



python™

プログラミング勉強会

公式サイトから資料を
DLできます。
パスワード: pk_py1116
puzzleknot.wp.xdomain.jp



puzzleknot
Tohoku University Programming Club

補足: 条件式

- かつ
(条件式1) `and` (条件式2)
- または
(条件式1) `or` (条件式2)

補足: 条件式

- aが3で割り切れる かつ 5で割り切れる
`a%3 == 0 and a%5 == 0`

3

補足: 高度なスライス

- a[始点: 終点: スライスの増分]
- 例えばa[s:t:k]というスライスの場合

$$a_s, a_{s+k}, a_{s+2k}, \dots, a_{s+nk} (s + nk < t)$$

が順番に得られます

4

補足: 高度なスライス

- これを応用すると…
listの逆順を `a[::-1]` で取得できます

```
>>> a
[1, 2, 3, 4, 5, 6, 7, 8, 9]
>>> a[::-1]
[9, 8, 7, 6, 5, 4, 3, 2, 1]
```

5

補足: 高度なスライス

- スライスはオブジェクトとしては別物になります(もとのオブジェクトを変更しない)

```
>>> a[::-1]
[9, 8, 7, 6, 5, 4, 3, 2, 1]
>>> a
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

6

補足: range 関数

- listの中身として順番に並んだ数列が欲しい場合

```
a = [0, 1, 2, 3, 4]
for i in a:
```

の代わりに

```
for i in range(5):
```

と書けます

7

補足: range 関数

```
for i in range(始点, 終点):
```

たとえば

```
for i in range(1, 5):
```

と打つと, i には1,2,3,4 が順番に格納
されていきます
(最後の数が (終点)-1 であることに注意)

8

補足: range 関数

- 【高度な話です】
range はデータ型的一种です
(list, tupleと同じ順序付けされている
集合: シーケンス と呼ばれるもの)
- for文はシーケンスやdict型などの
オブジェクトを順番に取り出し、格納
していきます(for文で取り出すことが
できるオブジェクトを **iterable**な
オブジェクトといいます)

9

補足: range 関数

- 【高度な話です】
シーケンスもiterableであり、なおかつ
順序付けされているので取り出す順番
が一意です
- str型(文字列)もシーケンスの一種なの
で、for文で回せます
- シーケンスはa[1]やb[1:3]のように
インデックスやスライスが使えます

10

補足: range 関数

- 【高度な話です】
dict型やset型は順序付けされておらず、
for文で取り出したときに順序が一意に
定まりません
- これらはインデックス、スライスが使
えません

11

補足: range 関数

- 【高度な話です】
iterable(for文で順番に取り出せる)
 - シーケンス(順番が一意)
 - list
 - tuple
 - **range**
 - str(文字列)
 - set
 - dict

12

補足: range 関数

- 【高度な話です】
list, tupleは同名の関数を使うことでオブジェクトを生成できます
- 引数（カッコに与える値）にiterableなオブジェクトを与えることで、そのオブジェクトをlist / tuple に変換したものが得られます

13

補足: range 関数

- 【高度な話です】
そのため、単純な等差数列のlistやtupleを生成したい場合は、rangeオブジェクトをlistに変換すると一発でできます

```
# 初項5, 末項-28, 公差-3の等差数列
a = list(range(5, -29, -3))
print(a)
# 和も簡単に出せますね
print(sum(a))
```

```
output: [5, 2, -1, -4, -7, -10, -13, -16, -19, -22, -25, -28]
        -138
```

14

やってみよう 1の解答例

```
a = input() # 2人の出す手を受け取ります
b = input()

if(a == "G"): # aの出す手で場合分け
    if(b == "G"): # さらにbの出す手で場合分け
        print("DRAW!")
    elif(b == "C"):
        print("A WIN!")
    elif(b == "P"): # ここでelseを使うと,
        print("B WIN!") # G, C, P以外の入力はずべてPになってしまいます
elif(a == "C"):
    if(b == "G"):
        print("B WIN!")
    elif(b == "C"):
        print("DRAW!")
    elif(b == "P"):
        print("A WIN!")
elif(a == "P"):
    if(b == "G"):
        print("A WIN!")
    elif(b == "C"):
        print("B WIN!")
    elif(b == "P"):
        print("DRAW!")
```

やってみよう 2の解答例

```
for i in range(1, 101): # iはループで1から100まで動きます
    # 先に「かつ」の場合を調べる必要があります
    if(i % 3 == 0 and i % 5 == 0): # iが3と5で割り切れる
        print("Fizzbuzz")
    if(i % 3 == 0):
        print("Fizz")
    if(i % 5 == 0):
        print("buzz")
    else: # どれでも割り切れない
        print(i)
```