

## ソフトウェア演習Ⅲ〔課題 4:クラス継承〕 青野雅樹

Java 言語でも類似のクラスの継承の課題を行った。ここでは、Python 言語で以下のプログラムを作成し、実行結果(kadai4.ps)とあわせ ZIP 等にまとめ、Moodle にアップせよ。締め切りは 11 月 3 日（火）が祝日のため 11 月 10 日（火）までとする。

- ① Shape（2 次元図形）クラスを**基底クラス**として作成せよ。
- ② Triangle（三角形）クラスを Shape クラスの**派生クラス**として作成せよ。
- ③ Trapezoid（台形）クラスを Shape クラスの**派生クラス**として作成せよ。ここで台形の上底と下底は X 軸と平行とし、 $x_1 < x_2$   $x_3 < x_4$  とする、詳細は次ページの図参照
- ④ Circle（円）クラスを Shape クラスの**派生クラス**として作成せよ。
- ⑤ \_\_main\_\_を含む kadai4.py を作成し、三角形と台形と円を合計 3N 個 ( $2 \leq N \leq 20$ ) (位置や大きさを) ランダムに発生させ、最後に、Shape クラスのプリント関数 (ps\_print 関数)ならびに area 関数を呼んで、発生させた図形の総面積を PostScript 内 (末尾) にプリントせよ。(注 : PostScript の出だしにもコメントで、氏名と学籍番号、日付を出力すること。)

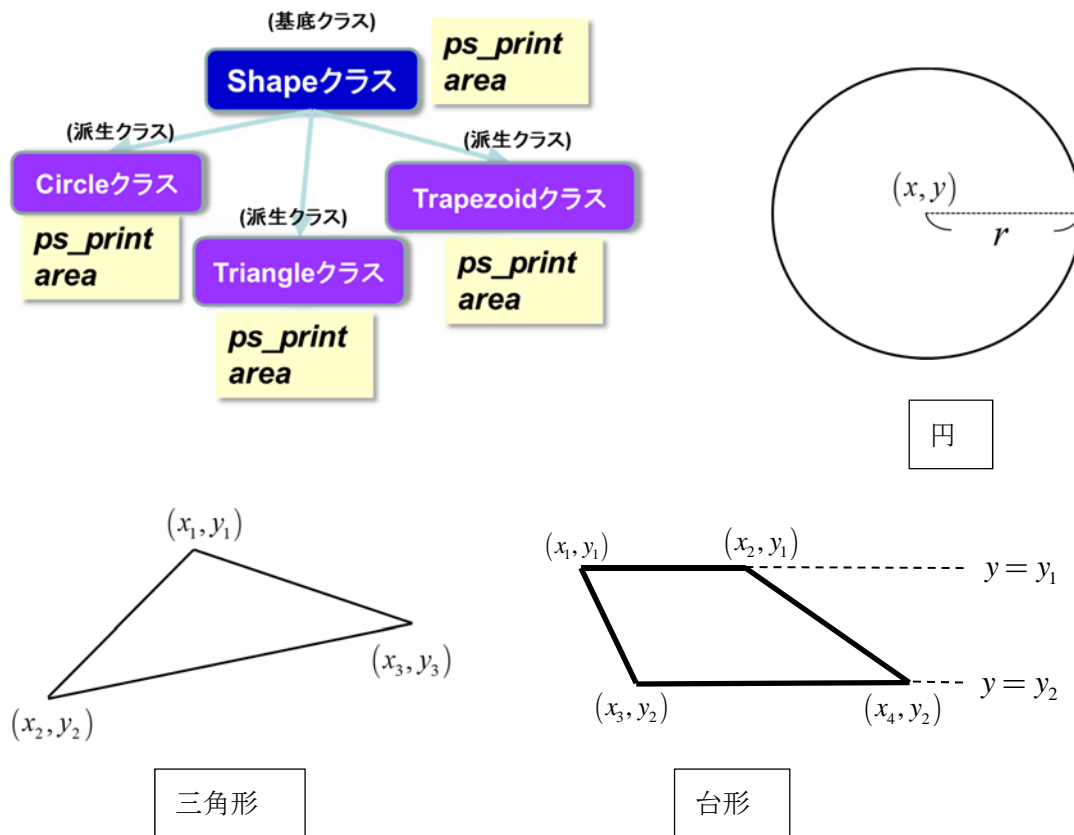
### 【コメントとヒント】

PostScript に発生する図形に関して、 $6 \leq 3N \leq 60$ ,  $0 \leq R,G,B \leq 1.0$ ,  $0 \leq x \leq 580.0$ ,  $0 \leq y \leq 700.0$  としてください。円の中心座標は(x,y)に準じてください。半径も x 値に準じてください。(XRANGE, YRANGE) =(580.0, 700.0)と定義してください。

課題の要点は、クラスの継承です。Shape クラスは**基底クラス（スーパークラス）**と呼ばれ、これを継承する Triangle クラス、Trapezoid クラス、Circle クラスは**派生クラス（サブクラス）**と呼ばれます。基底クラスで宣言された関数（area 関数）はオーバーライドされます。実際、面積計算は図形によって異なりますが、アクセスする場合は、area 関数を呼び出すと、自動的に派生クラスの area 関数を呼び出してくれます。たとえば、PostScript での円の出力は、以下のように x y r 0 360 arc の行 (stroke の直前の行) が円を定義しており、(x,y) は円の中心座標で r が半径を表します。

```
%%円
0.1 0.2 0.91 setrgbcolor
newpath
151.0 400.1 124.5 0 360 arc
stroke
```

三角形と台形の描画は、出だしの色と最後の stroke は円と同じで、違うのは、最初の頂点 (x,y) に x y moveto で移動し、以降、x y lineto で線分を結ぶことで行います。最後に closepath で図形を閉じてください。詳細は、後述のサンプルを参照してください。



三角形の符号付き面積は、 $Area = 0.5 * \sum_{i=1}^3 (x_i y_{i+1} - x_{i+1} y_i)$  で計算できます。ただし、 $x_4 = x_1$  で  $y_4 = y_1$  です。正負の値があり得ますので、絶対値をとるようにしてください。図のように左回りに頂点を定義したときに正となる量です。台形では、ここでの規則として  $x_1 < x_2$  ( $y = y_1$ ) とし、 $x_3 < x_4$  ( $y = y_2$ ) としてください。 $y_1$  と  $y_2$  の大小関係はどちらでも結構です。具体的なクラスは、以下のようにしてください。ここで列挙している変数等は、クラス内にあるべき最小要素なので、他のメンバー変数、メンバー関数などを定義して構いません。また、クラス内の関数の第一引数には、**self** が入りますが、説明では省略しています。実装では **self** を第一引数に付けてください。

基底である Shape クラス (クラス名=Shape)

メンバー変数:

変数名	値	概要
R	実数値	赤色成分
G	実数値	緑色成分
B	実数値	青色成分

コンストラクタ：

引数の数	引数の型	概要
3	(R, G, B)	R,G,B にセットする

メソッド（関数）：

メソッド名	引数型	戻り値型	概要
area	なし	実数値	<u>面積計算</u>
ps_print	なし	なし	色を PS で書き出す

派生クラス：円クラス（クラス名=Circle）

メンバー変数：

メンバー変数名	型	概要
x,y	2 つの実数値のタプル	円の中心座標 (x,y)
radius	実数値	円の半径（図では r）

コンストラクタ：

引数の数	引数の型	概要
6	(R,G,B,x,y,radius)	色, 中心座標, 半径

メソッド（関数）：

メソッド名	引数型	戻り値型	概要
area	なし	実数値	円の面積計算
ps_print	なし	なし	円のデータを PS 形式で書き出す

派生クラス：三角形クラス（クラス名=Triangle）

メンバー変数：

変数名	型	概要
x1,y1,x2,y2,x3,y3	実数値	3 頂点の座標値

コンストラクタ：

引数の数	引数の型	概要
9	(R,G,B,x1,y1,x2,y2,x3,y3)	色と 3 頂点

メソッド（関数）：

メソッド名	引数型	戻り値型	概要
area	なし	実数値	三角形の面積計算
ps_print	なし	なし	三角形のデータを PS 形式で書き出す

派生クラス：台形クラス（クラス名=Trapezoid）

メンバー変数:

変数名	型	概要
x1,y1,x2,y2,x3,y3,x4,y4	実数値	台形の 4 点の座標値

コンストラクタ:

引数の数	引数の型	概要
11	(R,G,B,x1,y1,x2,y2,x3,y3,x4,y4)	色と 4 頂点(x の条件に注意)

メソッド (関数):

メソッド名	引数型	戻り値型	概要
area	なし	実数値	台形の面積計算
ps_print	なし	なし	台形のデータを PS 形式で書き出す

メイン関数の処理手順


- (1) 引数 (図形の発生回数) のチェック ( $2 \leq N \leq 20$ )
- (2) 各種初期化 (乱数、総面積)
- (3) N 回ループ (ループ内で 3 種類の図形を発生させる) 色をランダムに発生
  - (ア) 円 (中心(x,y),半径 rad,  $x=[0,XRANGE]$ ,  $y=[0,YRANGE]$ ,  $rad=[0,0.25*XRANGE]$ )
  - (イ) 三角形の発生、座標値は円の中心と同様
  - (ウ) 台形の発生、x1,x2,x3,x4 の x は円の中心と同様だが、 $x1 < x2$ ,  $x3 < x4$  をチェック、y は y1 と y2 のみ円の中心座標の y と同様の範囲でランダムに発生。x の範囲も円の x の範囲と同様とする。
  - (エ) 上の 3 つの図形は (\_\_main\_\_ で定義する) shape\_list に append していく。
- (4) 3N 回、PS でプリント、この際、shape = shape\_list[i] から 2 つの関数 (area, ps\_print) にアクセスし、自動的に派生クラスのそれぞれの関数が呼び出せることを確認し、ファイルにアウトプット

Adobe の <https://www.adobe.com/content/dam/acom/en/devnet/actionscript/articles/psrefman.pdf>

に PostScript のマニュアルがあります。PostScript のチュートリアル例としては、

<https://www-cdf.fnal.gov/offline/PostScript/BLUEBOOK.PDF> にあります。図形のチュートリアルとしては、<http://paulbourke.net/dataformats/postscript/> がわかりやすいです。

【実行例】

\$ python kadai4.py 16  kadai4.ps

以下は、出力される PostScript ファイル (kadai4.ps) の例です (一部のみ)。

```
%%!PS-Adobe-2.0
%%File: kadai4.ps
%%*****
%%課題3：青野雅樹, 01162069
%%日付:2020年10月2日18時43分28秒
%%内容：クラスの継承を用いた2次元図形クラスの作成
%%*****
%%1番目の図形は円です
%% 色：
0.459513 0.850986 0.843484 setrgbcolor
%% 円：面積 = 12765.2
newpath
468.198 419.826 63.7439 0 360 arc
stroke
%%2番目の図形は三角形です
%% 色：
0.164469 0.133408 0.787166 setrgbcolor
%% 三角形：面積 = 80403.4
newpath
554.371 674.653 moveto
72.9878 644.728 lineto
560.019 340.953 lineto
554.371 674.653 lineto
closepath
stroke
%%3番目の図形は台形です
%% 色：
0.804806 0.180023 0.0960385 setrgbcolor
%% 台形：面積 = 26805.8
newpath
203.235 106.748 moveto
368.561 106.748 lineto
183.235 345.13 lineto
123.662 345.13 lineto
closepath
stroke

..... (3N個の図形)
```

%% 総面積は 931547です  
showpage

以下はPostScript（kadai4.ps）にps2pdfでPDF化した際の可視化例です。

