

■[学習 16-問題 2] すごろくのシミュレーション

以下の説明を読み、設問に答えよ。

次のコードは、40 ますあるすごろくをサイコロ 1 個を繰り返し振っていくつであがるかのシミュレーションを行う関数 `sugoroku1()` と、それを繰り返し 1000 回実行して、あがるまでの回数の分布をリストに記録する関数 `test()` を含んでいる。

```
import random

def sugoroku():
    place = 0
    count = 0
    while place < 40:
        place = place + random.randint(1, 6)
        count = count + 1
    return count

def test():
    a = [0]
    for x in range(1000):
        n = sugoroku()
        if len(a) <= n:
            # 配列に a[n]がまだなければ、
            a.extend([0 for x in range(n - len(a) + 1)]) # 必要な 0 の並びを追加
            a[n] = a[n] + 1
    print(a)

test()
```

上のプログラムを実行した出力の例を示す。

```
[0, 0, 0, 0, 0, 0, 0, 0, 0, 13, 52, 146, 223, 210, 188, 97, 44, 22, 4, 1]
```

シミュレーションを次のように修正したい。

- (a) 現在は 40 ます目に到達したら、サイコロがぴったりでなくてもあがれることになっている。ぴったりでない場合はあがれず、元の場所にとどまるように変更せよ。(sugoroku2)
- (b) 上と類似しているが、ぴったりでない場合は余ったぶんだけ出発点の方向に戻るように変更せよ。(sugoroku3)
- (c) すごろくには「ふりだし」がつきものである。10、20、30 のコマに止まったときは出発点に戻るように変更せよ。(sugoroku4)

それぞれに対応する関数 sugoroku2~sugoroku4 を選択肢群から行を選んで並び構成せよ。同じ選択肢を複数回用いてよい。

選択肢群:

- ア def sugoroku():
- イ return count
- ウ count = 0
- エ place = 0
- オ count = count + 1
- カ place = place + 1
- キ place = place + dice
- ク place = place + random.randint(1, 6)
- ケ place = 40 - (40 - place)
- コ dice = randint(1, 6)
- サ while place < 40:
- シ if place + dice > 40:
- ス if place > 40:
- セ if place + dice < 40:
- ソ if place < 40:
- タ if place % 10 == 0:
- チ if place % 10 != 0:
- ツ (インデントを 1 レベル戻す)

解答例 :

```
def sugoroku2():  
    place = 0  
    count = 0  
    while place < 40:  
        dice = random.randint(1, 6)  
        if place + dice <= 40:  
            place = place + dice  
            count = count + 1  
    return count
```

```
def sugoroku3():  
    place = 0  
    count = 0  
    while place < 40:  
        dice = random.randint(1, 6)  
        place = place + dice  
        if place > 40:  
            place = 40 - (place - 40)  
        count = count + 1  
    return count
```

```
def sugoroku4():  
    place = 0  
    count = 0  
    while place < 40:  
        place = place + random.randint(1, 6)  
        if place < 40:  
            if place % 10 == 0:  
                place = 0  
            count = count + 1  
    return count
```