

# command line による editor 操作の習熟プログラムの開発

情報科学科 27014533 和田創熙

## 1 研究の動機

最初は西谷によって開発された shunkuntype(ターミナル上で実行するタイピングソフト)の再開発をテーマにしていたが、タイピングに特化したソフトを開発しても同じようなものが Web 上に大量に配布されており、それ以上の付加価値を持ったソフトを開発しようと考えた。

プログラマにとって作業を効率化する要因として editor 操作に関しては、

ツールはプログラマ自身の手の延長である。これは他のどのようなソフトウェアツールよりも Editor に対して当てはまる。テキストはできる限り簡単に操作できる必要があり、一つの強力な editor の習熟は作業の効率化に欠かせない [1]。

と記述されている。西谷研究室ではタイピング、Ruby 言語、Emacs による editor 操作、CUI 操作の習熟が作業効率に非常に大きな影響を与えるため習熟を勧めているのでこれらの習熟を目的としたソフトを開発しようと考えた。

## 2 editor\_learner の概要

本章では、editor\_learner のメソッドについて説明する。練習用コードは Ruby 入門の教科書を参考にしている [2]。

### 2.1 initialize

各種コマンドを実行した時、自動的に動くメソッドである。作業を行うためのファイルの作成、gem で install した先のパスを格納したインスタンス変数の定義がメインである。

### 2.2 random\_check

コマンド実行から終了までの動作の概要は以下の通り。

- 用意されたソースコードから 1 つ選ばれ、question.rb にコピーされる。
- 新しいターミナルが開かれ、question.rb の内容を answer.rb に写経する。
- 写経し終わると前のターミナルに戻り”check”とコマンドラインで入力する。
- 正しければ終了、正しくなければ間違えた箇所のみが表示され、再度確認、入力、正誤判定を繰り返す。

### 2.3 sequential\_check

コマンド実行から終了までの動作の概要は以下の通り。

- chapter と問題番号を引数として入力。対応した問題が

q.rb にコピーされる。

- 新しいターミナルが開かれ、写経する。
- 写経し終わった後の動作は 2.2 の 4 番からは random\_check と同じ動作である。

一度写経し終わった問題を再度行う時には、作成されたファイルの削除を行うメソッドが別途用意されている。

## 3 他のソフトとの比較

表 1 は人気タイピングソフトと editor\_learner の利点、欠点とユーザーインタフェースを表にしたものである。

CUI に近い PTYPING が高評価な理由はコードに関する単語を入力できるという点であった。それに加えて editor\_learner は editor 操作やコードの入力、実行ができる。

GUI ベースのソフトには継続性に関しては劣る。しかし CUI ベースであればキーバインド使用により作業の高速化、効率化が期待される。そのため editor\_learner はプログラマ向けの機能を多数有しているソフトである。

表 1 人気タイピングソフトと editor\_learner の利点、欠点とユーザーインタフェース。

	利点	欠点	UI
editor_learner	editor操作やキーバインドなどの習熟が可能	間違えた箇所がわかりにくい。	CUI
PTYPING	豊富なプログラム単語の入力可能	プログラムの実行、プログラムの入力が不可能。	CUI
e-typing	間違いの多かった箇所などがわかる。	同じ内容のものが多いため慣れてしまう。	GUI
寿司打	GUIにより、モチベーションが継続されやすい。	ローマ字入力のみなので約物の入力が一切ない。	GUI

## 4 終わりに

editor\_learner は {} や () など約物の入力やカーソル移動、ファイルの開閉、保存などの CUI 操作を全てキーバインドで行う。そのため、editor\_learner はプログラマにとって作業を効率化させるソフトと言え、ソフトの目的に沿った技術の向上が期待される。

## 参考文献

- [1] Andrew Hunt, David Thomas, 「達人プログラマー」, (オーム社, 2016 年)。
- [2] 伊藤淳一, 「プロを目指す人のための Ruby 入門」, (技術評論社, 2017 年)