

Chương 7. Các lệnh nâng cao



TS. Cao Xuân Phương

Khoa Toán–Thống kê, Trường đại học Tôn Đức Thắng

Ngày 31 tháng 7 năm 2020

7.1. Chia nhỏ tập tin nguồn

- Khi làm việc với một tài liệu lớn, ta cần tách tập tin nguồn thành nhiều phần. LaTeX cung cấp hai lệnh để thực hiện điều đó.

- Lệnh

`\input{filename}`

được dùng trong phần thân của tài liệu để đưa vào nội dung của tập tin `filename`. LaTeX sẽ gọi tập tin `filename` và xếp nội dung của nó chèn thêm vào đúng vị trí của lệnh trong tập tin nguồn.

- Lệnh

`\include{filename}`

được dùng trong phần thân của tài liệu để đưa vào nội dung của tập tin `filename`. LaTeX sẽ bắt đầu trang mới trước khi biên dịch nội dung của tập tin `filename`.

7.1. Chia nhỏ tập tin nguồn

- Nếu trong phần đầu của tài liệu, ta khai báo lệnh

```
\includeonly{danh sách các tập tin}
```

thì chỉ có các tập tin trong danh sách đó mới được thực hiện bởi lệnh `\include`. Chú ý rằng các tập tin trong danh sách được cách nhau bởi các dấu phẩy.

7.2.1. Tạo lệnh mới

- Để tạo một lệnh mới, ta dùng lệnh

```
\newcommand{name}[num]{definition}
```

Về cơ bản, lệnh này yêu cầu phải có hai đối số: `name` là tên của lệnh cần tạo và `definition` là định nghĩa của lệnh. Đối số `num` trong dấu móc vuông là đối số tùy chọn, dùng để chỉ số đối số theo sau lệnh `name`. Ta có thể tạo một lệnh mới với tối đa 9 đối số.

7.2. Các lệnh, môi trường và gói lệnh mới

Ví dụ:

```
\newcommand{\tdtu}{Ton Duc Thang University}
```

This is \tdtu.

This is Ton Duc Thang University.

7.2. Các lệnh, môi trường và gói lệnh mới

Ví dụ:

```
\newcommand{\R}{\mathbb{R}}
```

Let f be a function defined on \R .

Let f be a function defined on \mathbb{R} .

7.2. Các lệnh, môi trường và gói lệnh mới

Ví dụ:

```
\newcommand{\QED}{\Box}
```

```
\ldots This completes the proof. \QED
```

... This completes the proof. \square

Ví dụ:

```
\newcommand{\thisis}[1]{This is the \emph{#1} example  
in the section.}  
  
\begin{itemize}  
  \item \thisis{first}  
  \item \thisis{second}  
\end{itemize}
```

Thẻ #1 xác định nơi cần thay bởi đối số thứ nhất.

- This is the *first* example in the section.
- This is the *second* example in the section.

7.2. Các lệnh, môi trường và gói lệnh mới

- LaTeX không cho phép tạo một lệnh mới trùng tên với một lệnh đã có. Nhưng có một lệnh đặc biệt có thể dùng để bỏ lệnh cũ là `\renewcommand`. Cấu trúc của lệnh này là tương tự như lệnh `\newcommand`.

7.2.2. Tạo lệnh môi trường mới

- Ta có thể tạo một môi trường mới bằng lệnh

```
\newenviroment{name}[num]{before}{after}
```

Như lệnh `\newcommand`, lệnh `\newenviroment` có thể có hoặc không có đối số tùy chọn `num`. Đối số `before` là phần được thực hiện trước phần văn bản được đặt trong môi trường. Đối số `after` là phần được thực hiện khi kết thúc tiến trình thực hiện bởi lệnh `\end{name}`.

7.2. Các lệnh, môi trường và gói lệnh mới

Ví dụ:

```
\newenvironment{CM}{\textbf{Chúng minh.}}{${\Box$}}
```

Chúng minh. Ta có ... \square

7.2. Các lệnh, môi trường và gói lệnh mới

- LaTeX cũng không cho phép tạo một môi trường mới trùng tên với một môi trường đã có. Nhưng có một lệnh đặc biệt có thể dùng để bỏ môi trường cũ là `\renewenvironment`. Cấu trúc của lệnh này là tương tự như lệnh `\newenvironment`.

7.2.3. Tạo gói lệnh

- Nếu ta định nghĩa nhiều môi trường và lệnh mới, phần mở đầu của tài liệu sẽ trông khá dài. Để khắc phục điều này, ta có thể tạo một gói lệnh LaTeX chứa tất cả các lệnh và môi trường mà mình định nghĩa, sau đó dùng lệnh `\usepackage` để kích hoạt gói lệnh vào trong tài liệu.
- Một gói lệnh phải được viết ra thành một tập tin riêng với phần mở rộng là `.sty`.
- Lệnh `\ProvidesPackage` được đặt ở phần đầu của tập tin chứa gói lệnh. `\ProvidesPackage` cung cấp cho LaTeX tên của gói lệnh.

7.2. Các lệnh, môi trường và gói lệnh mới

Ví dụ:

```
\ProvidesPackage{CXP}  
  
\newcommand{\QED}{$\Box$}  
  
\newcommand{\R}{$\mathbb{R}$}  
  
\newenvironment{Eq}{\begin{equation}}{\end{equation}}
```