

# Bash のれんしゅう

## 初心者～中級者向け

えび ([rsk0315](#))

2018 年 7 月 9 日

## 0 はじめに

Bash で「g++ hoge.cpp」と「./a.out」くらいしか打ったことがない競技プログラマはそれなりに多いのではないのでしょうか。もしあなたがそれに該当しているのであれば、それは非常にもったいないことです。Bash を使いこなして周りに一目置かれるようになりましょう！

私としては「デフォルトで入っているシェルを使いこなせると、多くの環境に順応できてよさそうだなあ」くらいの気持ちです。

## 1 基本操作・キーバインド

キーバインドを覚えることはシェルを使いこなすことの第一歩です<sup>\*1</sup>。Bash のデフォルトのキーバインドは Emacs のキーバインドと似ています。vimmer の方々はこれを機に Emacs のキーバインドを覚えるもよし、Bash のキーバインドも vim-like にする（後述）もよしです。

C-x は「Ctrl を押しながらキー x を押す」、M-x は「Alt を押しながらキー x を押す」ことを指します。M-x が別の（OS など？）キーバインドと衝突する場合などは「ESC を押してからキー x を押す」ことで代用可能です。M-C-x は「Ctrl と Alt を押しながらキー x を押す」か「ESC を押してから、Ctrl を押しながらキー x を押す」ことを指します。また、x y は「x を押した後に y を押す」ことを指します。なお、「SPC」はスペースキー、「RET」はエンターキー、「DEL」は**バックスペースキー**を指します。

以下では、次のような形式でキーバインドを上げていきます。

- コマンド名：割り当てられたキー
  - それに関する説明

コマンド名は、それを別のキーに割り当てたいときや、キーバインドを解除したいときに必要になります。多くのコマンドは関連する単語の頭文字から来ているので、コマンド名を書いておくと覚えるときの手助けにな

---

<sup>\*1</sup> 他のあらゆることも第一歩として考えられますが、ここではこれを採用します。

るかと思って載せています。

## 1.1 デフォルトで有効なキーバインド

主なものを紹介します。これ以外にもいくつか存在していますが、初心者が覚えておくによさそうなものに絞りました。

### 1.1.1 カーソル移動に関するコマンド

- `forward-char` : `C-f`
- `backward-char` : `C-b`
  - 前後に一文字ずつ移動します。個人的には、この程度なら矢印キー（左右）に頼ってもいいかなと思っています。
- `forward-word` : `M-f`
- `backward-word` : `M-b`
  - 一単語分移動します。単語は英数字からなる列を指します。
  - `Ctrl` を押しながら矢印キー（左右）で代用可能かもしれません？
- `beginning-of-line` : `C-a`
- `end-of-line` : `C-e`
  - 行の先頭・末尾に移動します。アルファベットの先頭の `A` と `End` の `E` です（は？）

### 1.1.2 履歴に関するコマンド

- `accept-line` : `RET`
  - その行の内容を確定させます。これはおそらくみなさんが知っていますね。
- `previous-history` : `C-p`
- `next-history` : `C-n`
  - 履歴を一つずつ辿ります。矢印キー（上下）に頼ってもいいかなと思っています。
- `beginning-of-history` : `M-<`
- `end-of-history` : `M->`
  - 最も古い履歴、最も新しい履歴（現在の入力行）へ移動します。
- `reverse-search-history` : `C-r`
- `forward-search-history` : `C-s`
  - コマンドのインクリメンタル検索をします。
  - デフォルトでは、`C-s` はバインドされていますが**実行できません**（後述）
- `non-incremental-reverse-search-history` : `M-p`
- `non-incremental-forward-search-history` : `M-n`
  - インクリメンタルでない検索をします。
  - UI がひどくて、検索であることが初見ではわかりませんが、`:` が出てきていれば正しいです。
- `operate-and-get-next` : `C-o`
  - その行の内容を実行した後、次の行の内容を表示します。一続きのコマンドを再実行したいときに

便利です.

### 1.1.3 テキスト編集に関するコマンド