

4. Function (함수) - Answer

Initial Commit: 2020.08.07

Private lesson (OOP Basics with Python, August 2020)

본 답안들은 제시된 문제를 푸는 방법 중 하나일 뿐입니다.

(1-1) 정수를 입력받아 n 변수에 저장하고, 파라미터 x에 대해 x번만큼 Hello, World!를 출력하는 함수 rep를 정의하여 rep(n)을 호출하세요. rep의 return값은 없습니다.

```
def rep(x):  
    for i in range(0, x):  
        print('Hello, World!')  
  
n = int(input('Insert a number: '))  
rep(n)
```

(1-2) 1-1번 문제에 대한 추가 도전 과제입니다. rep 함수의 파라미터를 x가 아닌 n으로 바꾸어 실행하고, 그 결과를 1-1번의 결과와 비교해보세요.

```
def rep(n):  
    for i in range(0, n):  
        print('Hello, World!')  
  
n = int(input('Insert a number: '))  
rep(n)
```

(2) 정수를 입력받아 a 변수에 저장하고, 파라미터의 세제곱을 return하는 함수 f를 정의하여 f(a)를 호출한 결과를 출력하세요.

```
def f(x):  
    return x*x*x  
  
a = int(input('Insert a number: '))  
print(f(a))
```

(3) 국어, 수학, 영어 점수를 정수로 입력받아 각각 korean, math, english 변수에 저장하고, 점수의 평균을 return하는 함수 avg와 해당 점수의 등급을 return하는 함수 cal을 정의하여, 세 점수에 의한 등급을 출력하세요.

(평균이 90점 이상이면 A등급, 80점 이상이면 B등급, 70점 이상이면 C등급, 그 외의 경우 D등급입니다.)

```
def avg(x, y, z):
    return (x + y + z) / 3

def cal(n):
    if n >= 90:
        return 'A'
    elif n >= 80:
        return 'B'
    elif n >= 70:
        return 'C'
    else:
        return 'D'

korean = int(input('Korean score: '))
math = int(input('Math score: '))
english = int(input('English score: '))
print(cal(avg(korean, math, english)))
```

(4) 두 정수를 입력받아 각각 a, b 변수에 저장하고, 두 파라미터의 최대공약수를 return하는 함수 g를 정의하여 g(a, b)를 호출한 결과를 출력하세요.

```
def g(x, y):
    res = 1
    if x > y:
        c = x
        x = y
        y = c
    for i in range(2, x + 1):
        if x % i == 0 and y % i == 0:
            res = i
    return res

a = int(input('Insert number a: '))
b = int(input('Insert number b: '))
print(g(a, b))
```

(5-1) 양의 정수 n에 대한 팩토리얼은 n보다 작거나 같은 모든 양의 정수의 곱으로 정의되며, ! (느낌표)로 표시합니다.

(Ex. $6! = 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 720$)

정수를 입력받아 n 변수에 저장하고, 파라미터의 팩토리얼을 return하는 facto 함수를 정의하여 facto(n)을 호출한 결과를 출력하세요.

```
def facto(x):
    res = 1
    for i in range(2, x+1):
        res = res * i
    return res

n = int(input('Insert a number: '))
print(facto(n))
```

(5-2) 5-1번 문제에 대한 추가 도전 과제입니다. facto 함수를 **재귀함수**로 정의하여 동일한 결과를 얻으세요.

```
def facto(x):
    if x <= 1:
        return 1
    return x * facto(x-1)

n = int(input('Insert a number: '))
print(facto(n))
```

(6-1) 피보나치 수열은 일반적으로 1, 2, 3, 5, 8, 13... 과 같이 n번째 원소가 (n-1)번째 원소와 (n-2)번째 원소의 합인 수열으로 정의됩니다.

정수를 입력받아 n 변수에 저장하고, 피보나치 수열의 (파라미터)번째 원소를 return하는 fibo 함수를 **재귀함수**로 정의하여 fibo(n)을 호출한 결과를 출력하세요.

```
def fibo(n):
    if n <= 1:
        return 1
    if n == 2:
        return 2
    return fibo(n-1) + fibo(n-2)

n = int(input('Insert a number: '))
print(fibo(n))
```

(6-2) 6-1번 문제에 대한 추가 도전 과제입니다. 재귀함수로 정의된 fibo 함수는 파라미터가 큰 값 (50 이상)일 때 처리 시간이 유의미하게 증가합니다. fibo 함수의 시간 복잡도를 계산해보고, 처리 시간을 감소시킬 수 있는 아이디어를 생각해 보세요.

- (Answer) 시간 복잡도: $O(2^{(n/2)})$, Memoization으로 처리 시간 감소

(7) 양의 정수를 입력받아 n 변수에 저장하고, n 에 대한 다음 문제의 답을 return하는 함수 stair을 정의하여 호출한 결과를 출력하세요.

" n 개의 칸을 가지는 계단이 있습니다. 당신은 계단을 한 번에 1칸 또는 2칸씩 오를 수 있습니다. 계단을 모두 오르는 경우의 수를 구하세요."

```
def stair(x):
    if x <= 1:
        return 1
    elif x == 2:
        return 2
    return stair(x-1) + stair(x-2)

n = int(input('Insert a number: '))
print(stair(n))
```

(Ex) 두 정수를 입력받아 변수 n, x 에 저장하고, x 에 대해 $\sin x$ 의 n 차 테일러 다항식을 계산한 값을 return하는 함수 sine을 **재귀함수**로 정의하여 $\sin(n, x)$ 를 호출한 결과를 출력하세요.

```
def power(a, b): # Calculate a^b
    res = 1
    for i in range(0, b):
        res = res * a
    return res

def facto(x): # Calculate x!
    if x <= 1:
        return 1
    return x * facto(x-1)

def sine(n, x):
    if n == 0:
        return x
    if n % 2 == 0:
        return sine(n-1, x) + power(x, 2*n+1) / facto(2*n+1)
    return sine(n-1, x) - power(x, 2*n+1) / facto(2*n+1)

n = int(input('Determine the degree of sin calculation (n): '))
```

```
x = float(input('Insert a number (x): '))  
print(sine(n-1, x))
```