PE1: PE of 26-10-2018 (Solutions)

Master in Informatics and Computing Engineering
Programming Fundamentals
Instance: 2018/2019

Here you have a possible solution for each question of the Practical on computer Evaluation.

1. Check Armstrong number

Write a Python program that checks if a number **`num`** with 3 digits, given by user input, is an Armstrong number or not. In an Armstrong number of 3 digits, the sum of the cubes of each digit is equal to the number itself.

Use Spyder3 to create a new file named **`question1.py`** in your folder named **`PE1`**.

For example:

- for num=`153`, the output is: `True`
- for num=`234`, the output is: `False`
-

Save your program in the file **`question1.py`** inside the folder **`PE1`**.

Solution:

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

# initialize sum
sum = 0

# find the sum of the cube of each digit
aux = num
while aux > 0:
    digit = aux % 10
    sum += digit ** 3
    aux //= 10

if num == sum:
    print(True)   # num,"is an Armstrong number"
else:
    print(False)  # num,"is not an Armstrong number"
```

2. The sum of the double

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

For example:

- for d=`3` and num=`135`, the output is `10` (because of 2*5)
- for d=`2` and num=`135`, the output is `16` (because of 2*3+2*5)
- for d=`3` and num=`102`, the output is `0`
- for d=`2` and num=`12345`, the output is `24`

Save your program in the file `question2.py` inside the folder `PE1`.

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 += 2 * digit
    aux //= 10

print(sum)
```

3. Ages

Write a Python program that has two lists of equal size referenced by variables `names` (a list of strings) and `ages` (a list of integers), with values of your choice. The program prints all pairs name-age where name is from list `names` and age is from list `age` at the same position.

For example:

- for `names = ["bart", "marie", "jo"]` and `ages = [23, 75, 19]`, the output is `bart-23 marie-75 jo-19`
- for `names = ["mary", "john"]` and `ages = [13, 95]`, the output is `mary-13 john-95`

Save your program in the file `question3.py` inside the folder `PE1`.

Solution:

```
names = ["bart", "marie", "jo"]
ages = [23, 75, 19]

result = ""
i = 0
for name in names:
    j = 0
    for age in ages:
        if i == j:
            result += name + "-" + str(age) + " "
        j += 1
    i += 1

print(result)
```

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`**

Save your program in the file **`question4.py`** inside the folder **`PE1`**.

Solution:

```python
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. Octal converter

Write a Python program that converts a decimal number (base 10) **`dec`**, given by user input, into an octal number (base 8). Decimal numbers are base 10 numbers and use only digits from 0 to 9, inclusive. Octal numbers can use digits from 0 to 7, inclusive.

For example:

- for dec=`9`, the output is the octal number **`11`**
- for dec=`64`, the output is the octal number **`100`**
- for dec=`23456`, the output is the octal number **`55640`**

Save your program in the file **`question5.py`** inside the folder **`PE1`**.

Solution:

```
dec = int(input("Decimal number = "))

octal = 0
aux = dec
place = 0
while aux:
    octal = aux % 8 * 10 ** place + octal
    aux //= 8
    place += 1

print(str(octal))
```

The end.

FPRO, 2018/19