.

- **hyperref** định nghĩa các lệnh cho phép có các link trong văn bản PDF.

V.d.

```
\href{http://www.tex.ac.uk/}{%
the \TeX\ Archive}
```

INPUT

the  $\text{\TeX}$  Archive

OUTPUT

- **ifpdf** định nghĩa cấu trúc điều kiện `\ifpdf` để xác định  $\text{\LaTeX}$  hay là  $\text{PDF}\text{\LaTeX}$  đang được dùng. V.d.

```
\ifpdf
A PDF\LaTeX\ document
\else
A \LaTeX\ document
\fi
```

INPUT

A  $\text{PDF}\text{\LaTeX}$  document

OUTPUT

## Ví dụ

```
\documentclass[a4paper]{article}
\begin{document}
\today
\end{document}
```

INPUT

Ngày 15 tháng 1  
năm 2021

OUTPUT

```
\documentclass[a4paper]{article}
\usepackage[short]{datetime}
\begin{document}
\today
\end{document}
```

INPUT

Ngày 15 tháng 1  
năm 2021

OUTPUT


## Tài liệu tự cài đặt

- *Các gói* chưa có trong cài đặt  $\text{T}_{\text{E}}\text{X}$  có thể được tải từ *Kho tài nguyên  $\text{T}_{\text{E}}\text{X}$* .
- Gói+tài liệu có thể được đóng gói trong tệp với phần mở rộng `.dtx`
- Gói cần được phổ biến với kịch bản cài đặt(`.ins`)
- Tài liệu sử dụng có thể được sinh ra bằng cách  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  `.dtx`. Ví dụ:  
▶ `latex datetime.dtx`
- Các gói có thể được giải nén bằng cách  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  kịch bản cài đặt. Ví dụ:  
▶ `latex datetime.ins`



## Bài tập 13 (trang 15)

- Trở lại tệp `sectioning.tex` trong bài tập 4
  - Sử dụng gói `babel`. Có nhiều tùy biến cho ngôn ngữ kiểm tra nội dung của thư mục:  
`c:\texmf\tex\generic\babel\`  
Các option là các tên file `.sty`.
- Hoặc trở lại tài liệu tạo ra trong bài tập 12.
  - Dùng `hyperref` để các tham chiếu thành các link, dùng `PDFLATEX` để tạo file PDF.

 Để thử nghiệm việc extract tài liệu và mã từ `.dtx` file, tải gói `datetime` package từ <http://theoval.cmp.uea.ac.uk/~nlct/latex/packages/index.html>

## Trích dẫn tài liệu

- Môi trường `thebibliography`

```
\begin{thebibliography}{2}
```

```
\bibitem{clarke83} G. M. Clarke and D. Cooke.
```

```
\emph{A basic course in statistics}.
```

```
Chapman and Hall, 2nd edition, 1983.
```

```
\bibitem{goossens93} M. Goossens and F. Mittelbach.
```

```
\emph{The \LaTeX\ companion}.
```

```
Addison-Wesley, 1993.
```

```
\end{thebibliography}
```

- Sử dụng `\cite[text]{key-list}` để trích dẫn một tài liệu tham khảo.

## Ví dụ

```
See Goossens \emph{et
al.}~\cite{goossens93}
\ldots
```

```
\begin{thebibliography}{1}
\bibitem{goossens93}
M. Goossens and
F. Mittelbach.
\emph{The \LaTeX\
companion}.
Addison-Wesley, 1993.
\end{thebibliography}
```

INPUT

See Goossens *et al.* [1]  
...

### References

- [1] M. Goossens and  
F. Mittelbach. *The  
 $\LaTeX$  companion*.  
Addison-Wesley,  
1993.

OUTPUT

## BibT<sub>E</sub>X

Dùng BibT<sub>E</sub>X để sinh ra [thebibliography](#).

- CSDL lớn (.bib) có nhiều tài liệu tham khảo.
- BibT<sub>E</sub>X chỉ lựa chọn các tài liệu được trích dẫn.
- Các bản ghi được sắp xếp.
- Và được định dạng giống nhau.

## CSDL Tài liệu tham khảo (.bib)

```
@entry type{keyword,
 field = "text",
 :
 field = "text"
}
```

```
@book{kreyszig88,
 author = "Kreyszig, Erwin",
 title = "Advanced Engineering Mathematics",
 publisher = "Wiley",
 edition = "6th",
 year = 1988
}
```

## Định dạng tên tác giả

Tên tác giả cần được hiển thị theo một trong các dạng sau đây:

- *forenames von surname*
- *von surname, forenames*
- *von surname, jr, forenames*

Ví dụ:

Dữ liệu

"Alex Thomas von Neumann"

"John Chris {Smith Jones}"

"van de Klee, Mary-Jane"

"Smith, Jr, Fred John"

"Maria {\uppercase{d}e La} Cruz"

Compare last example with:

"Maria De La Cruz"

Hiển thị ("**abbrv**" style)

A.T. von Neumann

J.C. Smith Jones

M.-J. van de Klee

F.J. Smith, Jr

M. De La Cruz

M. D. L. Cruz (Incorrect!)

## Nhiều tác giả

Trong tài liệu nhiều tác giả, tên các tác giả phân chia nhau bằng từ khóa and

```
@book{goossens97,
 author = "Goossens, Michel and Rahtz, Sebastian and
 Mittelbach, Frank",
 title = "The \LaTeX\ graphics companion: illustrating
 documents with \TeX\ and {PostScript}",
 publisher = "Addison Wesley Longman, Inc",
 year = 1997
}
```

## Dữ liệu tháng

- Kiểu tên tài liệu tham khảo luôn luôn có 3 chữ số cho tháng :  
jan, feb, mar, ...
- Luôn dùng cách rút gọn để đảm bảo thống nhất.

```
@inproceedings{talbot97,
 author = "Talbot, Nicola and Cawley, Gavin",
 title = "A fast index assignment algorithm for
 robust vector quantisation of image data",
 booktitle = "Proceedings of the I.E.E.E. International
 Conference on Image Processing",
 address = "Santa Barbara, California, USA",
 month = oct,
 year = 1997
}
```



## Ví dụ (incollection)

```
@incollection{wainwright,
 author = "Wainwright, Robert B.",
 title = "Hazards from {Northern} Native Foods",
 booktitle = "\emph{Clostridium botulinum}: Ecology and
 Control in Foods",
 chapter = 12,
 pages = "305--322",
 editor = "Hauschild, Andreas H. W. and Dodds,
 Karen L.",
 publisher = "Marcel Dekker, Inc",
 year = 1993
}
```

## Khai báo CSDL và kiểu tài liệu tham khảo

Trong *mã nguồn* L<sup>A</sup>T<sub>E</sub>X (.tex):

- Khai báo kiểu tài liệu tham khảo:

`\bibliographystyle{style-name}`

Các kiểu thường dùng:

*plain*    Các tài liệu được sắp xếp theo thứ tự ABC.

*unsrt*    Các tài liệu được sắp xếp theo thứ tự trích dẫn.

*alpha*    Các tài liệu được sắp xếp theo thứ tự các nhãn gồm tên tác giả và năm xuất bản.

*abbrv*    Các tài liệu được sắp xếp theo thứ tự họ, tháng, năm và tên tạp chí thu gọn.

- Khai báo CSDL:

`\bibliography{name}`

## Ví dụ

Trong filename.tex (database.bib chứa CSDL):

```
This is the document \ldots
```

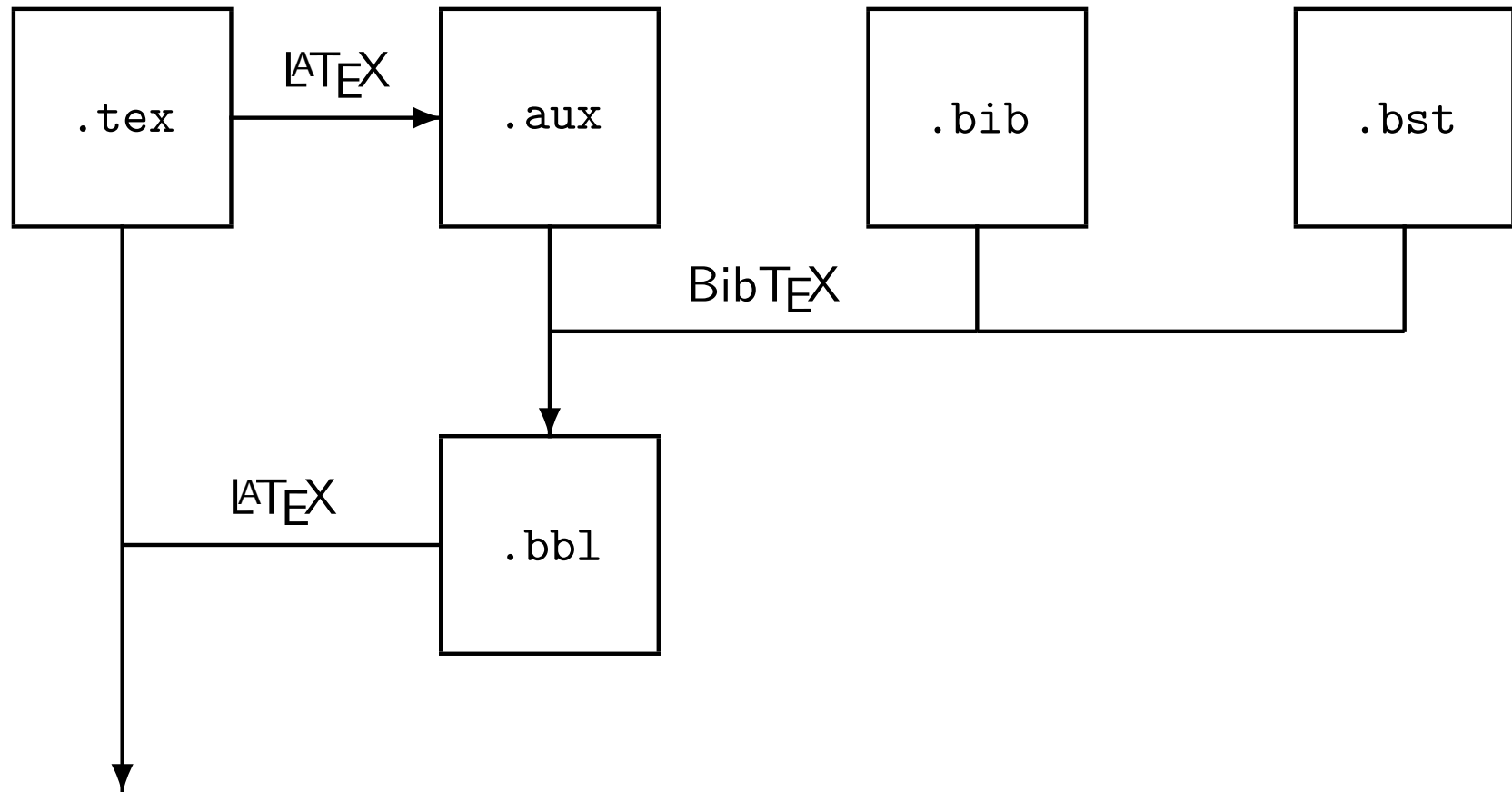
```
\bibliographystyle{plain}
```

```
\bibliography{database}
```

INPUT

TeXnicCenter tự động gọi BibTeX .

## Quá trình $\text{\LaTeX}$ /Bib $\text{\TeX}$



\bibliographystyle{plain}

- [1] Gavin C. Cawley and Nicola L. C. Talbot. A fast index assignment algorithm for vector quantization over noisy transmission channels. *I.E.E. Electronic Letters*, 32(15):1343–1344, July 1996.
- [2] Erwin Kreyszig. *Advanced Engineering Mathematics*. Wiley, 6th edition, 1988.
- [3] Nicola L. C. Talbot and Gavin C. Cawley. A quadratic index assignment algorithm for vector quantisation over noisy transmission channels. In *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, volume 18, pages 195–199, November 1996.
- [4] Nicola L. C. Talbot and Gavin C. Cawley. A fast index assignment algorithm for robust vector quantisation of image data. In *Proceedings of the I.E.E.E. International Conference on Image Processing*, Santa Barbara, California, USA, October 1997.

## \bibliographystyle{alpha}

- [CT96] Gavin C. Cawley and Nicola L. C. Talbot. A fast index assignment algorithm for vector quantization over noisy transmission channels. *I.E.E. Electronic Letters*, 32(15):1343–1344, July 1996.
- [Kre88] Erwin Kreyszig. *Advanced Engineering Mathematics*. Wiley, 6th edition, 1988.
- [TC96] Nicola L. C. Talbot and Gavin C. Cawley. A quadratic index assignment algorithm for vector quantisation over noisy transmission channels. In *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, volume 18, pages 195–199, November 1996.
- [TC97] Nicola L. C. Talbot and Gavin C. Cawley. A fast index assignment algorithm for robust vector quantisation of image data. In *Proceedings of the I.E.E.E. International Conference on Image Processing*, Santa Barbara, California, USA, October 1997.

\bibliographystyle{ieeetr}

- [1] E. Kreyszig, *Advanced Engineering Mathematics*. Wiley, 6th ed., 1988.
- [2] N. L. C. Talbot and G. C. Cawley, “A quadratic index assignment algorithm for vector quantisation over noisy transmission channels,” in *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, vol. 18, pp. 195–199, Nov. 1996.
- [3] G. C. Cawley and N. L. C. Talbot, “A fast index assignment algorithm for vector quantization over noisy transmission channels,” *I.E.E. Electronic Letters*, vol. 32, pp. 1343–1344, July 1996.
- [4] N. L. C. Talbot and G. C. Cawley, “A fast index assignment algorithm for robust vector quantisation of image data,” in *Proceedings of the I.E.E.E. International Conference on Image Processing*, (Santa Barbara, California, USA), Oct. 1997.

`\bibliographystyle{acm}`

- [1] CAWLEY, G. C., AND TALBOT, N. L. C. A fast index assignment algorithm for vector quantization over noisy transmission channels. *I.E.E. Electronic Letters* 32, 15 (July 1996), 1343–1344.
- [2] KREYSZIG, E. *Advanced Engineering Mathematics*, 6th ed. Wiley, 1988.
- [3] TALBOT, N. L. C., AND CAWLEY, G. C. A quadratic index assignment algorithm for vector quantisation over noisy transmission channels. In *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing* (Nov. 1996), vol. 18, pp. 195–199.
- [4] TALBOT, N. L. C., AND CAWLEY, G. C. A fast index assignment algorithm for robust vector quantisation of image data. In *Proceedings of the I.E.E.E. International Conference on Image Processing* (Santa Barbara, California, USA, Oct. 1997).



`\bibliographystyle{apalike}`

Kiểu này cần gói `apalike`.

Cawley, G. C. and Talbot, N. L. C. (1996) A fast index assignment algorithm for vector quantization over noisy transmission channels. *I.E.E. Electronic Letters*, 32(15):1343–1344.

Kreyszig, E. (1988) *Advanced Engineering Mathematics*. Wiley, 6th edition.

Talbot, N. L. C. and Cawley, G. C. (1996). A quadratic index assignment algorithm for vector quantisation over noisy transmission channels. In *Proceedings of the Institute of Acoustics Autumn Conference on Speech and Hearing*, volume 18, pages 195–199.

Talbot, N. L. C. and Cawley, G. C. (1997). A fast index assignment algorithm for robust vector quantisation of image data. In *Proceedings of the I.E.E.E. International Conference on Image Processing*, Santa Barbara, California, USA.

## Tùy biến BiBTeX

- Có thể tạo một kiểu BibTeX (.bst) khác so với các kiểu chuẩn từ một trong các kiểu chuẩn bằng cách copy file cũ vào tên mới, rồi thay đổi.
- Chú ý: BibTeX là ngôn ngữ bậc thấp, không dành cho người yếu tim!
- Giải pháp thay thế: makebst. makebst.tex là một kịch bản T<sub>E</sub>X/L<sub>A</sub>T<sub>E</sub>X có thể dùng để tạo ra một tệp .bst. Chỉ cần gõ:  
► `latex makebst`  
trong MS-DOS , và theo các chỉ dẫn.

## Bài tập 14 (trang 16)

- Lập (JabRefs) 1 CSDL BibT<sub>E</sub>X chứa các tài liệu trong hình 15 trang 17 của handouts, tạo ra tài liệu như trogn hình vẽ.
  - Thay đổi các kiểu bibliography để hiển thị theo thứ tự trích dẫn. (Cần có kiểu `unsrt`). Thử các kiểu khác như `alpha`, `abbrv` và `acm` để thấy sự khác biệt.
  - Nếu có một số các trích dẫn [3,2,4] có thể in theo kiểu [2–4]. Gói `citesort` định nghĩa lại câu lệnh `\cite` để làm việc đó. Thử dùng gói với `unsrt`.
- ⚠ Nếu có thể, thử tạo một kiểu bibliography style của mình bằng `makebst`.

## Lengths- Các chiều dài

- $\text{\LaTeX}$  có các lệnh biểu diễn các chiều dài, ví dụ như `\textwidth`.
- Có hai loại chiều dài: *rigid-tuyệt đối* and *rubber-tương đối*.
- Rigid length-chiều dài tuyệt đối có giá trị cố định, ví dụ `4in`.
- Rubber length-Chiều dài tương đối có giá trị theo khoảng, ví dụ: `2in plus 0.1in minus 0.1in`.  $\text{\LaTeX}$  sẽ tính toán và tìm ra giá trị thích hợp.
- Hầu hết các chiều dài trong  $\text{\LaTeX}$  là các chiều dài cố định.

## Các đơn vị thường dùng

|    |                                                              |
|----|--------------------------------------------------------------|
| pt | Point ( $\frac{1}{72.27}$ in)                                |
| bp | Big point-điểm lớn, point của PostScript( $\frac{1}{72}$ in) |
| mm | Millimetre (2.845pt)                                         |
| cm | Centimetre (28.45pt)                                         |
| in | Inch (25.4mm)                                                |
| ex | Chiều cao của chữ x thường trong font hiện tại               |
| em | Chiều rộng của chữ M hoa trong font hiện tại                 |

## Thay đổi chiều dài

- Giá trị mới của một chiều dài được khai báo bởi:

```
\setlength{cmd}{length}
```

Ví dụ:

```
\setlength{\textwidth}{6in}
```

- Tăng chiều dài bằng:

```
\addtolength{cmd}{length}
```

ví dụ để chiều rộng của đoạn văn tăng 1in:

```
\addtolength{\textwidth}{1in}
```

## Chiều dài

- Có 3 câu lệnh khác để thay thế chiều dài:

`\settowidth{cmd}{text}`

`\settoheight{cmd}{text}`

`\settodepth{cmd}{text}`

Các lệnh này đặt chiều dài *cmd* chiều rộng, chiều cao hoặc chiều sâu của *văn bản*. Đoạn văn bản hiện tại không được hiển thị.

- Để tạo ra một chiều dài mới:

`\newlengthcmd`

- Để hiển thị giá trị của một chiều dài:

`\thecmd`

## Ví dụ

```
% define new length
\newlength\mylen
% set it to the width of the text
\settowidth{\mylen}{Hello}
% Display the value
Width=\the\mylen.
```

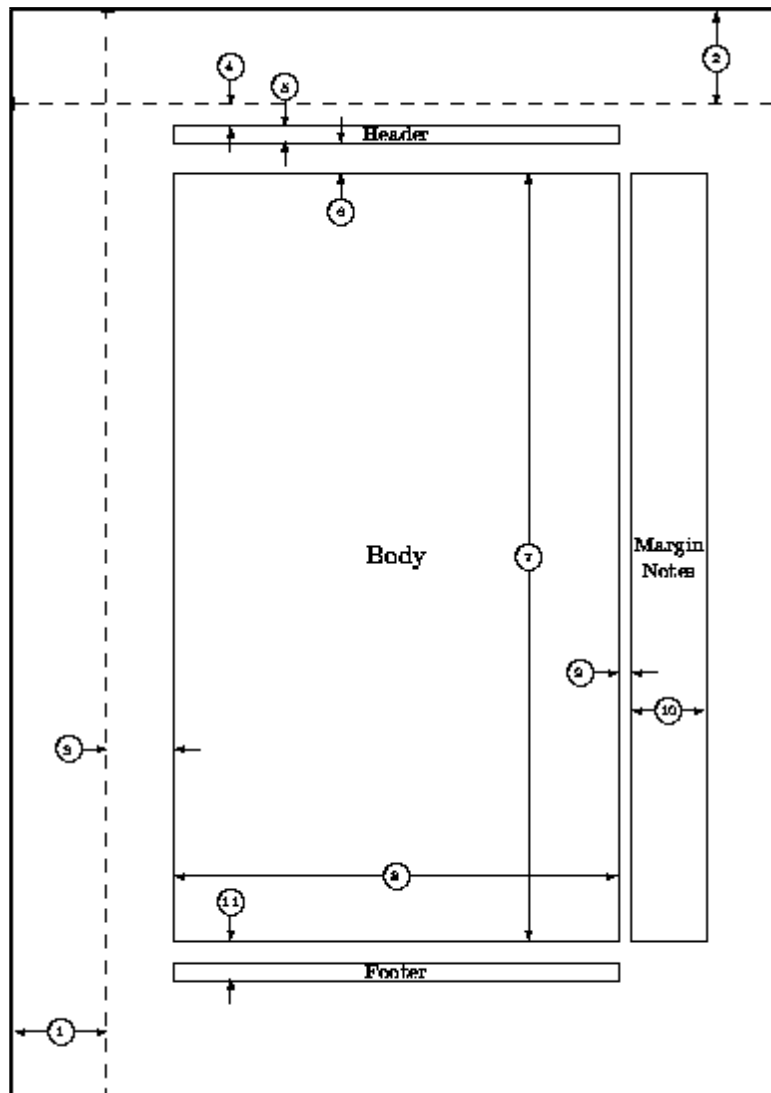
INPUT

Width=22.4945pt.

OUTPUT



# Chiều dài của Layout



- 1 one inch + `\hoffset`
- 2 one inch + `\voffset`
- 3 `\oddsidemargin`
- 4 `\topmargin`
- 5 `\headheight`
- 6 `\headsep`
- 7 `\textheight`
- 8 `\textwidth`
- 9 `\marginparsep`
- 10 `\marginparwidth`
- 11 `\footskip`

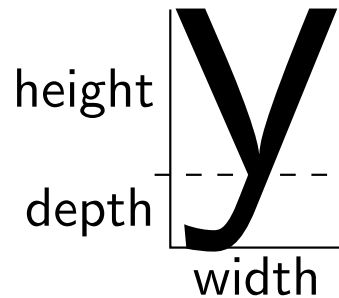
(Đồ thị sinh ra bằng cách sử dụng gói `layout`)

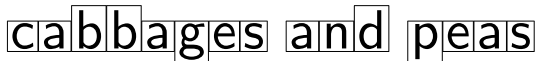
## Bài tập 15 (trang 17)

- Trở lại tài liệu trong bài tập 1
- Thay đổi chiều dài của tab đầu đoạn văn (`\parindent`) thành 0pt
- Thay đổi khoảng cách giữa các đoạn văn (`\parskip`) thành 3ex.

## Đóng khung

- Nội dung có thể được chia thành các khung nhỏ.
- Mỗi khung có chiều cao, rộng, sâu.
- Khung đơn giản nhất là một chữ cái



- Ví dụ: chuỗi ký hiệu “cabbages and peas” gồm 15 boxes:  

- Giữa các box là một loại “glue-keo”.
- Các box phức tạp được tạo thành từ các box nhỏ hơn.
- NSD cần cố định các box với nhau theo các luật về định dạng.

## Boxes

- Boxes được xử lý như các đối tượng đơn giản.
- có thể xuất hiện ở giữa dòng.
- nhưng không bao giờ có thể tách thành hai dòng.
- có thể được canh lề ngang dọc.
- Chúng ta đã sử dụng một vài loại bố.

## Ví về Box: môi trường **tabular**

Baseline

```
\begin{tabular}[c]{l}line 1\\line 2\\line 3\end{tabular}
\begin{tabular}[b]{l}line 1\\line 2\\line 3\end{tabular}
\begin{tabular}[t]{l}line 1\\line 2\\line 3\end{tabular}
```

INPUT

|          |        |        |        |
|----------|--------|--------|--------|
|          |        | line 1 |        |
|          | line 1 | line 2 |        |
| Baseline | line 2 | line 3 | line 1 |
|          | line 3 |        | line 2 |
|          |        |        | line 3 |

OUTPUT

## Các loại Box cơ bản

- `\mbox{contents}`

Đây là loại box đơn giản nhất.

- Text trong khung không bị tách dòng
- Các công thức được định dạng như text.
- Kích thước box tự thay đổi cho phù hợp với nội dung.

- `\makebox[width][alignment]{contents}`

Giống `\mbox`, với các tham số về chiều rộng, về canh lề: `left`, `right` or `c`.

## Ví dụ dùng `\mbox`

```
\begin{displaymath}
y = x \mbox{ and }
z = x + y
\end{displaymath}
```

INPUT

$$y = x \text{ and } z = x + y$$

OUTPUT

Now some in-line  
`$x = 1, \ldots, n$` maths.

INPUT

Now some in-line  $x =$   
 $1, \dots, n$  maths.

OUTPUT

Now some in-line  
`\mbox{$x = 1, \ldots, n$}`  
 maths.

INPUT

Now        some        in-line  
 $x = 1, \dots, n$  maths.

OUTPUT

## Ví dụ dùng `\makebox`

```
Here is \makebox[1in][r]{\em a 1in} box
```

INPUT

Here is *a 1in* box

OUTPUT

```
\makebox[0pt][l]{/////}Hello!
```

INPUT

~~Hello!~~

INPUT



## Boxes có khung

`\fbox` and `\framebox` : Giống `\mbox` và `\makebox`, nhưng có một khung hình chữ nhật xung quanh box.

Here is a `\fbox{box}`

INPUT

Here is a box

OUTPUT

Here is `\framebox[1in][r]{\em a 1in}` box

INPUT

Here is  *a 1in* box

OUTPUT

## Các chiều dài của `\fbox` and `\framebox`

- `\fboxsep` : Có một khoảng cách giữa khung và nội dung của box.  
Ví dụ:

```
\setlength{\fboxsep}{10pt}
\fbox{Some Text}
```



- `\fboxrule` : Chiều dài của khung. For example:

```
\setlength{\fboxrule}{4pt}
\fbox{Some Text}
```



## Gói **fancybox**

Gói **fancybox** *package* cung cấp 4 lệnh tương tự như `\fbox`:

Input

`\ovalbox{An oval frame}`

`\Ovalbox{A thicker oval frame}`

`\doublebox{A double frame}`

`\shadowbox{A shadow frame}`

Output

An oval frame

A thicker oval frame

A double frame

A shadow frame

## Ví dụ

Here is a `\ovalbox{box}`

INPUT

Here is a box

OUTPUT

Here is `\ovalbox{\makebox[1in][r]{\em a 1in}}`  
box

INPUT

Here is *a 1in* box

OUTPUT

## Ví dụ

Here is a `\ovalbox{box}`

INPUT

Here is a box

OUTPUT

Here is `\ovalbox{\makebox[1in][r]{\em a 1in}}`  
box

INPUT

Here is *a 1in* box

OUTPUT

## Ví dụ

Here is a `\doublebox{box}`

INPUT

Here is a box

OUTPUT

Here is `\doublebox{\makebox[1in][r]{\em a 1in}}`  
box

INPUT


Here is *a 1in* box

OUTPUT

## Ví dụ

Here is `\shadowbox{box}`

INPUT

Here is a 

OUTPUT

Here is `\shadowbox{\makebox[1in][r]{\em a 1in}}`  
box

INPUT

Here is  box

OUTPUT

## Gõ một đoạn văn trong Box

`\parbox[alignment][height]{width}{contents of box}`

Ví dụ:

A paragraph within a box : `\parbox{0.75in}{This box is three quarters of an inch wide}` so there!

INPUT

Một đoạn văn trong box : This box is three quarters of an inch wide so there!

OUTPUT



## Môi trường `minipage`

```
\begin{minipage}[alignment][height]{width}
```

```
Some text. \begin{minipage}{0.4\textwidth} The width of this
minipage is 0.4 times the width of the text
body\footnote{Note we can also have a footnote}.
\end{minipage} Some more text.
```

INPUT

The width of this minipage  
is 0.4 times the width of the

Some text. text body<sup>a</sup>. Some more text.

---

<sup>a</sup>Note we can also have a  
footnote

OUTPUT

## Gói `shapepar` (`\diamondpar`)



Gói

`shapepar` cho

phép tạo các hiệu

ứng đặc biệt. Có 4 kiểu

hiệu ứng đặc biệt: `diamond`,  
`square`, `heart` và `nut shaped`.

Có thể định nghĩa các loại  
khác bằng `\shapepar`.

Tham số phải là  
cả một đoạn  
văn bản.



```
\diamondpar{With the ... }
```

## Gói **shapepar** (`\squarepar`)

Với gói **shapepar** có thể có các hiệu ứng khác. Có 4 loại hiệu ứng: diamond, square, heart and nut shaped. Có thể định nghĩa các hiệu ứng khác sử dụng `\shapepar`. Tham số là cả đoạn văn bản.

```
\squarepar{With the ... }
```

## Gói **shapepar**: (`\heartpar`)

Với gói **shapepar** có thể có các hiệu ứng khác. Có 4 loại hiệu ứng: diamond, square, heart and nut shaped. Có thể định nghĩa các hiệu ứng khác sử dụng `\shapepar`.

Tham số là cả đoạn văn bản.



```
\heartpar{With the ... }
```

## Gói **shapepar**: (`\heartpar`)

Với gói **shapepar** có thể có các hiệu ứng khác. Có 4 loại hiệu ứng: `di-`, `amond`, `square`, `heart` and `nut` shaped. Có thể định nghĩa các hiệu ứng khác sử dụng `\shapepar`. Tham số là cả đoạn văn bản.

```
\shapepar\nutshape{With the ... }
```

## Làm nổi và làm chìm các Boxes

- Boxes có thể được làm chìm hoặc làm nổi sử dụng:

`\raisebox{lift}[depth][height]{contents}`

```
some text \raisebox{2ex}{some raised}
\raisebox{-1ex}{some lowered}
```

INPUT

```
some text some raised
 some lowered
```

OUTPUT

## Rules

- Rule là một hình chữ nhật: `\rule[lift]{width}{height}`

```
Some text
\rule{0.5in}{10pt}
\rule[-3pt]{0.5in}{10pt}
some text.
```

INPUT

Some text   some text.

OUTPUT

## Ví dụ

Ví dụ này dùng `\settowidth`, `\makebox` và `\rule`:

```
\newlength\mylen
\settowidth{\mylen}{Some Text}%
\makebox[0pt][l]{\rule[0.5ex]{\mylen}{1pt}}%
Some Text
```

INPUT

~~Some Text~~

OUTPUT

Hơi tởm, cầu kỳ! Cần định nghĩa các lệnh mới!



## Ví dụ

```
\newlength\mylen
\newcommand{\strikethrough}[1]{%
\settowidth{\mylen}{#1}%
\makebox[0pt][l]{\rule[0.5ex]{\mylen}{1pt}}%
#1}

\strikethrough{Some More Text}
```

INPUT

~~Some More Text~~

OUTPUT

Ví dụ này không cho phép ngắt dòng. Dùng gói `ulem` để làm việc đó.

## Struts

Rule có chiều dài 1 gọi là *strut*:

```
\fbox{text}
\fbox{\rule[-10pt]{0pt}{20pt}text}
text\rule{1in}{0pt}text
```

INPUT



OUTPUT

## Saveboxes

- Cho phép ghi lại đoạn văn bản để dùng sau.
- Khai báo một savebox mới: `\newsavebox{cmd}`.
- Ghi một đoạn text vào savebox:
  - command:  
`\sbox{cmd}{text}`
  - `lrbox` environment:  
`\begin{lrbox}{cmd}`  
*text*  
`\end{lrbox}`
- To display the contents of a savebox: `\usebox{cmd}`

## Ví dụ dùng `\sbox`

```
\newsavebox{\mysbox}
\sbox{\mysbox}{Some interesting text}

\usebox{\mysbox}\
\fbox{\usebox{\mysbox}}\
\Ovalbox{\usebox{\mysbox}}
```

INPUT

Some interesting text

Some interesting text

Some interesting text

OUTPUT

## Ví dụ dùng **lrbox**

```
\newsavebox{\mysbox}
\begin{lrbox}{\mysbox}
Some more interesting text.
\end{lrbox}

\usebox{\mysbox}\
\fbox{\usebox{\mysbox}}
```

INPUT

Some more interesting text.  
Some more interesting text.

OUTPUT

## Macros hay Saveboxes?

Dùng macro (hay *command*):

```
\newcommand{\sometext}{Some text}

\sometext.\\
\sffamily \sometext.\\
\ttfamily \sometext.
```

INPUT

```
Some text.
Some text.
Some text.
```

OUTPUT

T<sub>E</sub>X phải xử lý “Some text” 3 lần.

## Macros hay Saveboxes?

Dùng savebox:

```
\newsavebox{\mysbox}
\sbox{\mysbox}{Some text}

\usebox{\mysbox}.\.
\sffamily \usebox{\mysbox}.\.
\ttfamily \usebox{\mysbox}.
```

INPUT

```
Some text.
Some text.
Some text.
```

OUTPUT

TEX chỉ phải xử lý “Some text” một lần.

## Bài tập ?? (trang ??)

- ④ Tạo văn bản như hình ?? trang ?? của handout.
- ⚠ Tạo văn bản như hình ?? trang ?? của handout.
  - Thay đổi canh lề dọc của `minipage`.
  - Thử các chú thích trong và ngoài môi trường `minipage`.
  - Thử dung `\parbox` thay cho `minipage`.
  - Thử các loại khung xung quanh `minipage`.



## Hình vẽ và ảnh

- Có thể tạo ra các ảnh sử dụng các lệnh  $\text{\LaTeX}$  cho vẽ hình. Gói `pstricks` hoặc `pgf` có thể tạo ra những hình ảnh phức tạp, tuy nhiên hơi khó khăn với một số người.
- Cách đơn giản hơn là dùng các chương trình xử lý ảnh, rồi nhúng vào trong tài liệu  $\text{\LaTeX}$  bằng lệnh:

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

trong gói `graphicx` .

- Các loại ảnh phụ thuộc vào loại đầu ra. thường là PS và EPS (với `dvips`) và PDF hay PNG (với `PDF $\text{\LaTeX}$` ).

## Cú pháp

`\includegraphics[options]{filename}`

Các thông số thường dùng:

`angle=x` quay ảnh một góc  $x^\circ$

`width=len` phóng/thu về chiều rộng *len*.

`height=len` phóng/thu về chiều cao *len*.

`scale=x` phóng/thu.

`trim=lx by rx ty` cắt ảnh để góc dưới bên trái là (*lx*, *by*) và góc trên bên phải là (*rx*, *ty*).

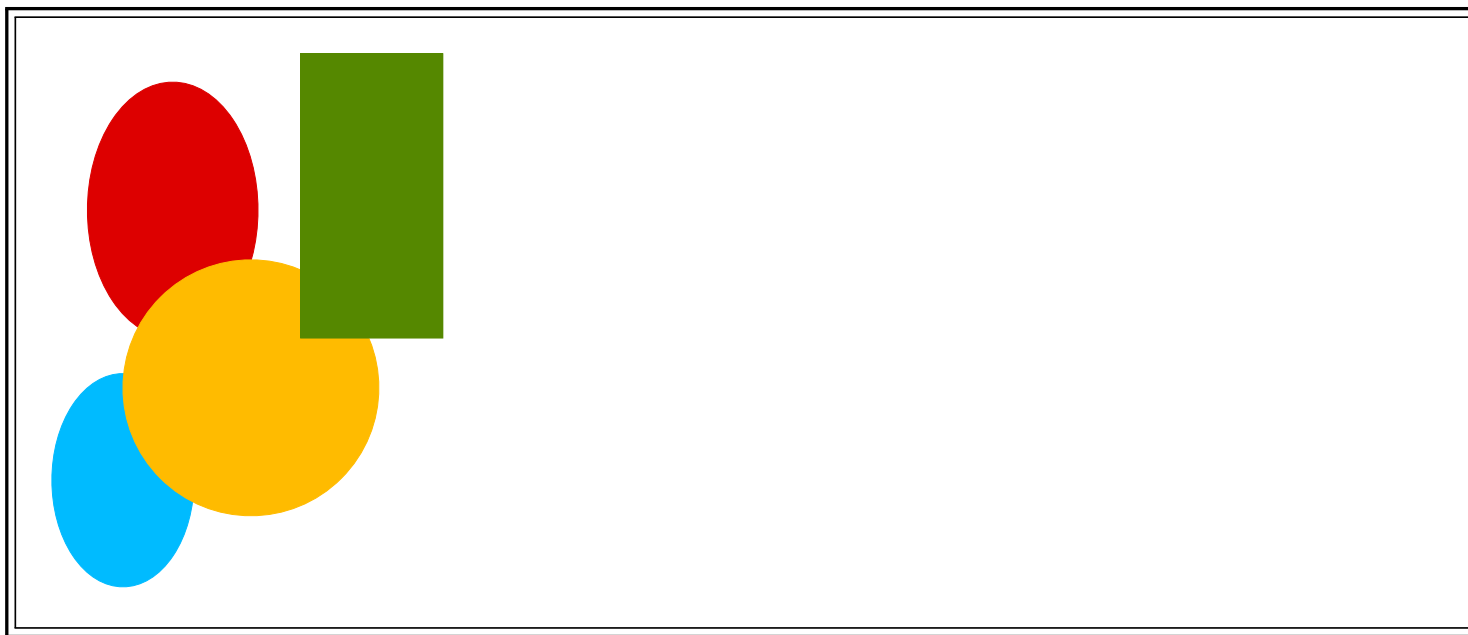
`draft` không hiển thị ảnh, chỉ hiển thị khung xung quanh.

## Ví dụ

- Không cần chỉ ra phần mở rộng của ảnh.

```
\includegraphics[angle=90,width=1in]{shapes}
```

INPUT



OUTPUT

- Nếu là  $\text{\LaTeX}$  dùng `shapes.ps`, nếu  $\text{PDF}\text{\LaTeX}$  dùng `shapes.pdf`.

## Các lệnh khác của gói **graphicx**

- `\rotatebox{angle}{text}`  
Xoay *text* một góc *angle*.
- `\scalebox{h scale}[v scale]{text}`  
Phóng/thu *text*.
- `\reflectbox{text}`  
Soi gương *text*
- `\resizebox{h length}{v length}{text}`  
Thay đổi kích thước *text* rộng *h length* cao *v length*. Ký hiệu ! có thể dùng để giữ tỷ lệ nếu chỉ có một kích thước được đưa ra.

## Ví dụ

```
\rotatebox{45}{Some text}
```

INPUT

Some text

OUTPUT

```
\scalebox{1.5}{Some text}
```

INPUT

Some text

OUTPUT

```
\reflectbox{Some text}
```

INPUT

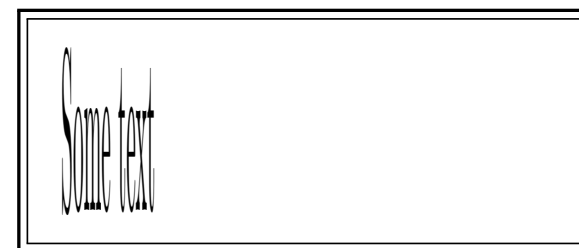
txət əmo2

OUTPUT

## Ví dụ

```
\resizebox{6mm}{1cm}{Some
text}
```

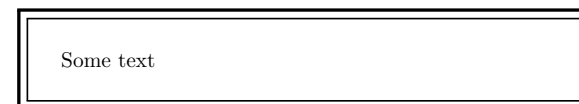
INPUT



OUTPUT

```
\resizebox{6mm}{!}{Some
text}
```

INPUT



OUTPUT

## Bài tập 16 (trang 18)

- tải về `shapes.ps` (hoặc `shapes.pdf` nếu muốn dùng PDF $\LaTeX$ ) to your directory rồi nhúng vào tài liệu.
- Canh giữa ảnh dùng `\centerline`.
- Đóng khung ảnh.
- Phóng to, thu nhỏ và quay ảnh.
- Sử dụng tùy biến `draft` của gói `graphicx` và xem kết quả.

## Hình vẽ và bảng

- Hình vẽ và bảng là các *floats* — được treo-*floated* vào đối tượng gần nhất.
- Hình vẽ và bảng có tiêu đề và số. Khai báo tiêu đề:  
`\caption[short caption]{caption text}`
- $\text{\LaTeX}$  quản lý các số thứ tự tự động. Các đối tượng treo có thể được tham chiếu bằng `\label` and `\ref`.
- Hình vẽ được khai báo bằng môi trường `figure`.
- Bảng được khai báo bằng môi trường `table`.
- `figure` và `table` không được có ngắt trang.



## Hình vẽ ví dụ

```
\begin{figure}[tbh]
\centerline{\includegraphics[height=1.25cm]{shapes.ps}}
\caption{Some shapes}
\label{fig:shapes}
\end{figure}
```

INPUT



Hình 1: Some shapes

OUTPUT

## Ví dụ về bảng

```

\begin{table}[tbh]
\caption{An example table}
\label{tab:example}
\vspace{10pt}
\centerline{
\begin{tabular}{l|ll}
& A & B\\\hline
I & 0.5 & 1.0\\
II & 12 & 14
\end{tabular}
}
\end{table}

```

INPUT

Bảng 1: An example table

|    | A   | B   |
|----|-----|-----|
| I  | 0.5 | 1.0 |
| II | 12  | 14  |

OUTPUT

## Các hình bổ sung

Hai hình vẽ có thể được đặt trong một **figure** :

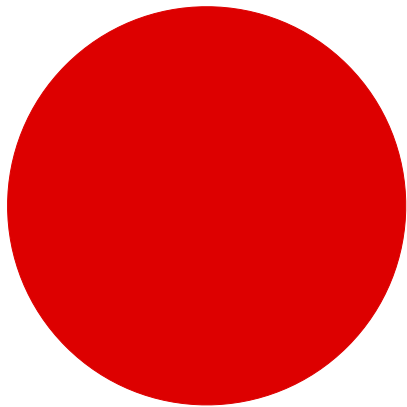
```
\begin{figure}[tbh]
\begin{minipage}{0.4\textwidth}
\centerline{\includegraphics{circle.ps}}
\caption{A Circle}\label{fig:circ}
\end{minipage}
\begin{minipage}{0.5\textwidth}
\centerline{\includegraphics{rectangle.ps}}
\caption{A Rectangle}\label{fig:rect}
\end{minipage}
\end{figure}
```

Figure~\ref{fig:circ} shows a circle.

Figure~\ref{fig:rect} shows a rectangle.

INPUT

## Các hình bổ sung



Hình 2: A Circle



Hình 3: A Rectangle

Hình 2 vẽ đường tronf. Hình 3 vẽ hình chữ nhật.

OUTPUT

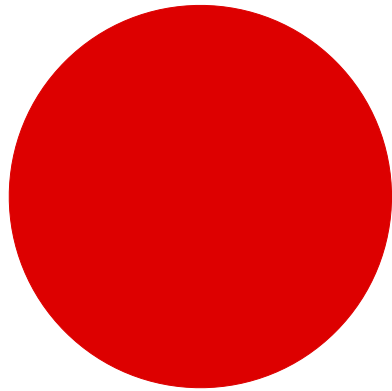
## Các hình con

Subfigures - hình con tạo bởi lệnh `\subfigure[caption]{contents}` which is trong gói `subfigure`.

```
\begin{figure}[tbh]
\centering \subfigure[A Circle]{\label{fig:circle}%
\includegraphics[height=1in,clip]{circle.ps}}
\hspace{0.5in}
\subfigure[A Rectangle]{\label{fig:rectangle}%
\includegraphics[height=1in,clip]{rectangle.ps}}

\caption{(a) A Circle, (b) A Rectangle}
\label{fig:subfigex}
\end{figure}
Figure~\ref{fig:circle} shows a circle,
Figure~\ref{fig:rectangle} shows a rectangle.
```

INPUT



(a) A Circle



(b) A Rectangle

Hình 4: (a) A Circle, (b) A Rectangle

Figure 4(a) shows a circle, Figure 4(b) shows a rectangle.

OUTPUT

## Danh sách hình vẽ, bảng

- Danh sách các hình vẽ có thể được tạo ra bằng:  
`\listoffigures`
- Danh sách các bảng có thể được tạo ra bằng::  
`\listoftables`
- Thông thường được đặt sau **mục lục**.
- Cần  $\text{\LaTeX}$  văn bản hai lần để cập nhật danh sách.

## Bài tập 17 (trang 19)

- Copy `circle.ps`, `rectangle.ps` và `shapes.ps` vào thư mục hiện tại (hoặc `circle.pdf`, `rectangle.pdf` và `shapes.pdf`).
- tạo tài liệu chứa hình 16 và 17 bảng 11 trong handout.
- Thêm một danh sách các bảng và các hình vẽ vào tài liệu.



## Định nghĩa môi trường mới

Môi trường mới có thể được định nghĩa bằng

`\newenvironment{env-name}[n][default]{begin-code}{end-code}`

```
\newenvironment{bfitemize}%
{\begin{bfseries}\begin{itemize}}%
{\end{itemize}\end{bfseries}}
```

```
\begin{bfitemize}
\item First item
\item Second item
\end{bfitemize}
```

INPUT

- First item
- Second item

OUTPUT

## Ví dụ một môi trường có tham số

```
\newsavebox{\fminibox}

\newenvironment{fminipage}[2][c]%
{\begin{lrbox}{\fminibox}\begin{minipage}[#1]{#2}}%
{\end{minipage}\end{lrbox}}%
\shadowbox{\usebox{\fminibox}}
```

INPUT

```
\begin{fminipage}{1.5in}
Some text in a 1.5
inch framed minipage
\end{fminipage}
```

INPUT

Some text in a 1.5 inch  
framed minipage

OUTPUT

## Bài tập ?? (trang ??)

- Tạo ra một môi trường mới `exercise` kẻ một đường thong ngang ở đầu và ở cuối của môi trường. Ví dụ đoạn mã sau:

```
\begin{exercise}
```

```
Some text.
```

```
\end{exercise}
```

sẽ hiển thị thành:

---

Some text.

---

- Nếu môi trường có một ngắt đoạn ở trước, sẽ có một khoảng cách nhỏ trước đường ngang đầu tiên do ngắt đoạn này. Có thể bỏ qua bằng lệnh: `\noindent`.

## Con đếm

- Là các biến nguyên có giá trị tăng dần.
- Các lệnh: `\chapter`, `\section`, `\footnote`, `\caption` đều có các con đếm tương ứng.
- Các môi trường `equation`, `enumerate`. đều có các con đếm tương ứng.

- Định nghĩa một con đếm mới:

`\newcounter{ctr-name} [outer-ctr]`

Ví dụ: `\newcounter{exercise}`.

- Khởi tạo lại một con đếm khi con đếm khác tăng 1 đơn vị:  
`\newcounter{exercise}[chapter]`

## Thay đổi giá trị của con đếm

- `\stepcounter{ctr}` tăng giá trị của con đếm 1 đơn vị.
- `\refstepcounter{ctr}` giống như trên, nhưng cho phép tham chiếu tới con đếm dùng `\ref` và `\label`.
- `\setcounter{ctr}{value}` gán giá trị *value* cho con đếm.
- `\addtocounter{ctr}{value}` thêm *value* vào con đếm.
- `\value{ctr}` Tạo ra các giá trị để dùng trong lệnh `\setcounter` và `\addtocounter`.

## Hiển thị giá trị của con đếm

- Lệnh `\the $\textcolor{blue}{ctr}$`  hiển thị giá trị của con đếm  $\textcolor{blue}{ctr}$ . Ví dụ:
  - `\thepage` hiển thị giá trị của con đếm  $\textcolor{teal}{page}$ .
  - `\thesection` hiển thị giá trị của con đếm  $\textcolor{teal}{section}$  counter.
  - `\theslide` hiển thị giá trị của con đếm  $\textcolor{teal}{slide}$  counter.

```
This is slide number
\theslide
```

INPUT

```
This is slide number
197
```

OUTPUT

- Mặc định lệnh `\the $\textcolor{blue}{ctr}$`  hiển thị bằng số ả rập.
- `\the $\textcolor{blue}{ctr}$`  có thể định nghĩa lại bằng `\renewcommand` để hiển thị theo dạng khác.

## Dạng số đếm chuẩn

- `\arabic{ctr}` hiển thị *ctr* bằng số Ả Rập
- `\roman{ctr}` hiển thị *ctr* bằng số La mã thường
- `\Roman{ctr}` hiển thị *ctr* bằng số La mã hoa
- `\alph{ctr}` hiển thị *ctr* bằng các chữ cái (giá trị con đếm cần nhỏ hơn 26)
- `\Alph{ctr}` hiển thị *ctr* bằng các chữ cái hoa (giá trị con đếm cần nhỏ hơn 26)
- `\fnsymbol{ctr}` hiển thị *ctr* giống các chú thích nhỏ. (chỉ dùng trong chế độ công thư)

Chỉ nên dùng các lệnh này khi thay đổi lệnh `\thectr`

## Ví dụ

- `\thechapter` hiển thị giá trị con đếm `chapter` .
- `\renewcommand{\thechapter}{\Roman{chapter}}`  
định nghĩa lại `\thechapter` để hiển thị số trang bằng chữ số La mã hoa.
- `\renewcommand{\thefootnote}{\alph{footnote}}`  
hiển thị số thứ tự chú thích bằng chữ cái thường.
- `\newcounter{lemma}[section]`  
định nghĩa số đếm mới cho các bổ đề, khởi động lại mỗi section.
- `\renewcommand{\thelemma}{\thesection.\arabic{lemma}}`  
Hiển thị bổ đề theo dạng 4.3



## Định dạng con đếm bổ sung(gói **datetime**)

Gói **datetime** cung cấp các lệnh sau đây để hiển thị con đếm:

|                                             |                                                                       |
|---------------------------------------------|-----------------------------------------------------------------------|
| <code>\ordinal{<i>counter</i>}</code>       | Display the value of <i>counter</i> as an ordinal                     |
| <code>\ordinalstring{<i>counter</i>}</code> | Display the value of <i>counter</i> as an ordinal written out in full |
| <code>\Ordinalstring{<i>counter</i>}</code> | As above, but with the initial letters in uppercase                   |
| <code>\numberstring{<i>counter</i>}</code>  | Display the value of <i>counter</i> as a string                       |
| <code>\Numberstring{<i>counter</i>}</code>  | As above but with the initial letter in uppercase                     |

## Ví dụ

|                                    |                       |
|------------------------------------|-----------------------|
| <code>\ordinal{slide}</code>       | 201 <sup>st</sup>     |
| INPUT                              | OUTPUT                |
| <code>\ordinalstring{slide}</code> | two hundred and first |
| INPUT                              | OUTPUT                |
| <code>\Ordinalstring{slide}</code> | Two Hundred and First |
| INPUT                              | OUTPUT                |
| <code>\numberstring{slide}</code>  | two hundred and one   |
| INPUT                              | OUTPUT                |
| <code>\Numberstring{slide}</code>  | Two Hundred and One   |
| INPUT                              | OUTPUT                |

## Định dạng con đếm mở rộng (Gói **datetime** v2.4)

Version 2.4 của **datetime** bổ sung thêm các lệnh sau đây:

|                                           |                                                                                      |
|-------------------------------------------|--------------------------------------------------------------------------------------|
| <code>\binary{<i>counter</i>}</code>      | hiển thị giá trị của <i>counter</i> bằng số nhị phân                                 |
| <code>\octal{<i>counter</i>}</code>       | hiển thị giá trị của <i>counter</i> bằng hệ octan                                    |
| <code>\hexadecimal{<i>counter</i>}</code> | hiển thị giá trị của <i>counter</i> bằng số thập lục phân                            |
| <code>\aaalph{<i>counter</i>}</code>      | hiển thị giá trị của <i>counter</i> bằng dạng: a ... z aa ...zz aaa ...              |
| <code>\abalph{<i>counter</i>}</code>      | hiển thị giá trị của con đếm <i>counter</i> dưới dạng: a ... z aa ab ...az ba bb ... |

## Ví dụ

`\binary{slide}`

INPUT

11001011

OUTPUT

`\octal{slide}`

INPUT

313

OUTPUT

`\hexadecimal{slide}`

INPUT

cb

OUTPUT

`\aaalph{slide}`

INPUT

uuuuuuuu

OUTPUT

`\abalph{slide}`

INPUT

gu

OUTPUT

## Con đếm của các danh sách

4 tầng `enumerate` có thể lồng nhau. Mỗi tầng có một con đếm riêng biệt:

| Counter              | Representation             | Default                      | Example |
|----------------------|----------------------------|------------------------------|---------|
| <code>enumi</code>   | <code>\theenumi</code>     | <code>\arabic{enumi}</code>  | 1       |
|                      | <code>\labelenumi</code>   | <code>\theenumi.</code>      | 1.      |
| <code>enumii</code>  | <code>\theenumii</code>    | <code>\alph{enumii}</code>   | a       |
|                      | <code>\labelenumii</code>  | <code>\theenumii)</code>     | a)      |
| <code>enumiii</code> | <code>\theenumiii</code>   | <code>\roman{enumiii}</code> | i       |
|                      | <code>\labelenumiii</code> | <code>\theenumiii.</code>    | i.      |
| <code>enumiv</code>  | <code>\theenumiv</code>    | <code>\Alph{enumiv}</code>   | A       |
|                      | <code>\labelenumiv</code>  | <code>\theenumiv.</code>     | A.      |

## Thay đổi định dạng con đếm **enumerate**

```
\renewcommand{\theenumi}{\Roman{enumi}}
```

INPUT

```
\begin{enumerate}
\item\label{itm:first} First item
\item Second item
\end{enumerate}
Item~\ref{itm:first} \ldots
```

INPUT

```
I. First item
II. Second item

Item I ...
```

OUTPUT

## Thay đổi nhãn của **enumerate**

```
\renewcommand{\labelenumi}{\#\theenumi}
```

INPUT

```
\begin{enumerate}
\item\label{itm:first} First item
\item Second item
\end{enumerate}
Item~\ref{itm:first} \ldots
```

INPUT

```
#1 First item
#2 Second item

Item 1 ...
```

OUTPUT

## Thay đổi định dạng của **enumerate**

```
\renewcommand{\theenumi}{\Numberstring{enumi}}
\renewcommand{\labelenumi}{\theenumi:}
```

INPUT

```
\begin{enumerate}
\item\label{itm:first} First item
\item Second item
\end{enumerate}
Item~\ref{itm:first} \ldots
```

INPUT

```
One: First item
Two: Second item

Item One ...
```

OUTPUT



## Bài tập ?? (trang ??)

- Thay đổi môi trường tạo ra trong bài tập ?? để đoạn mã sau:

```
\begin{exercise}
```

Some text.

```
\end{exercise}
```

hiển thị thành:

---

### Exercise 1

Some text.

---

- Giá trị của con đếm khởi tạo lại mỗi bài tập.
- Tham chiếu sử dụng `\label` và `\ref`.

## Các lệnh điều kiện

- Giống các ngôn ngữ lập trình khác, T<sub>E</sub>X có các cấu trúc điều kiện.
- Câu lệnh điều kiện của T<sub>E</sub>X có dạng:

```
\iftype ... \else ... \fi
```

- Đã sử dụng (`ifpdf` package):

```
\ifpdf ... \else ... \fi
```

- Lệnh `\ifcase` giống lệnh `switch` trong C. Ví dụ:

```
\ifcase\month \or Jan\or Feb\or Mar\or Apr\or May\or
Jun\or Jul\or Aug\or Sep\or Oct\or Nov\or Dec\fi
```

## Ví dụ về (`\ifpdf` và `\pdfinfo`)

- `\ifpdf` thường dùng đi đôi với `\pdfinfo`.
- Lệnh `\pdfinfo` định nghĩa trong PDF $\LaTeX$  không trong  $\LaTeX$ .

```
\ifpdf
 \pdfinfo{
 /Title (A Sample Document)
 /Author (Nicola Talbot)
 /CreationDate (D:20040930140000)
 /ModDate (D:\pdfdate)
 /Subject (LaTeX Example)
 }
\fi
```

## Gói **ifthen**

- Gói **ifthen** (by Leslie Lamport, extended by David Carlisle) định nghĩa các lệnh điều kiện:
  - `\ifthenelse{test}{then text}{else text}`
  - `\whiledo{test}{do text}`
- Tham số *test* là biểu thức logic.
- Nếu *test* là true, *then text* hay *do text* sẽ được thực hiện.
- Nếu *test* là false, *else text* được thực hiện, hoặc khi dùng `\whiledo`, vòng lặp kết thúc.
- Đơn giản hơn các lệnh của T<sub>E</sub>X .

## Các lệnh logic

- Các biến được định nghĩa bằng:

```
\newboolean{name}
```

trong đó *name* (no backslash) là tên biến mới.

- Gán giá trị bằng lệnh

```
\setboolean{name}{value}
```

trong đó *name* tên biến, còn *value* là **true** hoặc **false**.

- Giá trị của biến Boolean có thể được sử dụng:

```
\boolean{name}
```

## Ví dụ

- Dạy cùng môn học có 2 lớp. Slide, note không thay đổi, chỉ có thời gian thay đổi.
- Giả sử một lớp học thứ sáu, 9:00AM và và thứ năm 13:00am.

- Định nghĩa biến logic

```
\newboolean{friday}
```

- Định nghĩa giá trị của biến. e.g.:

```
\setboolean{friday}{true}
```

- Can now use `\ifthenelse` and `\boolean`:

Classes are on

```
\ifthenelse{\boolean{friday}}{Fridays}{Mondays} at
```

```
\ifthenelse{\boolean{friday}}{9:00am}{10:00am}.
```

## Ví dụ

- Tất cả các lệnh có điều kiện của T<sub>E</sub>X như `\ifpdf` có giá trị tương ứng, such as `pdf`.
- `\ifthenelse` đơn giản hơn câu lệnh bậc thấp `\ifpdf` etc.
- Đoạn **earlier example** có thể viết thành:

```
\ifthenelse{\boolean{pdf}}{%
A PDF\LaTeX\ document
}{%
A \LaTeX\ document
}
```

INPUT

A PDF<sub>L</sub>A<sub>T</sub>E<sub>X</sub> docu-  
ment

OUTPUT

## Kiểm tra văn bản Text

- so sánh hai câu, dùng:  
`\equal{string 1}{string 2}`

- Ví dụ:

The work is written in

```
\ifthenelse{\equal{\lang}{English}}
{English} {another language}
```

Nếu `\lang` có giá trị English, đầu ra:

The work is written in English.

OUTPUT

If `\lang` has been defined as something else:

The work is written in another language.

OUTPUT



## Các số kiểm tra

- Các số có thể được cấp từ <, = and >
- Giá trị các counter bằng: `\value{name}`
- Ví dụ:

This

```
\ifthenelse{\value{page}=42}{is}{isn't}
my favourite page.
```

- To test whether a number is odd or even use `\isodd{value}`
- Example:

This page is an

```
\ifthenelse{\isodd{\value{page}}}{odd}{even}
numbered page.
```

## `\whiledo` bài tập

```
\newcounter{lines}
\whiledo{\value{lines}<6}
 {I will hand my homework in on time.\par
 \stepcounter{lines}}
```

INPUT

I will hand my homework in on time.  
I will hand my homework in on time.  
I will hand my homework in on time.  
I will hand my homework in on time.  
I will hand my homework in on time.  
I will hand my homework in on time.

OUTPUT

## Kiểm tra chiều dài

- Các chiều dài do các câu lệnh sau cung cấp  
`\lengthtest{relation}`
- Ví dụ:

```
This page is in
\ifthenelse{\lengthtest{\textwidth > \textheight}}
 {landscape} {portrait}
```

INPUT

```
This page is in landscape
```

INPUT

## Viết gói L<sup>A</sup>T<sub>E</sub>X

- Tên file phải tuân theo chuẩn của HĐH với phần mở rộng `.sty` .
- Các gói đều bắt đầu bằng dòng sau  
`\NeedsTeXFormat{LaTeX2e}`

- Chỉ ra tên của gói bằng  
`\ProvidesPackage{name} [version]`

Ví dụ nếu file tên `trunghq`.

```
\ProvidesPackage{trunghq}
```

Có thể chỉ ra version trong các tham số thêm:

```
\ProvidesPackage{example}[2004/05/21 v1.0 (A.N. Other)]
```

- Dòng cuối cùng của file cần là `\endinput`

## Ví dụ

Đây là một gói rất đơn giản. Định nghĩa lại lệnh `\today` để hiển thị ngày dưới dạng 21-5-2006.

```
\NeedsTeXFormat{LaTeX2e}

\ProvidesPackage{vardate}[2006/05/21 v1.0 (H.Q Trung)]

\renewcommand{\today}{\the\day-\the\month-\the\year}

\endinput
```

VARDATE.STY

## Tham số của gói

- Một số gói (v.d. `graphicx`) có các tham số được truyền vào gói
- Các tham số mới có thể được định nghĩa bằng

```
\DeclareOption{option}{code}
```

trong đó *option* là giới hạn của cáo *code*. Vậy  $\text{\LaTeX}$  có thể thực hiện khi có tham số này .

- Thao tác mặc định với các tham số chưa định nghĩa được khai báo bằng

```
\DeclareOption*{code}
```

Trong *code*, các lệnh sau có thể được sử dụng:

`\CurrentOption`    tên của tham số

`\OptionNotUsed`    Đánh dấu tham số chưa xử lý.

## Tham số của gói

- Các lệnh được xử lý bằng

|                                              |                                |
|----------------------------------------------|--------------------------------|
| <code>\ExecuteOptions{<i>options</i>}</code> | danh sách các tham số          |
| <code>\ProcessOptions</code>                 | xử lý theo thứ tự định nghĩa   |
| <code>\ProcessOptions*</code>                | xử lý theo thứ tự gán giá trị. |

- Có thể pass các tham số cho các gói khác:

`\PassOptionsToPackage{option list}{package name}`

- Có thể tải các gói khác dùng

`\RequirePackage{package name}`

## Mở rộng gói **datetime** — vardate.sty

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{vardate}

\RequirePackage{ifthen}

\newboolean{dashdate}

\DeclareOption{dashdate}{\setboolean{dashdate}{true}}
\DeclareOption{nodashdate}{\setboolean{dashdate}{false}}
\DeclareOption*{\PassOptionsToPackage{\CurrentOption}{datetime}}

\ExecuteOptions{dashdate}
\ProcessOptions

\RequirePackage{datetime}

\newdateformat{dashdate}{\THEDAY-\THEMONTH-\THEYEAR}
\ifthenelse{\boolean{dashdate}}{\dashdate}{}
\endinput
```



## Viết các lớp tài liệu

Rất giống viết các gói, nhưng:

- Dùng `\ProvidesClass` thay cho `\ProvidesPackage`
- Dùng lệnh

```
\PassOptionsToClass{options}{class name}
```

để truyền tham số *options* vào lớp mà sau đó được tải bằng

```
\LoadClass{class name}
```

## Mở rộng lớp **report** — myrep.cls

```
\NeedsTeXFormat{LaTeX2e}
```

```
\ProvidesClass{myrep}
```

```
\DeclareOption*{\PassOptionsToClass{\CurrentOption}{report}}
```

```
\ProcessOptions
```

```
\LoadClass{report}
```

```
\RequirePackage[margins=1in]{geometry}
```

```
\renewcommand{\thechapter}{\Roman{chapter}}
```

```
\endinput
```

## @ Commands

- Ký hiệu @ có ý nghĩa khác nhau trong gói và trong lớp (.cls/.sty), hay trong tài liệu (.tex).
- Trong tài liệu .tex file, @ là ký hiệu, không được dùng cho tên lệnh.  
V.d. \c@page sẽ tạo ra: @page.
- Trong lớp .cls or .sty , @ là ký hiệu có thể có trong tên lệnh.  
V.d. \c@page là lệnh (biểu diễn nội bộ của con đếmpage.)
- Các lệnh có @ là các lệnh nội bộ, chỉ dùng trong các tệp gói và lớp.

## Định nghĩa lại kiểu trang

- Lệnh `\pagestyle{style}` gọi lệnh `\ps@style` là một lệnh định nghĩa lại phần đầu và cuối của một trang.
- Khai báo phần đầu và phần cuối: `\@oddhead`, `\@evenhead`, `\@oddfoot` và `\@evenfoot`. Đây là các lệnh cần định nghĩa lại.

- Ví dụ: định nghĩa kiểu trang mới `example`:

```
\newcommand{\ps@example}{%
 \renewcommand{\@oddhead}{}
 \renewcommand{\@evenhead}{}
 \renewcommand{\@oddfoot}{\hfill\thepage}
 \renewcommand{\@evenfoot}{\thepage\hfill}
}
```

## Thay đổi tiêu đề của các section

Tiêu đề của Sections, subsections có thể được thay đổi bằng cách định nghĩa lại `\section`, `\subsection` .... Trong các định nghĩa cần sử dụng

`\@startsection{type}{level}{indent}{before}{after}{style}`

để định dạng tiêu đề.

*type* Kiểu section. (section, subsection etc)

*level* Số biểu diễn mức section

*indent* Indentation từ lề trái

*before* Cách dòng trước tiêu đề.

*after* Cách dòng sau tiêu đề.

*style* Khai báo kiểu được dùng (e.g. `\bfseries`)

## Ví dụ

Cần thay đổi tiêu đề của section hiển thị bằng font nghiêng to:

```
\renewcommand{\section}{\@startsection
{section}% % the name
{1}% % the level
{0mm}% % the indent
{-\baselineskip}% % the before skip
{0.5\baselineskip}% % the after skip
{\normalfont\large\itshape}} % the style
```

## Thay đổi tiêu đề chương

Nếu dùng kiểu `report` tiêu đề chương có thể được thay đổi bằng cách định nghĩa lại:

- `\@makechapterhead` cho các chương đánh số (`\chapter`)
- `\@makeschapterhead` cho các chương không đánh số (`\chapter*`)

Thay đổi tiêu đề của phần:

- `\@part` cho các phần đánh số (`\part`)
- `\@spart` cho các phần không đánh số (`\part*`)

Cách dễ nhất là lấy code từ file lớp cơ bản, sau đó thay đổi theo nhu cầu.

## Ví dụ

Thay đổi tiêu đề chương để có một dòng kẻ trước và sau tiêu đề, tiêu đề hiển thị bằng chữ thường.

```
\renewcommand{\@makechapterhead}[1]{%
 \vspace*{50\p@}%
 {\parindent \z@ \raggedright \normalfont
 \hrule % horizontal line
 \vspace{5pt}% % add some vertical space
 \ifnum \c@secnumdepth >\m@ne
 \huge\scshape \@chapapp\space \thechapter % Chapter followed by number
 \par\nobreak
 \vskip 20\p@
 \fi
 \interlinepenalty\@M
 \Huge \scshape #1\par % chapter title
 \vspace{5pt}% % add some vertical space
 \hrule % horizontal rule
 \nobreak
 \vskip 40\p@
 }
```



## Recommended Reading

- [1] “A Guide to  $\text{\LaTeX} 2_{\epsilon}$ : document preparation for beginners and advanced users”, Helmut Kopka and Patrick W. Daly, Addison-Wesley (1995).
- [2] “The  $\text{\LaTeX}$  Companion”, Michel Goossens, Frank Mittelbach and Alexander Samarin (Addison-Wesley, 1994).
- [3] “The  $\text{\LaTeX}$  Graphics Companion”, Michel Goossens, Sebastian Rahtz and Frank Mittelbach, Addison-Wesley (1997).
- [4] “The  $\text{\LaTeX}$  Web Companion”, Michel Goossens and Sebastian Rahtz, Addison-Wesley (1999).

# Index