

# 便利に使う Vim 入門

---

北海道大学理学院 宇宙理学専攻 M1 人見祥磨

令和 2 年 7 月 31 日

# Vim とは何か

---

# Vim とは

- Vi improved
- テキストエディタ vi を拡張したもの
- 現在では、vi で vim (の機能を制限したもの) が起動することが多い

```
% vi --version
```

```
VIM - Vi IMproved 8.1 以下略
```

## テキストエディタとは

テキストファイルを編集するためのソフトウェア

**スクリーンエディタ** テキストファイルの中身を画面に表示しながら、カーソルを移動させてテキストを編集するもの

**ラインエディタ** テキストファイルの中身を表示せず、行単位でコマンドを利用しながら編集するもの

vi は ラインエディタ ex の流れを汲む

# Vim の使い方

## Vim を使いこなすのに参考になるもの

**vimtutor** Vim を一切知らない人が、一通りの機能を扱えるようになるための公式チュートリアル

ヘルプファイル 正確で詳しい情報が載っている

## ヘルプファイルの表示方法

Vim ノーマルモードで、**:help <調べたいもの>** と打ち込むことで表示できる

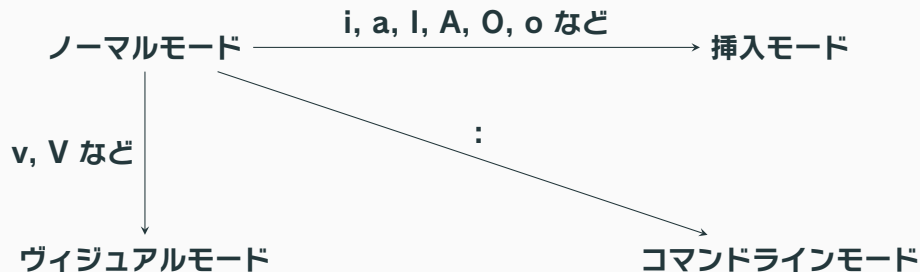
日本語のヘルプファイルを導入するにはひと手間必要

<https://vim-jp.org/vimdoc-ja/index.html> からも日本語ドキュメントを参照できる

# Vim のモード

Vim は複数のモードを持つ

複数のモードを行き来してテキストの編集を行う



Esc キーでノーマルモードに戻る

## それぞれのモードの役割

ノーマルモード	カーソル移動、ヤンク、貼り付け、削除などをするモード
挿入モード	文字入力などをするモード
ビジュアルモード	範囲を選択するモード
コマンドラインモード	コマンドを実行するモード

### Ex モード

Ex モードというモードもある（ノーマルモードで Q で移行）

コマンドを複数続けて実行するモード

<Esc> ではノーマルモードに戻れない

:visual でノーマルモードに戻る

## Vim の設定

---

# Vim の設定ファイル

Vim の設定はコマンドラインモードで行うことができる

例: `:syntax on` でカラーリングをオンにできる

よく使う設定を毎回設定するのは面倒

~/.vimrc ファイルにコマンドを書き込んでおくと、Vim 起動時に読み込んでくれる

## defaults.vim

vimrc ファイルが存在しない場合に読み込まれるもの

新規の Vim ユーザーにとって使いやすい設定がまとめられている

コマンドラインモードでのタブ補完や、マウスの利用設定などがある

defaults.vim の詳細は `:help defaults.vim-explained`



## .vimrc の例

```
1 " から始まる行はコメント
2 " defaults.vim を確実に読み込ませる
3 unlet! skip_defaults_vim
4
5 " defaults.vim を読み込む
6 source $VIMRUNTIME/defaults.vim
7
8 " 日本語のヘルプファイルが利用できるときは利用する
9 set helplang=ja,en
10
11 "Tab を明示的に表示する。( \ のあとにはスペース )
12 set list
13 set listchars=tab:^\
```

## ノーマルモードでの操作

---

## モード間の移動

コマンド	意味	移行するモード
a	カーソルの後ろに文字を追加 (append)	挿入モード
A	行の後ろに文字を追加	挿入モード
i	カーソルの前に文字を挿入 (insert)	挿入モード
I	行の先頭に文字を挿入	挿入モード
o	カーソルの次の行に新しい行を追加 (open?)	挿入モード
O	カーソル位置に新しい行を追加	挿入モード
s	カーソル位置の文字を削除して挿入モード	挿入モード
S	行を削除して挿入モード	挿入モード
v	文字単位のヴィジュアルモード	ヴィジュアルモード
V	行単位のヴィジュアルモード	ヴィジュアルモード
C-v	矩形ヴィジュアルモード	ヴィジュアルモード
r	カーソル位置の文字を 1 文字置き換え (replace)	置換モード
R	カーソル下の文字をどんどん置き換えるモード	置換モード

# モーション

カーソルを移動させるコマンド

範囲を表すこともできる (後述)

数字と組み合わせて繰り返し回数を指定できる

i.e. 2w → 2 単語移動

コマンド	意味
h j k l	← ↓ ↑ →
w e b	単語単位の移動 (word, end, back)
\$ ^	行末・行頭へ移動
G gg	ファイル先頭・末尾へ移動
%	対応するカッコへ移動

その他の移動コマンドは `:help quickref`

コマンド	意味
y	ヤング (yank) (コピー)
d	削除 (delete)
c	削除して挿入モードに移行 (change)

オペレーターは範囲を表すコマンドと組み合わせる

i.e. c\$ → 行末まで削除して挿入モードに移行

## テキストオブジェクト

a (ひとつの “a” まとまり) または i (inner) と組み合わせて範囲を指定する

コマンド	意味
iw	カーソルがある単語
a( a)	( ) を含めた全体

ci{ とすれば { } の中身を削除して挿入モードに移行

## その他のコマンド

コマンド	意味
D	行末まで削除 (d\$)
C	行末まで削除して挿入モードに移行 (c\$)
dd	1 行削除
yy	1 行ヤंक
p	カーソルの後ろに貼り付け
P	カーソルの手前に貼り付け
x	1 文字削除
J	行を連結
u	アンドウ
C-r	リドゥ
.	繰り返し
ZZ	変更を保存して終了 (変更がなければ何もしない) (:x)
ZQ	何もせず終了 (:q!)

# ヴィジュアルモード

---

# ビジュアルモードとは

範囲を選択するモード

範囲を選択したあとに c d y などを組み合わせる

矩形ビジュアルモード (C-v) では、複数行への挿入 (I) や編集もできる

> や < でインデントを増減することができる



# コマンドラインモード

---

# コマンドラインモードでのコマンド

コマンド	意味
<code>:e filename</code>	<code>filename</code> を編集
<code>:w</code>	上書き保存
<code>:help</code>	ヘルプを表示
<code>:sp :vsp</code>	画面を分割表示
<code>:tabnew</code>	タブページを利用
<code>!:cmd</code>	外部コマンド <code>cmd</code> を使用
<code>:r!cmd</code>	<code>cmd</code> の結果を挿入

## .vimrc で自前のコマンドを定義できる

- 1 " :pLaTeX で、編集しているファイルを `platex` で処理する
- 2 " % は編集しているファイル名に展開される `:r` で拡張子を取り除く
- 3 `command pLaTeX !ptex2pdf %:r`