

WaPEN および
PyPEN における
新機能の実装

中西渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

WaPEN および PyPEN における 新機能の実装

中西渉

watayan@meigaku.ac.jp
名古屋高等学校

2021-11-13

日本情報科教育学会第 17 回研究会

- 1 はじめに
- 2 WaPEN, PyPEN について
- 3 WaPEN, PyPEN に行った追加実装
 - 変数宣言の廃止
 - 表示についての変更
 - 配列の初期化方法
 - 配列操作
 - 複合代入演算子
 - URL 生成
 - 辞書型のデータ
 - Plotly によるグラフ描画
 - File I/O もどき
- 4 おわりに

1 はじめに

DNCL…「情報関係基礎」プログラミング問題で使われる擬似言語

2006 PEN（大阪学院大，大阪市立大で開発）

2012 PenFlowchart（フローチャートを付加）

2018 WaPEN（Web ブラウザ上に移植）

2019 PyPEN（構文を Python 風アレンジ）

「情報 I」試作問題，サンプル問題…新しい DNCL（Python 風）

→ WaPEN，PyPEN に新しい機能を追加

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

- 1 はじめに
- 2 WaPEN, PyPEN について
- 3 WaPEN, PyPEN に行った追加実装
 - 変数宣言の廃止
 - 表示についての変更
 - 配列の初期化方法
 - 配列操作
 - 複合代入演算子
 - URL 生成
 - 辞書型のデータ
 - Plotly によるグラフ描画
 - File I/O もどき
- 4 おわりに

2 WaPEN, PyPEN について

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

WaPEN

- DNCL で書かれたプログラムを Web ブラウザ上で実行できるプログラミング学習環境
- コード ↔ フローチャート
- ~~Python~~ に変換したコード (これは PyPEN の話)
- 同様に DNCL を使った学習環境
 - wPEN
 - どんくり
 - Tetra
 - XTetra

PyPEN...DNCL を Python 風のアレンジ

PEN

もし $a \geq 20$ ならば
| 「成人」を表示する
を実行する

PyPEN

もし $a \geq 20$ ならば:
表示する ("成人")

「情報 I」 試作問題, サンプル問題もこんな感じ...

- 前述の環境もこの新 DNCL に対応
- 「なでしこ」も対応表で読み替え可能
- Pictogramming も対応 (Picthon)

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

MIT ライセンス

サーバは使用しないのでローカルで使用可能
サンプルや問題の差し替えは WaPEN Tools で

- 1 はじめに
- 2 WaPEN, PyPEN について
- 3 WaPEN, PyPEN に行った追加実装
 - 変数宣言の廃止
 - 表示についての変更
 - 配列の初期化方法
 - 配列操作
 - 複合代入演算子
 - URL 生成
 - 辞書型のデータ
 - Plotly によるグラフ描画
 - File I/O もどき
- 4 おわりに

3 WaPEN, PyPEN に行った追加実装

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

PyPEN にかかりきり
WaPEN の開発が停滞
→ PyPEN を WaPEN に移植

両方に機能追加が可能

3.1 変数宣言の廃止

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

PyPEN は変数宣言なし
→移植した WaPEN も変数宣言なし
むしろ本来の DNCL に近づいた

変数が型をもたない
→入力を変更する必要がある
Python 風の `n=int(input())` ←忘れるとエラー

従来

整数 `n`
`n` に入力する

変更後

`n` に整数を入力する

3.2 表示についての変更

WaPEN および
PyPEN における
新機能の実装

中西渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

コンマ区切りで複数の値を指定

新規 実行 ステップ実行 リセット 変数確認 *

フローチャート コード→フローチャート コード→Python URL生成

Load Save ファイル名:

問題選択 採点

+ 0 -

1 表示する ("ABC"と"DEF"と123)	ABCDEF123
2 表示する ("ABC", "DEF", 123)	ABC DEF 123

3.3 配列の初期化方法

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

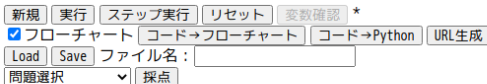
おわりに

従来の方法

- `a=[0,0,0,0,0,0]`

新しい方法（両者は微妙に違う）

- `a=6個の0`
- `a=[0]*6`



```
1 a=6個のrandom(5)
2 b=[random(5)]*6
3 変数を確認する
```

```
*** 変数確認 ***
a:[3,1,4,0,0,3]
b:[3,3,3,3,3,3]
---
```

長さの指定されていない配列の初期化

- 《配列》のすべての要素を《値》で初期化する

PyPEN では…

- Python でそのまま実行できるコードにしたい
- ∴ 実装する予定はない

WaPEN では…

- 実装してもいいかも
- 「どんくり」を真似ると良さげ

3.4 配列操作

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

《配列》の中に《値》

- Python の「《値》 in 《配列》」

語順が違うのはイマイチだが...

*

フローチャート

ファイル名:

```
1 a=100個のrandom(100)
2 もしaの中に50ならば:
3   ...表示する("Found!")
4 そうでなければ:
5   ...表示する("Not found!")
6 aを表示する
```

```
Found!
[100,67,57,4,68,31,8
,41,60,19,40,72,89,4
9,94,49,13,93,51,68,
100,55,35,64,14,57,0
,57,94,80,90,82,96,3
1,64,12,61,61,39,36,
100,94,33,45,67,14,2
8,97,31,30,64,60,66,
57,35,42,19,50,62,5,
89,16,88,48,18,91,64
,41,57,13,82,30,54,8
3,77,70,44,46,19,0,7
1,36,49,50,27,57,11,
91,79,5,49,98,86,25,
82,71,53,13,11,99]
```

《配列》の要素《変数》について繰り返す：

- Python の「for 《変数》 in 《配列》：」

語順が違うのはイマイチだが...

The screenshot shows the WaPEN/PyPEN web interface. At the top, there are several buttons: 新規 (New), 実行 (Run), ステップ実行 (Step Run), リセット (Reset), and 変数確認 (Check Variables) with an asterisk. Below these is a checked checkbox for フローチャート (Flowchart) and buttons for コード→フローチャート (Code to Flowchart), コード→Python (Code to Python), and URL生成 (Generate URL). There are also buttons for Load and Save, a text input for 文件名 (Filename), a dropdown for 問題選択 (Problem Selection), and a button for 採点 (Marking). Below the controls is a code editor with a line number column (1-8) and a toolbar with +, 0, and - buttons. The code in the editor is:

```
1 a=100個のrandom(100)
2 sum1=0
3 sum2=0
4 iを0からlength(a)-1まで1ずつ増やしなが：
5   ...sum1=sum1+a[i]
6 aの要素pについて繰り返す：
7   ...sum2=sum2+p
8 表示する(sum1,sum2)
```

To the right of the code editor is a terminal window showing the output:

```
4585 4585
---
```

配列の添字に負数やスライス

*

フローチャート

ファイル名:

▼

+ 0 -

```
1 a=["A","B","C","D"]
2 表示する(a[-1])
3 表示する(a[1:3])
```

```
D
[B,C]
---
```


3.5 複合代入演算子

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

DNCL にある構文

《変数》を《値》増やす

C などの言語

《変数》 += 《値》

そんなわけで真似した

+=, -=, *=, /=, ...

WaPEN は代入が「←」 ... 「+=」「+←」両方に対応

3.6 URL 生成

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

たとえばこんな風にリンクを埋め込むことができる

新規 実行 ステップ実行 リセット 変数確認
 フローチャート コード→フローチャート コード→Python URL生成
Load Save ファイル名:
問題選択 採点

+ 0 -	
<pre>1 a=0 2 b=random(8)+1 3 表示する("1から9の数字を当ててください") 4 a!=bの間: 5 ... aに整数を入力する 6 ... 表示する(a) 7 ... もしa>bならば: 8 ... 表示する("大きい") 9 ... そうでなければ: 10 ... もしa<bならば: 11 ... 表示する("小さい") 12 表示する("あたり") 13 . 14 .</pre>	<pre>https://watayan.net/ prog/PyPEN/? code=eJxtj8EuxkAUhff zFMyqExvdkah3mcYWiTc 4EyRaIjTVIiKk- FMhXVgJT3MyP1Zeode0F qWTuzp3vu_k2mRZpcm03 drY3oxWzFKsPm9nH_Urc U6XRzomcrqDVeJ5Xrb- qaI79e8Fcr_mmlghSmJX G2UXk1T- fZ0V328XakGeJR7n5YuQ P9jenc8ue3HYjppqsCRmd Iyq7LqJGeon21_WP0L5- II767sDiiTgnJG2IE7rD P_ggX5uWTxS0ct9w3HhD iOmaLtOmAxVfPp2Q</pre>

3.7 辞書型のデータ

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

新規 実行 ステップ実行 リセット 変数確認 *

フローチャート コード→フローチャート コード→Python URL生成

Load Save ファイル名:

問題選択

+ 0 -

```
1 animals={
2     ... "犬": "わん",
3     ... "猫": "にゃあ",
4     ... "雀": "ちゅん"}
5
6 keys := keys(animals)
7 keysの要素keyについて繰り返す:
8     ... 表示する(key, "は", animals[key], "と鳴く")
```

```
犬 は わん と鳴く
猫 は にゃあ と鳴く
雀 は ちゅん と鳴く
---
```

3.8 Plotlyによるグラフ描画

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

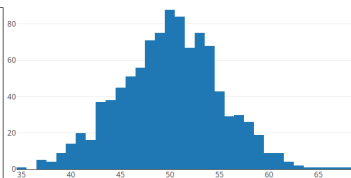
おわりに

Monaca の真似

新規 実行 ステップ実行 リセット 変数確認 *
 フローチャート コード → Python URL 生成
Load Save ファイル名:
問題選択 採点

+ 0 -

```
1 d=[]
2 iを1から1000まで1ずつ増やしながら:
3 ---- a=100個のrandom(1)
4 ---- s=0
5 ---- jを0から99まで1ずつ増やしながら:
6 ----- s+=a[j]
7 ---- dにsを追加する
8 グラフ描画({}, [{"type": "histogram", "x": d}])
```



3.9 File I/O もどき

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

Web ストレージで File I/O の真似事

新規 実行 ステップ実行 リセット 変数確認 *

フローチャート コード→Python URL生成

Load Save ファイル名:

問題選択 採点

```
+ 0 -
```

```
1 f=openr("crypt.txt")
2 Angoubun=[]
3 c=getchar(f)
4 cl="**の間：
5 --- Angoubunにcを追加する
6 --- c=getchar(f)
7 close(f)
8 a2z="abcdefghijklmnopqrstuvwxyz"
9 Hirabun=""
10 iを0からlength(Angoubun)-1まで1ずつ増やしながら：
11 --- bangou=sabun(Angoubun[i])
12 --- もしbangou!=-1ならば：
13 ----- Hirabun+=a2z[(bangou+16)%26]
14 --- そうでなければ：
15 ----- Hirabun+=Angoubun[i]
16 表示する(Hirabun)
17 -
```

they gave the last full measure of devotion -- that we here highly resolve that these dead shall not have died in vain -- that this nation, under god, shall have a new birth of freedom - and that government of the people, by the people, for the people, shall not perish from the earth.

アップロード ダウンロード 削除 全削除

crypt.txt

- 1 はじめに
- 2 WaPEN, PyPEN について
- 3 WaPEN, PyPEN に行った追加実装
 - 変数宣言の廃止
 - 表示についての変更
 - 配列の初期化方法
 - 配列操作
 - 複合代入演算子
 - URL 生成
 - 辞書型のデータ
 - Plotly によるグラフ描画
 - File I/O もどき
- 4 おわりに

4 おわりに

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

DNCL によるプログラミング演習は好ましくない?

- 読むための擬似言語→書くのには向いてない
- 紙上の演習，発展しない実習にならないか

すべては問題解決につながるべき

WaPEN, PyPEN がいい「踏み台」になることを望む

WaPEN および
PyPEN における
新機能の実装

中西 渉

はじめに

WaPEN, PyPEN
について

WaPEN, PyPEN
に行った追加実装

変数宣言の廃止

表示についての変更

配列の初期化方法

配列操作

複合代入演算子

URL 生成

辞書型のデータ

Plotly によるグラフ描画

File I/O もどき

おわりに

WaPEN, PyPEN の公開場所

筆者のサイト <https://watayan.net/prog/>

GitHub <https://github.com/watayan/WaPEN>

<https://github.com/watayan/PyPEN>