

Web ブラウザ上のプログラミング学習環境 PyPEN の改良

中西 渉 watayan@meigaku.ac.jp

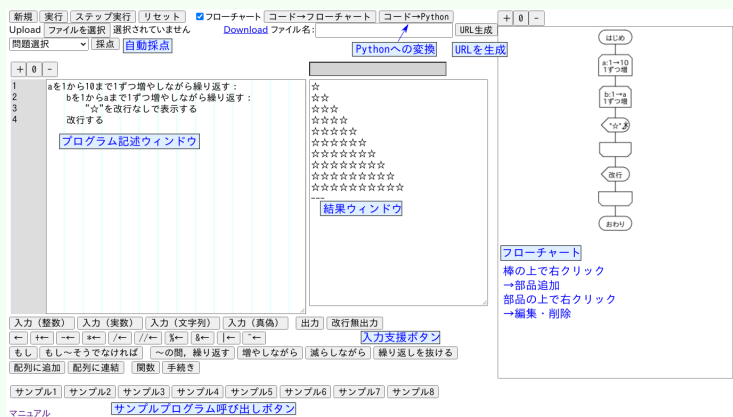
名古屋高等学校

■ PyPEN とは

- Web ブラウザ上のプログラミング学習環境
- xDNCL を Python 風に改良
インデントでブロックを表現
- PEN 譲りの「入力支援ボタン」
- PenFlowchart 譲りのフローチャート
コード→フローチャートは手動 (ボタン)
フローチャート→コードは自動
- 「コード→Python」ボタンで Python 用のコード出力
(必要なモジュールは import 文も)
- サンプルプログラム呼び出しボタン
- 自動採点機能つき
- JavaScript のみで実装
→ Web サーバ上でも、ローカルファイルでも実行可能
※ただし Internet Explorer は不可

[筆者のサイトの PyPEN のページ](#)

[GitHub の PyPEN のページ](#)



■ 改良点

- 演算子の追加・変更
 - ビット演算の追加
&とか|とか^とか
 - 複合代入演算子の導入
+<とか&<とか
 - =による代入も OK
だから+=とか&=とかも OK
(比較は=でも==でも OK ←これは以前から)
 - 文字列を「」から””に
- 配列操作を Python に近づける
「コード→Python」で出力するコードが Python でエラーにならないようにしたい
 - 値が代入されていない変数の参照はエラー
 - 配列の範囲外へのアクセスはエラー
 - Python の append 相当 → 「《配列》に《値》を追加する」
 - Python の extend 相当 → 「《配列》に《配列》を連結する」
 - 配列の添字に負数やスライスっぽいものを使えるようにする
 - 配列と辞書を分離

[フィボナッチ数列の例](#)

[スライスの例](#)

辞書の例は次のグラフの例を参照

- グラフ機能の強化 (Plotly の導入)
Monaca に影響されて実装
[グラフの例 \(フローチャートを非表示にするといい\)](#)
- URL でのプログラム読み込み
このポスター中の本章のリンクがそれ
URL の末尾に?code=に続いてコードを zlib 圧縮して Base64URL エンコードした文字列をつける
- MakeSample, MakeAnswer の Web アプリ化 (WaPENTools)
MakeSample : sample.js (サンプルのスクリプト) 生成プログラム
MakeAnswer : answer.js (自動採点の問題のスクリプト) 生成プログラム
Qt アプリで実装
→ SSS2019 「Web アプリの方がいいのでは」
→それはいい!
[筆者のサイトの WaPENTools のページ](#)
[GitHub の WaPENTools のページ](#)
- その他
 - 「コード→Python」の別窓表示を、結果ウィンドウ表示に変更
 - 「a, b, c を表示する」という構文を追加
 - 「《配列》の中に《値》という条件を追加 ([ガチャのプログラム例](#))
 - マニュアルを書いた

■ 今後の方針

- 構文拡張
PyPEN は xDNCL を「元になっている」が、xDNCL のプログラムは動かない
→ xDNCL にこだわる必要はなくなった (既に拡張している)
むしろ Python の構文をもっと取り入れるべきではないか
 - for in の構文
 - range にあたるもの
→どんな「日本語」にする?
- 変数のスコープの見直し
グローバル変数とローカル変数しかない
global にあたる区別がない
Python ではこんなコードが書けてしまうくらいきちんと (!) 管理している

```
for i in range(1,11):
    for i in range(i):
        print(i,end='')
    print('')
```

そこまで行かなくてももう少しスコープを整理したい

- 構文木の実行について
PyPEN は JavaScript への変換をしていない → 実行速度遅い
もうちょっと速くできないかなあ…
- WaPEN へのフィードバック
PyPEN にかかりきりで、WaPEN の開発が止まっている…
構文以外は大きい違いがないのだから、近いうちにフィードバックしたい

■ おわりに

PyPEN に「できないこと」はとても多い
「実用的」な機能を追加するか?
→むしろ教育用に特化するべきではないか
PyPEN で少し勉強すれば Python に移行できるようにしたい
ドリトルの [「1 時間で学ぶソフトウェアの仕組み」](#) のようなものを作りたい
(実際 Python への導入として PyPEN を使い、Python でわからなくなったら PyPEN に戻って考える、という使い方をさせているという実践の話は聞いている)

がついているところはリンクがあります。