

## エキスパート課題 A 「順次処理」

### 問1. 三角形の面積

「底辺が5、高さが8の三角形の面積を表示するプログラムを書いてください。」

<条件>

- ① 「底辺」、「高さ」、「三角形の面積」はそれぞれ**変数**を使います。
- ② **結果の表示**は、底辺と高さ、三角形の面積を、**変数**と**文字列**を使って表示します。

次のような出力にしてください。「底辺 5 高さ 8 の三角形の面積は 20.0」

<オプション>

- ・正しく表示されたら、底辺と高さの数値を変えて正しく表示されるか確かめましょう。

<アルゴリズム>

### 問2. テストの平均点

「テストの各教科の結果が「56,49,78」でした。これをリストにして、平均点を求めてください。」

<条件>

- ① 各教科の合計点は、**インデックス**番号を使って**配列(リスト)**の各要素を足し算してください。
- ② 教科の数は「リストの長さ」を求める**関数**を使って調べます。

リストの長さを調べるには **len 関数**を使います。使い方は、「len(リスト名)」です。

(使用例: color = [青,赤,黄,緑] print(len(color)) ⇒ 実行結果「4」と表示される)

- ③ **配列(リスト)**名、必要な**変数名**は各自で決めてください。

<アルゴリズム>

## エキスパート課題B 「繰り返し処理」

### 問3. 1~10の数字を表示

「**range 関数**を使って、「1 から 10」の数字を順番に表示してください。」

<オプション>

**range 関数**は複数の使い方ができます。それぞれを用いて問3のプログラムを作成してみてください。

<アルゴリズム>

### 問4. 掛け算九九

「掛け算九九のすべての答えを **for 文**を 2 回使って表示してください。」

結果イメージ:

1

2

3

…省略

72

81

<アルゴリズム>

## エキスパート課題 C 「条件分岐処理」

### 問5. 2つの数の比較

「変数 a と変数 b にそれぞれ整数を代入して比較し、等しい時は『2 つの値は等しいです』、異なる時は『a の方が大きいです』または『b の方が大きいです』と表示してください。」

<条件>

- ① 変数 a と変数 b を使います。
- ② 変数に値を代入する場合は、input 関数を使用してもよいですし、事前に数値を設定しておいてもよいです。

<オプション>

変数の値を変えて、正しい結果になるか試してみましょう。

<アルゴリズム>

### 問6. 出現率の異なるおみくじ

「おみくじプログラムを作りなさい。」

<条件>

0 から 8 の乱数を作成し、以下のように結果を振り分けて、結果を表示してください。

0 →大吉

上記以外の 2 以下 →吉

上記以外の 5 以下 →中吉

上記以外の 7 以下 →凶

それ以外 → 大凶

※random 関数を使用する。例えば 1~100 の乱数の発生プログラムは以下のようなになる。

int は整数型の変数を表す型

プログラム:

```
import random
kekka = random.randint(1,100)
print(kekka)
```

結果:23 ※結果はその都度異なる

※上記の結果は実行時にその都度変わります

<アルゴリズム>

## エキスパート課題 A 【順次処理】

### 問 1. 三角形の面積

正答例：

```
teihen = 5
```

```
takasa = 8
```

```
menseki = teihen * takasa / 2
```

```
print( '底辺', teihen , '高さ' , takasa , 'の三角形の面積は', menseki )
```

結果：

底辺 5 高さ 8 の三角形の面積は 20.0

### 問 2. テストの平均点

正答例 1：

```
tokuten =[56,49,78]
```

```
goukei = tokuten[0] + tokuten[1] + tokuten[2]
```

```
kyoukasu = len(tokuten)
```

```
heikin = goukei / kyoukasu
```

```
print(heikin)
```

正答例 2：

```
test_result = [56, 49, 78]
```

```
heikin = ( test_result[0] + test_result[1] + test_result[2] ) / len(test_result )
```

```
print( heikin )
```

結果：

61.0

## エキスパート課題 B 【繰り返し処理】

### 問3. 1～10の数字を表示

正答例 1 :

```
for i in range(10):  
    print(i + 1)
```

正答例 2 :

```
for i in range(1, 11):  
    print(i)
```

注 : range 関数には、start, stop, step の 3 つの引数がある。1 つしか指定しない場合は「stop」を指定したことになり、0 から始まり、stop の値は含まない範囲になることに注意する。

例 : range(5) → 0, 1, 2, 3, 4 ※結果的に()内の個数となる。

例 : range(5, 20, 5) → 5, 10, 15 ※20 は含まれない。

結果 :

```
1  
2  
… (省略)  
9  
10
```

### 問4. 掛け算九九

解答例 :

```
for i in range(1, 10):  
    for j in range(1, 10):  
        print(i * j)
```

結果 :

```
1  
2  
3  
…省略  
72  
81
```

## エキスパート課題 C 【条件分岐】

### 問5. 2つの数の比較

正答例 1 :

```
a = 5
b = 3
if a == b:
    print('2つの値は等しいです')
if a > b:
    print('aの方が大きいです')
if a < b:
    print('bの方が大きいです')
```

注：条件が複数 True になることはないので、elif を使わず、if 文だけでも書くことができます。

正答例 2 :

```
a = 6
b = 7
if a == b:
    print('2つの値は等しいです')
elif a > b:
    print('aの方が大きいです')
else:
    print('bの方が大きいです')
```

注：比較演算子の向きによって ( $a < b$  を  $b > a$  とするなど) 他にも書き方があります。

結果 :

(例) a に 5, b に 7 を代入した時  
bの方が大きいです

## 問6. 出現率の異なるおみくじ

正答例：

```
import random
kekka = random.randint(0,8)
if kekka ==0:
    print('大吉')
elif kekka<=2:
    print('吉')
elif kekka<=5:
    print('中吉')
elif kekka <=7:
    print('凶')
else:
    print('大凶')
```

結果：

乱数の値が

0 →大吉

1,2 →吉

3,4,5 →中吉

6,7 →凶

8 →大凶

となるので、9個の乱数を用いて、大吉と大凶は  $1/9$  と出にくく、中吉と凶は2倍、吉は3倍出やすくしている。