

Language Python

QFRM1

Zaouche-Dahmani Djaouida

2018-2019

Python is a very simple language, and has a very straightforward syntax. There are two major Python versions, Python 2 and Python 3. Python 2 and 3 are quite different. This course uses Python 3, because it more semantically correct and supports newer features.

1. Print the message "Hello word!!!".
2. (Variables) Python is completely object oriented, and not "statically typed". You do not need to declare variables before using them, or declare their type. Every variable in Python is an object. This course will go over a few basic types of variable.
 - Python supports two types of numbers - integers and floating point number. Execute and explain:

```
x=8
print(x)
y=8.9
print(y)
z=float(5)
print(z)
```

- Python also supports complex numbers. Execute and explain:

```
import cmath
z1=complex(1,2)
z2=complex(3,4)
print(z1)
print(z1+z2)
print(z1*z2)
print(z1.imag)
print(z1.real)
```

```
print(z1/z2)
print(z1**2)
print(z1*z1)
print(z1.conjugate())
```

3. (String) Python supports strings.
 - Strings are defined either with a single quote or a double quotes.

```
mystring = 'hello'
print(mystring)
mystring = "hello"
print(mystring)
```

- Get the character at position 1 (remember that the first character has the position 0).

```
print(mystring[1])
```

- Get the characters (substring) from position 1 to 3 (not included)
print(mystring[1:3])
- The length of a string is given by function len
print(len(mystring))
- The split() method splits the string into substrings if it finds instances of the separator:
- a = "Hello, World!"
print(a.split(","))
- Command-line String Input

```
print("Enter your name:")
x = input()
print(type(x))
print("Hello, ",x)
```

4. (Instruction if)

- Execute and observe:

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
```

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
    print("b is not equal to a")
```

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
    print("b is not equal to a")
print("Outside the if statement")
```

```
a = 33
b = 33
if b > a:
    print("b is greater than a")
elif a == b:
    print("a and b are equal")
print("Outside the if statement")
```

```
a = 200
b = 33
if b > a:
    print("b is greater than a")
elif a == b:
    print("a and b are equal")
else:
    print("a is greater than b")
print("Outside the if statement")
```

```
if a > b and c > a:
    print("Both conditions are True")
if a > b or a > c:
    print("At least one of the
    conditions is True")
```

- b. Change the values of a and b in order to modify the behavior of each previous if statement. Add print instructions for then and else instruction for each if statement.
- c. Write a program which assigns 3 to x, x+1 to y, y*9 to x and finally x+9 to x, then it displays x and y.
- d. Write a program python which displays the max of two variables x and y.
- e. Write a program python which displays the min of three variables x, y and z.
- f. Write a program python which makes the circular assignments



- g. Write a program which solves the equation $ax+b=c$, with a, b, c are float constant.
- h. Write a program python which solves the equation $ax^2+bx+c=0$, with a, b and c are float constants and a is not zero.
- i. Write a program which computes the derivative of a polynomial:

$$a_1x^{\alpha_1} + \dots + a_{n-1}x + a_n$$

5. (Loop, for) Execute and explain:

```
for x in range(6):
    print(x)

for x in range(6,12):
    print(x)

for x in range(2, 30, 3):
    print(x)
```

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)

fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
    if x == "banana":
        break

print("Outside of for")
```

6. (Loop, while) Execute and explain:

```
i = 1
while i < 6:
    print(i)
    i += 1

i = 1
while i < 6:
    print(i)
    if i == 3:
```

- a. Write a program which computes the factorial of a number n.
 - b. Write a program which: (1) reads a number n, (2) then asks the user to introduce n numbers, (3) finally displays the min, max and average of the n numbers.
 - c. Write a program which accepts an integer, then it displays the binary and hexadecimal representation. Use 'B' and 'H' at the end of binary and hexadecimal representations, respectively. For instance, 17 is represented by 1001B and 11H.
 - d. Write a program which accepts a hexadecimal representation and returns its decimal value.
7. (Array) Arrays are used to store multiple values in one single variable. Execute and explain:

```
X=['book','cd','pen','radio','car','bus']
for i in range(len(X)):
    print(X[i])
print('_____')
for x in X:
    print(x)
```

```
X=['book','cd','pen','radio','car','bus']
X.append('Djo')
X.pop(1)
X.remove('car')
X
```

- a. Write a program which computes the max, min and average of an integer array X.
- b. Write a program which build an array XX which is the reverse of a second array X.
- c. Write a program which splits an array X into two arrays X1 and X2 such that X1 contains the odd values of X and X2 the even values.
- d. Write a program which adds two arrays.
- e. Write a program which sorts an array X. Choose the selection algorithm.
- f. Define a matrix 4x3, $m = [[1,2,4],[5,6,7],[8,9,10],[11,12,14]]$ and then display the matrix.
- g. Write a program which returns the sum of a matrix's elements.
- h. Write a program which adds two matrixes.
- i. Write a program which assigns zero to the first element of each row of a matrix m.
- j. Write a program which does the multiplication of two matrixes m1 and m2.

6. (Function) A function is a block of code which only runs when it is called. You can pass data, known as parameters, into a function. A function can return data as a result.

- a. Call the following function:

```
def my_function():  
    print("Hello from a function")
```

- b. Call three times the following function:

```
def my_function(fname):  
    print(fname + " Refsnes")
```

- c. Call three times the following function:

```
def my_function(x):  
    return 5 * x
```

- d. Write a function prime which has one parameter n and displays a message which indicates if the parameter is prime or not

- e. Write a function which accepts two parameters a and b and computes the greatest common divisor of a and b.

- f. Write a function which accepts an array y and checks if y's elements are probabilities and follow a normal distribution.

- g. Write a function which accepts a matrix X nxn and computes its determinant.

- h. Write a function which accepts a matrix and returns the reversed matrix.