

# 発展的なプログラム 課題作成に使える いくつかの手法

23j1-210appendix

リクエストはTeamsのチャットで

# リストからランダムに取り出す

- `random.choice(リスト名)` を使う手もある

- サンプル

```
import random
te=['グー', 'チョキ', 'パー']
print(random.choice(te))
```

```
import random
te=['グー', 'チョキ', 'パー']
num=random.randint(0, 2)
print(te[num])
```

何を出したか数値で処理できる

# 入力した3桁の数をバラバラに

- 3桁の数を入力してもらい、1桁ずつ分割する方法
- for で分解しながら数値に直してリストに入れる
- サンプル

```
suu=input('3桁の数を入力して!')  
A=[]  
for i in range(3):  
    A.append(int(suu[i]))  
print(A)
```

# 3桁の数をランダムに作る1

- `random.randint()`

- サンプル

```
import random
A=random.randint(101, 999)
print(A)
```

# 3桁の数をランダムに作る2

- 各桁の数を別々に取り出せるように
- `random.randint()`

• サンプル

```
import random
A=[]
for i in range(3):
    if i==0:
        A.append(random.randint(1, 9))
    else:
        A.append(random.randint(0, 9))
print(A)
```

# 複数の条件

- and や or が使える

- サンプル

```
A=int(input('数を入力'))  
if A<10 or A>100:  
    print('2桁じゃないね')  
else:  
    print('2桁ですね')
```

# あるかないか

- あるか in を使って調べる

- サンプル

```
A=[1, 2, 3, 4, 5, 8, 9, 10]
B=int(input())
if B in A:
    print('すでにあります')
else:
    print('まだない数です')
```

# 場所の一致

- 2つのリストの重なる部分を調べる

- サンプル

```
A=[1, 2, 3, 4, 5, 8, 9, 10]
```

```
B=[1, 2, 3, 5, 6, 8, 8, 10]
```

```
for i in range(len(A)):
```

```
    if A[i]==B[i]:
```

```
        print(i, '番目の', A[i], 'は数と場所が一致')
```



# ちゃんと入力して

- 必ず1桁の数を入力させる
  - 終了するときには999と入力させる

## • サンプル

```
flg=False
while flg==False:
    suu=int(input('1桁の数を入力(終了は999)'))
    if suu==999:
        break
    elif suu>9:
        print('大きすぎます')
    elif suu<0:
        print('小さすぎます')
    else:
        print('入力ありがとう')
        flg=True
```