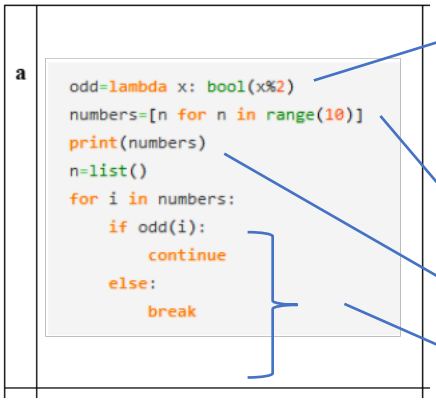
Assignment 4 – Script Language (Python)
Student: Bruno Lopes 2234087

Question1

a



```
odd=lambda x: bool(x%2)
numbers=[n for n in range(10)]
print(numbers)
n=list()
for i in numbers:
    if odd(i):
        continue
    else:
        break
```

This lamda will return False if x is even (x%2=0) or True if x is odd (x%2 !=0), because 0 is associate with False and 1 or any other number is associate with True.

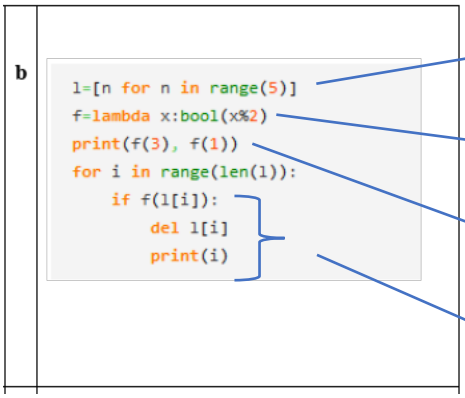
Variable numbers will be an array of numbers from 0 to 9

Print numbers 0 to 9 of variable numbers

This loop goes through the numbers, if odd(i) is True (it means “i” is odd) the loop continues, if “i” is even we stop the loop, but there is no printing in the loop.

==========

b



```
l=[n for n in range(5)]
f=lambda x:bool(x%2)
print(f(3), f(1))
for i in range(len(l)):
    if f(l[i]):
        del l[i]
        print(i)
```

Array with numbers from 0 to 4

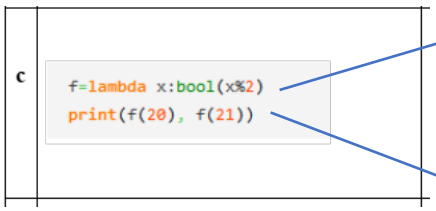
Function lamda will return False if x is even (x%2=0) or True if x is odd (x%2 !=0).

Print (True, True)

If f(l[i]) is True (i is odd), delete the value l[i] from the array l and after print the index i, this is cause an error, this function is changing the size of the array l, so len(l) will change in the middle of the loop, causing an error.

==========

c



```
f=lambda x:bool(x%2)
print(f(20), f(21))
```

This lamda will return False if x is even (x%2=0) or True if x is odd (x%2 !=0), because 0 is associate with False and 1 or any other number is associate with True.

False, True

==========

d

```
import functools
l=[1,2,3,4]
print(functools.reduce(lambda x,y:x*y,l))
```

Import the library functools

Create the array of numbers

This function reduce will reduce the array to one number, it works in pairs of numbers (x,y) doing the operation indicated (x*y) in the array l, always the first number with the next. So the operation will follow as:
1*2=2, 2*3=6, 6*4=24,

The return will be 24.

==========

e

```
l=[1, -2, -3, 4, 5]
def f1(x):
    return x<2
m1=filter(f1, l)
print(list(m1))
```

Array l with numbers: 1,-2,-3,4 and 5

Function that receive x and return True if x<2 or False if it is not.

The filter(f1,l) will filter the numbers in l and return only the numbers where f1 returns True, and store the result in m1.

As the return will be of type filter we need to cast to type list and then we can print the array m1.

==========

f

```
l=[1, -2, -3, 4, 5]
def f1(x):
    return x<-1
m1=map(f1, l)
print(list(m1))
```

Array l with numbers: 1,-2,-3,4 and 5

Function that receive x and return True if x<-1 or False if it is not.

The map(f1,l) will return the result of f1 (True or False) for each number of l

As the return will be of type map we need to cast to type list and then we can print the array m1.

==========

g

```
import functools
l=[1, 2, 3, 4, 5]
m=functools.reduce(lambda x, y:x if x>y else y, l)
print(m)
```

Import the library functools

Create the array of numbers called l

This function reduce will reduce the array to one number, it works in pairs of numbers (x,y) doing the operation indicated (x if x>y else y) in the array l, always the first number with the next. So the operation will follow as:
1>2? No, return 2, 2>3? No, return 3, 3>4? No, return 4, 4>5? No, return 5.

The return will be 5.

==========

h

```
m=reduce(lambda x: x-3 in range(4, 10))
print(list(m))
```

This example will not work because to use reduce we need import functools first and the function reduce needs 2 arguments and this example only use x.

==========

Question2

```
l=[1, 2, 3, 4, 5]
def f1(x):
    return x<0
m1=filter(f1, l)
print(list(m1))
```

the one line equivalent could be:

print(list(filter(lambda x: x<0, l)))

==========

Question3

```
def sf(a):
    return a%3!=0 and a%5!=0
m=filter(sf, range(1, 31))
print(list(m))
```

The output will be all numbers between 1 and 30 that are not multiple of 3 neither multiple of 5, so the output will be:
[1, 2, 4, 7, 8, 11, 13, 14, 16, 17, 19, 22, 23, 26, 28, 29]

References: Python Documentation - https://docs.python.org/3/library