

PyPEN における参照の値渡しの実装

中西 渉^{1,a)}

概要：筆者は大学入学共通テスト「情報」のプログラミング問題の表記に近い言語を用いたプログラム学習環境 PyPEN を開発している。PyPEN では値の代入や関数の引数についてはすべて値渡しで実装してきたが、配列や辞書について参照の値渡しも可能となる実装を行った。これは Python を含め多くの言語で行われている実装に近いものである。
本稿ではその実装の概要を示し、作成できるプログラムの変化について述べる。

Implementation of Pass by Value of Reference in PyPEN

Abstract: I have been developing a programming learning environment called PyPEN, which uses a language similar to the notation found in programming problems on the University Entrance Common Test in Informatics.

In PyPEN, all assignments and function arguments are originally implemented as pass-by-value. However, a new implementation enabled pass-by-value-of-reference for arrays and dictionaries, similar to the behavior seen in many programming languages including Python.

This paper outlines this implementation and discusses how it changes the kinds of programs that can be written in PyPEN.

1. はじめに

高校の教科「情報」は 2003 年度から必修である。それ以前は科目が選択必修であったものが 2022 年度から「情報 I」が共通で必修となり、さらに 2025 年度の大学入学共通テストに「情報」が導入されたことで、世間の情報科に関する認知はようやく向上したように思われる。

プログラミングでどのような言語を用いるかについては学習指導要領に定めがなく、教科書では複数の言語が採用されている。一方、大学入学共通テストでは日本語ベースの疑似コードを用いた出題がされており、2026 年度からの教科書ではそれを取り入れたものも刊行されるようである。

筆者は大学入学共通テストで使われるものに近い言語を用いたプログラミング学習環境 PyPEN[1] を開発している。もちろんこれを用いて授業をすることは教科の目的の一つでもある「問題解決」には適さないが、入試対策の演習や参考書では利用されている様子である。

この言語(?) はアルゴリズムを示すことが主目的であるから、明確な仕様は定められていない。そのため PyPEN

では筆者が考える仕様での実装を行ってきた。その中で、代入や関数の引数についてはすべて値渡しで実装してきたが、(多くの言語がそうしているように) 配列や辞書などの大きいオブジェクトについては参照渡しに適しているとも考えられる。

そこで PyPEN では基本的な値は値渡し、配列や辞書が参照の値渡しになるような実装を行った。本稿ではその実装の概要を示し、作成できるプログラムの変化について述べる。

2. PyPEN について

2.1 PyPEN 開発に至る経緯

2005 年頃、大阪市立大学松浦研究室および大阪学院大学西田研究室において、初学者向けプログラミング学習環境 PEN (Programming Environment for Novice) が開発された [2]。PEN で用いられる言語はセンター試験の「情報関係基礎」のプログラミング問題で用いられる疑似言語 DNCL を元にしたものであった。構文の雛形が入力できる「入力支援ボタン」を用いることで、日本語表記の揺れを防ぐなどの工夫がなされていた。

筆者は 2011 年頃に、PEN にフローチャート表示など

¹ 名古屋高等学校
Nagoya Senior High School
^{a)} watayan@meigaku.ac.jp

を追加した PenFlowchart を開発した。また PEN や Pen-Flowchart は Java アプリケーションであったため、Web ブラウザ上で動作するものを開発したいと考えた。そこで 2018 年から 2019 年にかけてこれらを Web アプリケーションとして作り直した WaPEN や、その構文を Python に近づけた PyPEN を開発した。

2.2 PyPEN の特徴

PyPEN は筆者が開発しているプログラミング学習環境である。日本語をベースにした言語を用いているが、Python のようなインデントによるブロック構造を持つ。大学入学共通テストのプログラミング問題がこれに近い形式であることが判明してからは、ある程度それに近づけるようにしてきた。

HTML, CSS, JavaScript を用いて実装されており、Web ブラウザ上で動作する。その際には Web サーバの特別な機能は使用しないので、ファイルをダウンロードしてローカル環境で動作させることもできる。

3. ○○渡し

3.1 ○○渡しの違い

引数の渡し方には「値渡し」「参照渡し」「参照の値渡し」といったものが考えられる、とは書いてはみたが、この分類には様々な考え方がある。ネットで検索してみると、渡すものの違いなのか、渡し方の違いなのか、また、参照渡しは参照という値を渡しているのかといったことについての意見が散見される*1。本稿ではそれを議論するつもりはない。

とはいえ区別をする必要はあるので、便宜上図 1 のようなプログラムでどのように振る舞うかによって表 1 のように分類することにする。これはあくまでも本稿で区別するために決めたものであって、世間一般で使われているものと同じだと主張しているわけではないことに注意されたい。

```
手続き f(a, b):  
  a[0] = 100  
  b = [7, 8, 9]  
  
a = [1, 2, 3]  
b = [4, 5, 6]  
f(a, b)  
表示する(a, b)
```

図 1: 変数への引数の例

どれを適用するかは言語の仕様や、制御するためのキーワードなどにより異なる。

*1 そしてしばしば「○○渡しは存在しない」という話になる。

表 1: 引数の渡し方による違い

引数の渡し方	a	b
値渡し	[1, 2, 3]	[4, 5, 6]
参照渡し	[100, 2, 3]	[7, 8, 9]
参照の値渡し	[100, 2, 3]	[4, 5, 6]

4. PyPEN における実装

4.1 実装の方法

PyPEN では変数管理テーブルという名のハッシュテーブルを作って変数名と値を対応付けている(図 2)。すべての値は Value クラスのインスタンスであり、変数名とそのインスタンスを結びつけるのが変数管理テーブルである。したがって、変数管理テーブル上で同じインスタンスに変数名を対応付ければ参照の値渡しは実装できる*2。

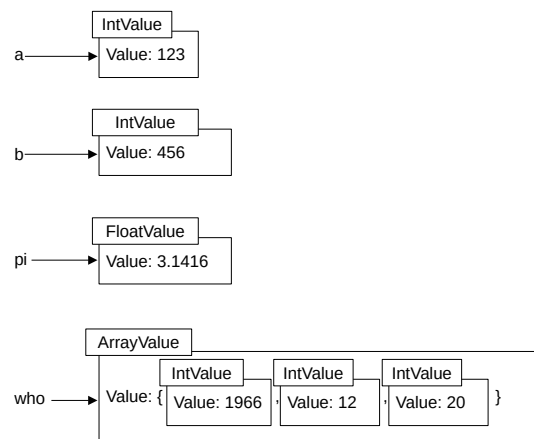


図 2: 変数テーブルの様子

従前は関数や手続きの引数として指定した変数は、複製したインスタンスを指すようにしていた(図 3)。これを(配列や辞書については)そのまま同じインスタンスを指すようにしたただけのことである(図 4)。

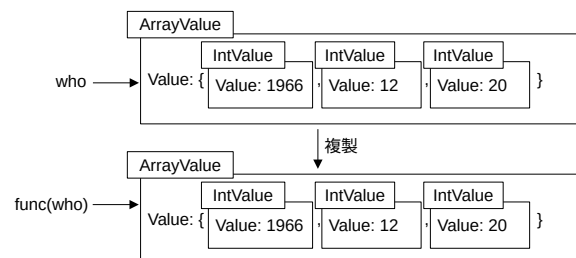


図 3: 値渡し

とはいっても値渡しが必要な場合もあるので、そのときには copy を使うことにした。

*2 変数名と値の対応自体をクラス化すれば参照渡しも実現可能かと考えるが、実装はしていない。

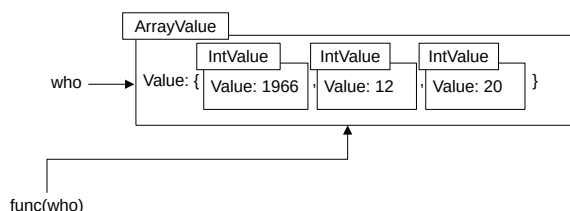


図 4: 参照の値渡し

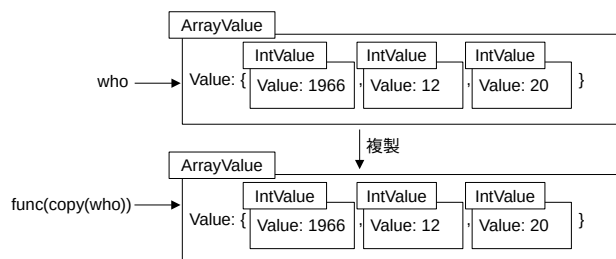


図 5: copy を使った値渡し

4.2 実装に伴う変更点

新しく組み込みメソッドとして copy を実装した。その他、配列操作を行う push, pop, shift, unshift を追加した。その理由については後述する。

5. 考察

電気通信大学 2025 年度入試問題（情報理工学域，前期日程）の大問 3[3] は，note の記事 [4] にある詳細な解説の通り，逆ポーランド記法の計算をスタックやキューで行うというものである。問題では「入れる ()」が記憶領域 M に対する push，「取り出す 1()」が pop，「取り出す 2()」が shift に相当する関数として用意されている。

従前の PyPEN ではスタック（あるいはキュー）を操作するにあたって，処理されたスタック（あるいはキュー）を返す関数を作ってその戻り値を同じ名前の変数に代入する方法しかできなかった（図 6）。この過程で配列のコピーが複数作られるので非常に非効率であるし，pop や shift では変化した配列と，取り出した値の両方を戻り値にする必要があるため，その 2 つを要素とする配列を返す必要があった^{*3}。

当初は新しい PyPEN で図 7 とすればいいと考えたのだが，M の参照が変わってしまうので元の配列には変更が及ばない。そのため push や pop, shift, unshift を組み込みメソッドとして実装することになった。

結局のところ，一定の破壊的な組み込みメソッドを実装する必要があると考えられる。その対応として clear や del を実装しようとは考えている。しかし append や extend にあたるものを「M に v を追加する」「～に～を連結する」というような構文を作って文法レベルで定義していたこととの整合性が危うい。

関数 push(M, v):
M に v を追加する # Python の M.append(v) に相当
M を返す

関数 pop(M):
v = M[-1]
{M[:-1], v} を返す

関数 shift(M):
v = M[0]
{M[1:], v} を返す

```
M = []
M = push(M, 1)
a = pop(M)
M = a[0]
v = a[1]
```

図 6: 従前の PyPEN でのスタック操作

手続き push(M, v):
M に v を追加する

関数 pop(M):
v = M[-1]
M = M[:-1] # 破壊的な del メソッドがあれば del(M, -1)
v を返す

関数 shift(M):
v = M[0]
M = M[1:] # 破壊的な del メソッドがあれば del(M, 0)
v を返す

```
M = []
push(M, 1)
v = pop(M)
```

図 7: 新しい PyPEN でのスタック操作（失敗）

6. おわりに

大学入学共通テストの疑似コードについて，大学入試センターの方から仕様を細かく定めることはないだろう。したがって「それに近い実行環境」を作るなら，細部は実装者がそれぞれに決める必要がある。

大げさにいえばこの作業は言語の実装なのであるから，その者の技術的思想によって実装は異なるものになるだろう。冒頭に述べたように，PyPEN はあくまでも演習用の環境であり，そのような細部が問題になることは現実的には発生しないはずである。

しかし（筆者も含めて）せっかく言語があるのだからいろいろ作ってみたい人は世の中にいる。WaPEN でコンウェイのライフゲームを作った生徒も過去にいたし，PyPEN で Brainf*ck のインタプリタを書いた人もいた。ドリトルや Scratch のような広い世界を持つためには，教材という枷から開放されることが求められるのだろう。

^{*3} Python のアンパック代入に相当するものを作れば話は別。

参考文献

- [1] 中西 渉: プログラミング学習環境 PyPEN の開発, 日本情報科教育学会 第 13 回研究会報告書 (2019).
- [2] 中村亮太, 西田知博, 松浦敏雄: プログラミング入門教育用学習環境 PEN, 情報処理学会研究報告 コンピュータと教育 (CE), Vol. 2005-CE-081, No. 104 (2005).
- [3] 電気通信大学: 2025 年度個別学力検査入試問題 (情報理工学域前期日程), 電気通信大学 (オンライン), 入手先 (<https://www.uec.ac.jp/education/undergraduate/admission/pdf/2025zen-a.pdf>) (参照 2025-05-19).
- [4] 安田 豊: 電気通信大学の 2025 年度入試問題から, 大問 3 - 数式を処理するプログラムを作る, 情報処理学会 (オンライン), 入手先 (<https://note.com/ipsj/n/n4aed36c58756>) (参照 2025-05-19).