

# DNCLによる授業実践および 「情報I」への転換

中西渉

watayan@meigaku.ac.jp  
名古屋高等学校

2022-02-19

- ① はじめに
- ② DNCL について
- ③ 授業実践
- ④ 「情報Ⅰ」に向けての展望

# はじめに

2022年度「情報I」開始

→ プログラミングが必須

- どのような言語を使うか
  - 「教員研修用資料」は Python 寄り
  - 教科書見本 → Python, JavaScript, VBA, Scratch あたり
- DNCL は?
  - 「情報関係基礎」で使用
  - 試作問題・サンプル問題で使用（新 DNCL）

- ① はじめに
- ② DNCL について
- ③ 授業実践
- ④ 「情報 I」に向けての展望

# 現在までの経緯

## DNCL とは

大学入試センター試験・大学入学共通テスト「情報関係基礎」プログラミング問題で使用されてきた、日本語ベースの擬似言語

1997年 BASIC, COBOL, Pascal で出題

1998年 日本語ベースの擬似言語「読めばわかる」

2002年 仕様公開

2003年 DNCL と命名

2011年 仕様が事前公開

2021年 共通テストにあわせて仕様調整

2025年 新 DNCL(?)

## 現 DNCL

```
n ← 【外部からの入力】
n > 1 の間,
|   もし n%2=0 ならば
|   |   n ← n ÷ 2
|   |   を実行し, そうでなければ
|   |   |   n ← 3*n+1
|   |   |   を実行する
|   |   n を表示する
|   を繰り返す
```

## 新 DNCL (?)

```
n = 【外部からの入力】
n > 1 の間繰り返す:
|   もし n%2=0 ならば:
|   |   n = n / 2
|   |   そうでなければ:
|   |   |   n = 3*n+1
|   |   |   表示する (n)
```

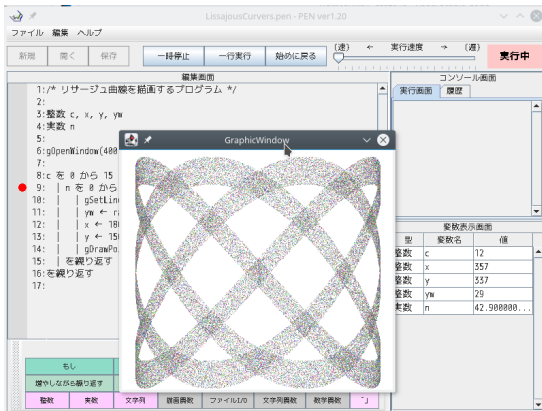
## Python だと...

```
n = int(input())
while n > 1:
    if n % 2 == 0:
        n = n / 2
    else:
        n = 3 * n + 1
print(n)
```

# 実行環境

PEN(Programming Environment for Novice) が TUATLE に対応

- 変数宣言などを追加→ xDNCL と命名
- 入力支援ボタン



## PenFlowchart

- フローチャート ↔ プログラム
- 実行部分はPEN そのまま

The screenshot displays the PenFlowchart software interface. On the left, a flowchart is shown with the following steps:

- はじめ
- $a = 0$
- $b = \text{random}(0) + 1$
- 「1から9の数字を当ててください」
- $a + b$  入力
- 「大きい」
- 「小さい」
- $a = b$  になるところまで

On the right, the PEN code is displayed:

```

1: 整数 a, b
2: a ← 0
3: b ← random(9) + 1
4: 「1から9の数字を当ててください」 を表示する
5: 繰り返し、
6: | a を入力する
7: | もし a > b ならば
8: | | 「大きい」 を表示する
9: | | を実行し、もう一度だけば
10: | | もし a < b ならば
11: | | | 「小さい」 を表示する
12: | | を実行する
13: | を実行する
14: 否、a = b になるところまで実行する
15: 「あたり」 を表示する
16:

```

At the bottom, there is a table titled "プログラム入力方法選択" (Program Input Method Selection):

なし	もし〜そとでなければ	〜の値、繰り返す	〜になるところまで実行する
繰り返しの回数を指定する	繰り返しの回数を指定する	入力	繰り返し入力
実行	実行	入力	入力
実行	実行	実行	実行



## PEN, PenFlowchart は Java アプリケーション

- インストールの手間
- JRE を敬遠する空気

### →Web ブラウザ上の DNCL 学習環境

- WaPEN
- どんくり
- wPEN
- Tetra, XTetra

### 新 DNCL への対応も

- PyPEN
- つちのこ
- ...

- ① はじめに
- ② DNCL について
- ③ 授業実践**
- ④ 「情報Ⅰ」に向けての展望

# 勤務校の環境

## 名古屋高等学校

- キリスト教主義
- 私立男子校（併設型中高一貫校）
- 高校 12 クラス/学年，中学 6 クラス/学年

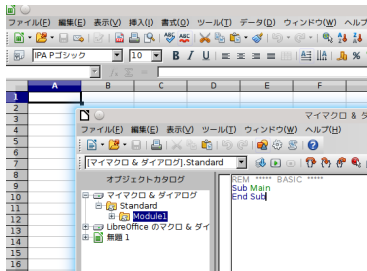
2003 年度に間に合うようにメディアセンター設置

- 情報教室 3，プレゼンテーションルーム 1
- 生徒端末は Linux
- ネットブート → Clonezilla によるディスクイメージ配布
- アカウントは LDAP で管理
- ホームディレクトリは NFS でマウント

# PEN 以前

2004 年度...OpenOffice.org の表計算マクロ

- セルと変数のやり取りが（VBA より）面倒  
→ 穴埋め形式で実習



```
Sub practice1  
oSheet = ThisComponent.getSheets.getByNamed("level1")  
a = oSheet.getCellByPosition(1,3).Value  
b = oSheet.getCellByPosition(1,4).Value  
-----  
' a と b の値を入れ替えなさい。  
-----  
oSheet.getCellByPosition(1,7).Value = a  
oSheet.getCellByPosition(1,8).Value = b  
End Sub
```

いじってはいけないところを間違っって編集するなどのトラブル  
→ 生徒には不評

## 2005 年度...Yabasic

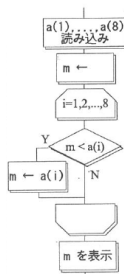
- テキストエディタで記述
- コンソールで実行

```

watayan@wata-x280:~$ yabasic test.bas
9
watayan@wata-x280:~$

test.bas - KWrite
data 6,3,4,9,7,5,2,4
dim a(8)
for i=1 to 8
  read a(i)
next i
m=a(1)
for i=1 to 8
  if m<a(i) then
    m=a(i)
  endif
next i
print(m)

```



## 課題6 最大値

```

m
i
data 6,3,4,9,7,5,2,4
dim a(8)
for i=1 to 8
  read a(i)
next i
m =
for i=1 to 8
  if m < a(i) then
    m = a(i)
  endif
next i
print s

```

# PEN

## 「教育用プログラミング言語に関するワークショップ 2006」

- Java
- JavaScript
- C/C++
- Basic
- Logo
- Lisp
- バンダイ「ワンダーボーグ」
- Squeak
- 言霊
- ひまわり/なでしこ
- ドリトル
- Tonyu
- PEN
- HSP
- Rosetta
- Viscuit

2006年度からPENを使用

→ 生徒には好評

- すべてが1画面
- 入力支援ボタン
- 日本語ベース

# PenFlowchart

PENで生徒がよくやる間違い

## 正しいプログラム

もし  $x=0$  ならば  
| 「ゼロ」を表示する  
を  
実行する

## よくある間違い

もし  $x=0$  ならば  
「ゼロ」を表示する

日本語的には正しいのだが...  
構文を壊さないようにしたい

- フローチャートでプログラムを生成しては?
- PenFlowchart の開発



## PenFlowchart の実行画面

The screenshot displays the PenFlowchart application interface. The main window is titled "PenFlowchart ver2.17" and shows a flowchart for a program. The flowchart starts with "はじめ" (Start), followed by "a ← 0", "b ← random(8)+1", and a loop "「1から9の数字を当ててください」" (Guess a number from 1 to 9). The user enters a value 'a'. A decision diamond "a > b" leads to "「大きい」" (Too big) if true, and another decision diamond "a < b" leads to "「小さい」" (Too small) if true. The process continues until "a = b" (Correct), then "a - b" and "になるまで" (until it becomes). The flowchart is displayed in a window titled "PEN ver1.20".

The execution window shows the following code:

```

1: 整数 a,b
2: a ← 0
3: b ← random(8)+1
4: 「1から9の数字を当ててください」 を表示する
5: 繰り返し,
6: | a を入力する
7: | もし a>b ならば
8: | | 「大きい」 を表示する
9: | | を実行し、そうでなければ
10: | | | もし a<b ならば
11: | | | | 「小さい」 を表示する
12: | | | | を実行する
13: | | | を実行する
14: | を、a=b になるまで実行する
15: 「あたり」 を表示する
16:

```

The execution window also shows a console output:

```

1から9の数字を当ててください
5
大きい
1
小さい
3
小さい
4
あたり

```

The execution window also shows a variable display table:

変数表示画面		
型	変数名	値
整数	a	4
整数	b	4

The execution window also shows a program input button table:

プログラム入力支援ボタン						
もし	もし~そうでなければ	~の裏、繰り返し	~になるまで実行する			
繰り返しの裏	繰り返しの裏	入力	出力	実行時出力	代入	
整数	実数	文字列	画面戻数	ファイル名	文字列戻数	数学戻数

# WaPEN

PenFlowchart は比較的好評だが

- フローチャートの画面が別
- インストールの手間→自宅でやりづらい

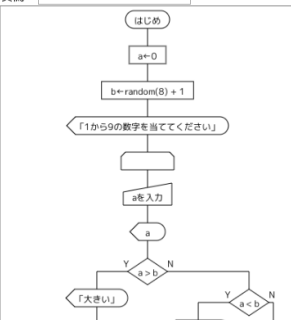
→Web アプリケーション化

フローチャート
  コード→フローチャート

Upload 
 選択されていません
 
 ファイル名:

1	整数 a,b	1から9の数字を当ててください
2	a←0	6
3	b←random(8)+1	大きい
4	「1から9の数字を当ててください」を表示する	2
5	繰り返し,	小さい
6	aを入力する	4
7	aを表示する	大きい
8	もしa>bならば	3
9	「大きい」を表示する	あたり
10	を実行し、そうでなければ	---
11	もしa<bならば	
12	「小さい」を表示する	
13	を実行する	
14	を実行する	
15	を、a=bになるまで実行する	
16	「あたり」を表示する	
17		
18		
19		
20		
21		

整数	a,b	サイズ:
実数		サイズ:0
文字列		サイズ:1
真偽		



## PyPEN

2019 夏

某氏 WaPEN の構文を Python っぽくしたの作ってみたい?  
 筆者 作ってみた

The screenshot displays the PyPEN web interface. At the top, there are navigation buttons: 新規, 実行, ステップ実行, リセット, and 変数確認. Below these are checkboxes for フローチャート (checked), コード→フローチャート, and コード→Python, along with a URL生成 button. There are also Load, Save, and ファイル名: fields, and a 問題選択 dropdown menu.

The main area is split into two panes. The left pane shows the Python code:

```

1 a=1
2 bを1から100まで1ずつ増やしながら:
3 .....a=a*b
4 .....表示する(bと"!="とa)

```

The right pane shows the execution output:

```

21=2
31=6
41=24
51=120
61=720
71=5040
81=40320
91=362880
101=3628800
111=39916800
121=479001600
131=6227028000
141=87178291200
151=1307674368000
161=20922789888000
171=355687420096000
181=6402373765728000
実行時エラーです
3行目:整数で表される範囲を越えました

```

At the bottom right, a flowchart visualizes the program's logic:

```

graph TD
    Start([はじめ]) --> A[a=1]
    A --> B[b=1→100  
1ずつ増]
    B --> C[a=a*b]
    C --> D{bと"!="とa}
    D --> E[ ]
    E --> F([おわり])

```

Below the code panes are input/output options: 入力(整数), 入力(実数), 入力(文字列), 入力(真偽), 出力, and 改行無出力. There are also buttons for 代入, 増やしながら, 減らしながら, 配列の要素について, 繰り返しを抜ける, 配列に追加, 配列に連結, 関数, 手続き, 値を返す, and 手続きを抜ける. At the bottom, there are buttons for サンプル1 through サンプル8.

At the bottom right of the flowchart area, there are buttons for アップロード, ダウンロード, 削除, and 全削除, and a text input field containing "crypt.txt".

まさか試作問題・サンプル問題がその形式だとは...

- ① はじめに
- ② DNCL について
- ③ 授業実践
- ④ 「情報 I」に向けての展望

# 「情報 I」に向けての展望

教科書見本で用いられている言語

- Python
- JavaScript
- VBA
- Scratch

どれを使っても結果に大差はない [24]

[24] 大学入学共通テストを見据えたプログラミング教育の言語選択に関する考察  
井手広康 情報処理学会研究報告コンピュータと教育 (CE) 2022-CE-163 No.14

## 試作問題・サンプル問題では新 DNCL

→ 新 DNCL で演習するといい? (入試対策?)

少し懸念していることは、DNCL だけを使ってプログラミングの授業が行われることがないように、ということです。やはり授業では、教科書で使われている実用的なプログラム言語を使っていただきたい。(略) プログラミングの授業は、教科書に掲載されてる言語を使って教えることになると思いますが、いずれかのプログラム言語で学んだ生徒であれば、この DNCL の仕様を知らなくても、問題の中に出てくるプログラムは、無理なく理解できると考えていますし、その逆、つまり DNCL は特異な仕様ではありませんので、問題の中のプログラムを日頃授業で使ってる言語で試すこともできると考えています。

[25] 大学入学共通テスト 新科目「情報 I」～サンプル問題等とそのねらい～  
水野修治 (大学入試センター問題調査官)

## 筆者の DNCL 観

- DNCL「だけ」の授業ではいけない  
「情報 I」の学習活動にプログラミングが活かされるべき  
→ 実用的な言語を使いこなせるようにしたい
- 導入としての意味はある  
思ったことを書くための補助輪

