Python講座

2016/5/12(木)

今回の内容

- 比較演算子(引用)
- 条件式の詳細①
- if文
- while文
- 条件式の詳細②
- 関数・組み込み関数①
- 組み込み関数②
- ユーザー定義(自作)関数
- 自作関数(例)
- 自作関数(注意点)
- turtle
- turtleに関する文
- 演習問題

比較演算子

- 以降のページに出てくる条件式に対して使用する

Pythonでの書き方	数学での書き方	意味
a == b	a = b	aはbと等しい
a != b	a ≠ b	aはbと等しくない
a < b	a ＜ b	aはbより小さい
a > b	a ＞ b	aはbより大きい
a <= b	a ≦ b	aはbより小さいか等しい(以下)
a >= b	a ≧ b	aはbより大きいか等しい(以上)

条件式の詳細①

- 条件式から得られるものは、TrueかFalse
- 条件式が満たされている場合はTrue(真)
- 条件式が満たされていない場合はFalse(偽)
- 条件式から得られたものによって、if文やwhile文の処理の有無が変わる

if文

- 条件式が満たされる場合、内容を実行する
- 条件式が満たされない場合、内容を実行しない

↓

- 条件式がTrueである場合、内容を実行する
- 条件式がFalseである場合、内容を実行しない

while文

- 条件式が満たされる間、内容を実行する
- 条件式が満たされない間、内容を実行しない

↓

- 条件式がTrueである間、内容を実行する
- 条件式がFalseである間、内容を実行しない

条件式の詳細②

- 条件式の位置にTrue、Falseを直接書くことも可能
- if Ture:

  print(“A”)

  >>> A
- while Ture:

  print(“A”)

  >>> A A A A …
- Trueのため、必ず実行される

※while Trueを使う場合は、breakを書かなければ無限ループするので注意

関数・組み込み関数①

- プログラムを再使用するためのしくみ

- print()、input()、sum()なども関数
- 正確には、組み込み関数というもの

  ※Python自体に元々組み込まれている関数のこと

http://docs.python.jp/3/library/functions.html(3.5.1)

組み込み関数②

- 組み込み関数ではない関数の使用例
- 例：乱数
- import random
  →randomの関数を組み込む
- random.random()、random.randint(値a,値b)
  →randomの関数を使う

ユーザー定義(自作)関数

- 以下の作業を「関数を定義する」という
- def 関数名(変数1):
  内容
- def 関数名(変数1):
  内容
  return 変数2
- 関数名はわかりやすいものが好ましい
- 変数1は引数(ひきすう)と言い、内容で使用できる
- returnでこの関数で返却するもの(変数2)を決める

自作関数(例)

```
def plus(x,y):
    x+=y
    return x

a=10
b=5
print(plus(a,b))
>>>
15
>>>
```

- 関数名：plus
- 引数：x、y
- 内容：xを＋yする
- 返却：x

↓

- a、bを引数として
  a＋bの値を返却する関数

自作関数(注意点)

- 関数内にreturnは複数書けるが、返却できるものは1つのみ

  →if文などで指定する場合が多い
- 関数外で使用している変数を関数内では使用不可

  →使う場合は引数で関数内に引っ張ってくる
- 関数内で使用している変数を関数外では使用不可

  →使う場合はreturnで関数外に返却

  →逆に内外で変数名が同じでも問題なし

turtle

- グラフィックを書くことができる
- import turtle
  →turtleを組み込む
- t=turtle.Pen()
  →タートルオブジェクト作成
  （初めて作るときはウインドウも作成される）

turtleに関する文

- http://docs.python.jp/3.3/library/turtle.html

t.forward(x)	まっすぐx進む
t.left(x)	反時計回りにx度回転
t.right(x)	時計回りにx度回転
t.reset	白紙にする
t.circle(r,x)	半径rを反時計にx度
t.goto(x,y)	座標(x,y)に進む
t.up	描画をしない
t.down	描画をする

演習問題①

- テーマ：

  第一回の演習問題①「ノット30(仮)」の改良
- 改良点

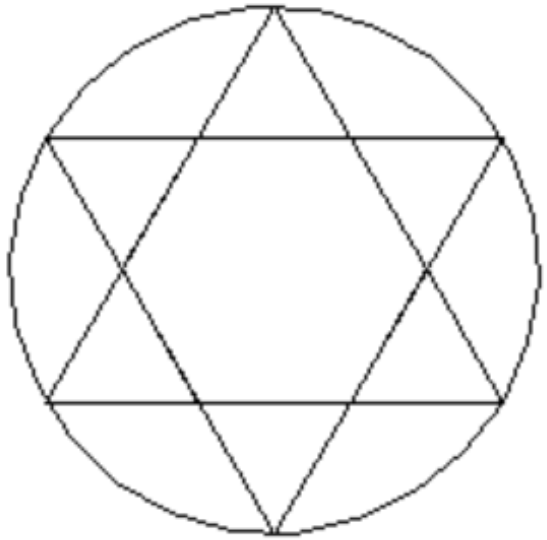
① Player vs CPU(1~8人) 対戦人数を選択

② Playerの順番を選択可能(またはランダム)

※Playerの勝敗のみ、CPUの誰が勝ったかは不問

演習問題②

- テーマ：六芒星の描画
- やること：turtleで図を描く



- 描画する際に使うもの
  import math
  math.sqrt(値)
- 値の平方根を返す