

# PE01: PE of 17/10/2019 (solutions)

---

Master in Informatics and Computing Engineering  
Programming Fundamentals  
Instance: 2019/2020

*An example of solutions for the 5 questions in this **Practical on computer evaluation**.*

## 1. Reverse integers

Write a Python program that, given an integer `num` by user input, computes its reverse (the number with the digits by the reverse order) and prints it.

You are not allowed to use string manipulation (for example string concatenation).

For example:

- for `num=766`, the output is `667`
- for `num=789`, the output is `987`
- for `num=45654`, the output is `45654`

Solution:

```
num = int(input("num? "))

tmp = num
rev = 0    # the reverse
while tmp > 0:
    d = tmp % 10
    rev = rev * 10 + d
    tmp = tmp // 10

print(rev)
```

## 2. The sum of the squares

Write a Python program that, given an integer with one digit **d** and another integer **num**, both provided by user input in that order, prints the sum of the squares of the digits of **num** greater than **d**.

For example:

- for **d=2** and **num=135**, the output is **34** (because of  $3*3+5*5$ )
- for **d=3** and **num=135**, the output is **25** (because of  $5*5$ )
- for **d=5** and **num=135**, the output is **0**

Solution:

```
d = int(input("Enter a digit: "))
num = int(input("Enter a number: "))

sum = 0
aux = num
while aux > 0:
    digit = aux % 10
    if digit > d:
        sum += digit * digit
    aux //= 10

print(sum)
```

### 3. Triangle Quest

Write a Python program that for an integer  $n$  (1-9), given by user input, prints a numerical triangle of height  $n-1$  like the one below:

```
1
22
333
4444
55555
...
```

For example:

- for  $n=1$ , the output is ""
- for  $n=3$ , the output is "1\n22"
- for  $n=4$ , the output is "1\n22\n333"
- for  $n=6$ , the output is "1\n22\n333\n4444\n55555"

Solution:

```
n = int(input("n? "))

for i in range(1, n):
    s = ''
    for j in range(i):
        s += str(i)
    print(s)
```

## 4. Triathlon

In a triathlon competition, there are 3 stages: 1.5 km of swimming, 40 km of cycling and, finally, 10 km of running. Each participant must complete all three under 4 hours and must have a minimum velocity of 2 km/h in the swimming stage, 20 km/h in cycling and 8 km/h in running.

Write a Python program that, given three times of completion **tS**, **tC** and **tR** (in hours; one for each stage) by user input, in this order, checks if the participant met all the requirements. If so, it should print the total time. Otherwise, it should print the first factor that caused the disqualification ("Time", "Swimming", "Cycling" or "Running", in this order).

For example:

- for **tS=0.4**, **tC=1.2**, **tR=0.4**, the output is: **2.0** (the total time)
- for **tS=1**, **tC=1**, **tR=4**, the output is: **Time**
- for **tS=0.5**, **tC=1**, **tR=2.2**, the output is: **Running**

Solution:

```
tS = float(input("Swimming stage time: "))
tC = float(input("Cycling stage time: "))
tR = float(input("Running stage time: "))

vS = 1.5 / tS
vC = 40 / tC
vR = 10 / tR

totalTime = tS + tC + tR

if totalTime > 4:
    print("Time")
elif vS < 2:
    print("Swimming")
elif vC < 20:
    print("Cycling")
elif vR < 8:
    print("Running")
else:
    print(totalTime)
```

## 5. Quaternary to decimal converter

Decimal numbers are base10 numbers that use only digits from 0 to 9, inclusive.

Quaternary numbers are base 4 numbers that use only digits 0, 1, 2 and 3.

Write a Python program that converts a quaternary number (base 4) **quat**, given by user input, into the corresponding decimal number (base 10).

For example:

- for quat=123, the output is the decimal number 27
- for quat=112233, the output is the decimal number 1455
- for quat=11, the output is the decimal number 5

Solution:

```
quat = int(input("Quaternary number = "))

dec = 0      # accumulator
power = 0    # digit position

while quat:
    last = quat % 10 # last digit
    dec += last * 4 ** power
    quat //= 10      # remove last digit
    power += 1       # increment position

print(dec)
```

**The end.**

*FPRO, 2019/20*