

# PENの動作を制御する

Property.ini 内で指定できる設定項目は以下の通り。パラメータは起動時に読み込まれる。

## 1 変数宣言の要/不要の設定

変数宣言の要/不要を設定する。

[使用例]

```
executer.var.declaration=0
```

設定値は以下の通り (デフォルト値は 0)。

	設定値	変数宣言がない場合の出力
0	変数宣言必須	エラー
1	変数宣言不要	警告表示
2	変数宣言不要	エラーと警告表示なし

## 2 配列の添字の範囲を設定

配列を宣言して変数を確保するとき、添字の範囲を決めることができる。

[使用例]

```
executer.array.origin=0
```

設定値は以下の通り。

設定値	配列 $X[n]$ を宣言した場合
0	$X[0], \dots, X[n]$ の要素を確保する (デフォルト値)
1	$X[0], \dots, X[n-1]$ の要素を確保する
2	$X[1], \dots, X[n]$ の要素を確保する

## 3 描画ウィンドウの原点の設定

描画ウィンドウで座標を指定するときの原点を設定する。

[使用例]

```
executer.graphic.origin=0
```

設定値は以下の通り。

0	左上を原点にする (デフォルト値)
1	左下を原点にする

## 4 入力支援ボタンの定義を別ファイルで行う

入力支援ボタンの設定を別ファイルに記述しておくことができる。定義ファイルまでのパスを設定値に記述しておく、PEN はその定義ファイルを用いて起動する。

通常は、このファイルを指定せずに PEN を起動することができる。

[使用例]

```
pen.button.path=./ButtonList.ini
```

## 5 フォントサイズを大きくして起動する設定

起動時のフォントサイズを大きくすることができる。

[使用例]

```
pen.teacher.flag=0
```

設定値は以下の通り。

0	通常の大きさ (デフォルト値)
1	フォントサイズを拡大して起動

図 1 は拡大したサイズ、図 2 は標準サイズでの表示。

## 6 デバックモードでの起動

デバックモードを利用することができる。デバック結果 (構文解析されたツリー) はコンソールに出力される。

[使用例] デバックモードでの起動方法

```
pen.debug.flag=1
```

と設定し、コマンドプロンプト上で、`java -jar PEN.jar` と入力し起動する。

設定値は以下の通り。

0	通常起動 (デフォルト値)
1	debug モードで起動

## 7 エラー時のダンプ動作設定

xDNCL の実行時にエラーが発生した場合、dump ファイルを出力させるように設定することができる。また、出力先の指定もできる。

[使用例]

```
pen.dump.flag=0
```

設定値は以下の通り。

0	dump ファイルを生成しない (デフォルト値)
1	dump ファイルを生成する

## 7.1 dump ファイルの暫定保存先を指定

dump ファイルの作成を指定した場合、ファイルの一時保存先を指定することができる。

出力先を設定しなければ HOME ディレクトリへ出力される。

(Windows の場合は、C:¥Document and Settings¥ ユーザー名)

出力先の指定方法は、

```
pen.dump.tempdir=一時保存先
```

## 7.2 一時保存した dump ファイルの最終保存先を指定

dump ファイルの作成を指定した場合、ファイルの最終保存先を指定することができる。

出力先を設定しなければ HOME ディレクトリへ出力される。

(Windows の場合は、C:¥Document and Settings¥ユーザー名)

出力されるファイル名は、「ユーザー名-コンピュータ名-数字.log」となる。

出力先の指定方法は、

```
pen.dump.destdir=ログファイル保存先
```

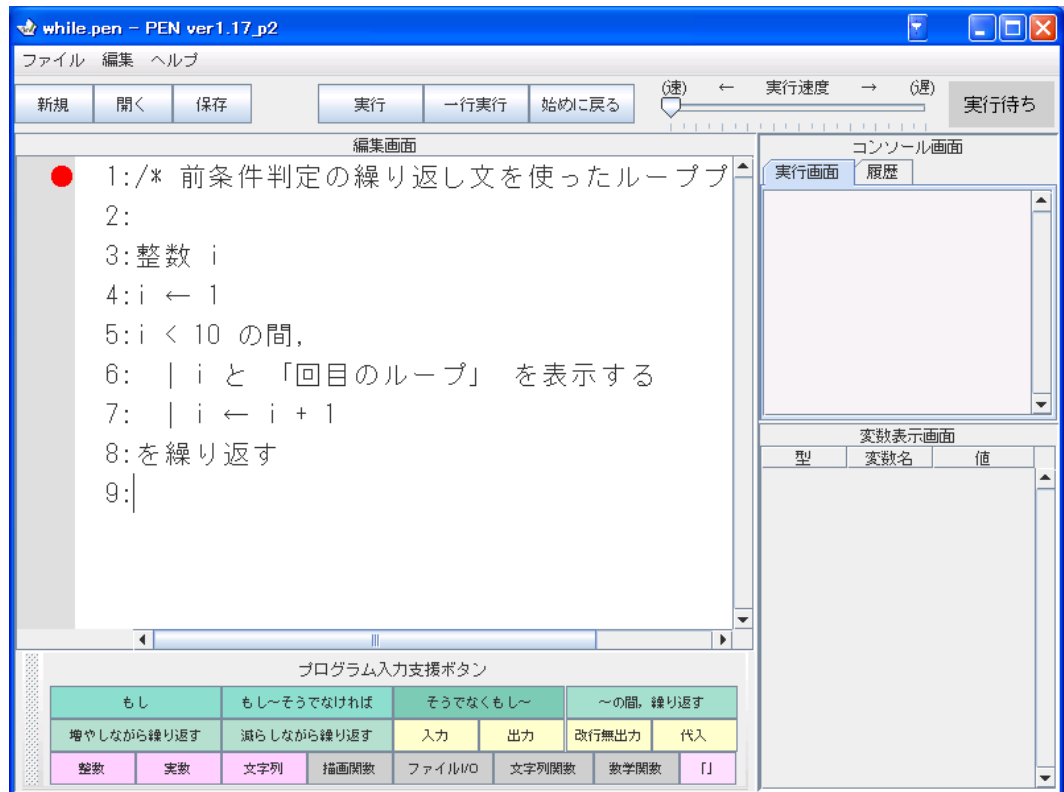


図 1 拡大サイズ

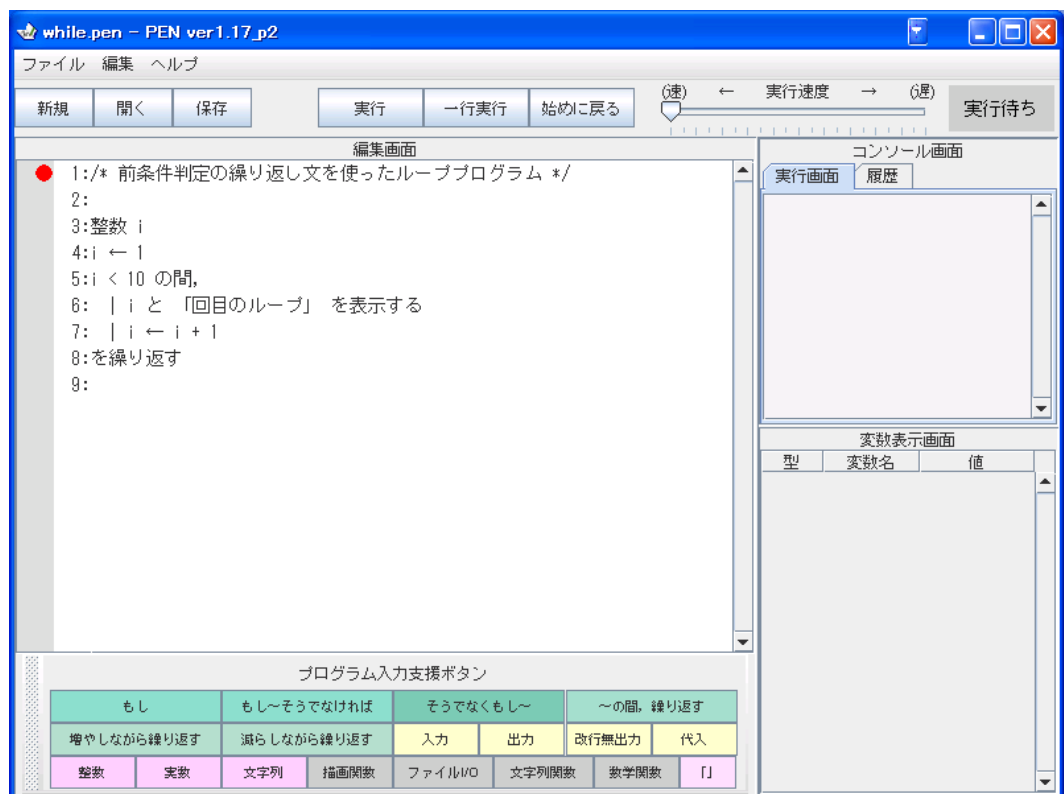


図 2 標準サイズ