ITI 1120
Labo # 1

Introduction to Python

1

1

Laboratory Objectives

Exercises in Python with:

- Variables
- Assignments
- Computing
- Functions

2

2

Variables

Variables are used to store values, for instance to avoid recalculating values.

Assign them names that will help identify them and clarify the code.

Valid variable names are composed of letters, numbers, underscored characters (_)

- Start with a letter or _
- Names with upper case letters are differents from those wth lower cases
- Valid names: ninja, Ninja, n_i_n_j_a
- Invalid names: 1337, 1337ninja

Python Convention : join words with _

- Valid names elite_ninja, leet_ninja, ninja_1337
- Invalid names 1337_ninja

3

3

Assignment

The assignment operator = (equal) is used to provide a value to a variable.

Note: two equal == is used to test equality

Example

```
my_name = "Grace Hopper"
print(my_name)
my_age = 25
print(my_age)
# Anniversary : add one
my_age = my_age + 1
print(my_age)
```

4

4

Exercise 1

Read two integer values from the keyboard (using input and convert it to int) then display the result of the integer division (using //) and the remainder / modulo (using %).

5

5

Computing in Python

```
# Examples: Temperature
# Transform Fahrenheit in Celsius
# c = 5 / 9 * (f - 32)
# use good variable names!

temp_Fahrenheit = 212
temp_Celsius = 5.0 / 9.0 * (temp_Fahrenheit - 32)

print(temp_Celsius)

# test it! 32 Fahrenheit is 0 Celsius
# 212 Fahrenheit is a 100 Celsius
```

6

6

Simples Functions: example

```
def fahrenheit_in_celsius(temp_Fahrenheit):
  "Convert temperature from Farenheit to Celsius"
  # temp_celsius is a local variable,
  # it exists in the body of the function
  temp_Celsius = 5.0 / 9.0 * (temp_Fahrenheit - 32)
  return temp_Celsius

# t_fahrenheit and t_celsius are global variables

t_fahrenheit = 212
t_celsius = fahrenheit_in_celsius(t_fahrenheit)
print(t_fahrenheit, "Fahrenheit is", t_celsius,  "Celsius.")
```

7

7

Exercise 2

Convert the temperature in the other direction, from Celsius to Fahrenheit, using a function.

Call the function several times and display the results.

8

8

Exercise 3

- Implement in Python the algorithm that computes the finale grade. Use a function. Call it.

DATA:
    hw_Average, midterm, final *(three numbers >= zero)*
RESULTS:
     total   *(final note)*
HEADER:
   total ← compute(hw_Average, midterm, final)
MODULE:
   total ← hw_Average *25/100 + midterm*25/100 +
           final*50/100

9

9

Exercise 4

Implement in Python the algorithm that computes the area of a triangle. Use a function.

DATA: side1, side2, side3
RESULTS: area
HEADER:
     area ← computeArea(side1, side2, side3)
HYPOTHESIS: side2, side2 and side3 are >0
MODULE:
   p ← side1 + side2 + side3
   area <- sqrt(p * (p – 2*side1) * (p – 2*side2) *
          (p – 2*side3))/4

10

10