Série 2 – Programmation Scientifique

# Exercice 1

1. Richtig
2. Falsch
3. Falsch
4. Falsch
5. Richtig
6. Falsch
7. Falsch, richtig wäre z.B. "x ← 5+1" bzw. "x ← 6"
8. Richtig
9. Falsch, der Operator dafür wäre "\*\*". Z.B. 2^4 ≙ 2\*\*4

# Exercice 2

1. A = -15 ; B = 41
2. A = 1, B = 1, C= 1
3. Nein, das stimmt so nicht. Angenommen, man definiert zuvor A ←1 und B ← 2.  
   B ← A bedeutet nun also, dass B den Wert von A annimmt, d.h. B = 1. (Nun haben also beide Variablen den Wert von A). Als nächstes betrachten wir die Anweisung A ← B. A nimmt jetzt also den neuen Wert von B an, d.h. A = 1. Somit wird ersichtlich, dass die Werte der Variablen A und B nicht vertauscht werden. Grund dafür ist, dass die Werte sich "über Zeit" ändern und nicht konstant bleiben.
4. C ← A, A ← B, B ← C

# Exercice 3

1. Der Unterschied zwischen / und // ist folgender: Der Operator / ist als Float-Division bekannt. Wird dieser Operator auf zwei Operanden angewandt, so ist der Quotient eine Fliesskommazahl. Der Operator // hingegen gibt einen geglätteten Wert zurück, also das selbe Resultat wie / aber gerundet zu einer Ganzzahl. Bsp: 10/3 = 3.3333(…), 10//3 = 3
2. 1. "!=" Der Ungleichheitsoperator prüft, ob seine Operanden nicht gleich sind. Falls dies stimmt gibt er das Resultat True, ansonsten False. Bsp: 1!=1 → False; 1!=2 → True
   2. "\*\*" Ist der Potenzoperator. Links vom Operator steht ein Operand, welcher die Basis ist, und rechts davon steht ein weiterer Operand, welcher der Potenz entspricht. Bsp: 4\*\*2 = 16
   3. "%" Der Modulus ist eine arithmetische Operation, die den Rest ergibt, wenn die erste Zahl durch die zweite Zahl geteilt wird. Bsp: 5 % 2 = 1, 4 % 2 = 0
3. Ein Bild, das Tisch enthält.

   Automatisch generierte Beschreibung
4. Ein Bild, das Tisch enthält.

   Automatisch generierte Beschreibung

T = True

F = False

1. Ein Bild, das Tisch enthält.

   Automatisch generierte Beschreibung
2. b or not(b) → immer True
3. 0
4. -21
5. 25
6. 64
7. 6.25
8. 6
9. 4
10. False
11. True
12. True
13. False
14. True
15. False
16. False
17. False
18. False

# Exercice 4

3) x = 4, y = 2, z = 16  
4) x = 20, y = 2, z = 16  
5) x = 20, y = 2, z = 55  
6) x = 20, y = 2, z = 69  
7) x = 10, y = 2, z = 69

# Exercice 5

1. x % 8 == 0 and x < 66
2. x % 2 != 0 and -2 <= x < 14
3. -6 < x <= 0 or 8 <= x < 42
4. a != b

# Exercice 6

a)

Variable name: “size”, type: Int (Nombre entier), valeur initiale: 200

Variable name: “flowers”, type: Int (Nombre entier), valeur initiale: 8

Variable name: “petals”, type: Int (Nombre entier), valeur initiale: 7

Variable name: “thickness”, type: Float (Nombre réel), valeur initiale: 3.5

Variable name: “color\_stem”, type: string (text), valeur initiale: “green”

Variable name: “color\_flower”, type: string (text), valeur initiale: “magenta”

Variable name: “flower\_ok”, type: Int (Nombre entier), valeur initiale: 0

Variable name: “petals\_ok”, type: Int (Nombre entier), valeur initiale: 0

b) Ligne 3, Ligne 9, Ligne 20, Ligne 27 und Ligne 37

c)

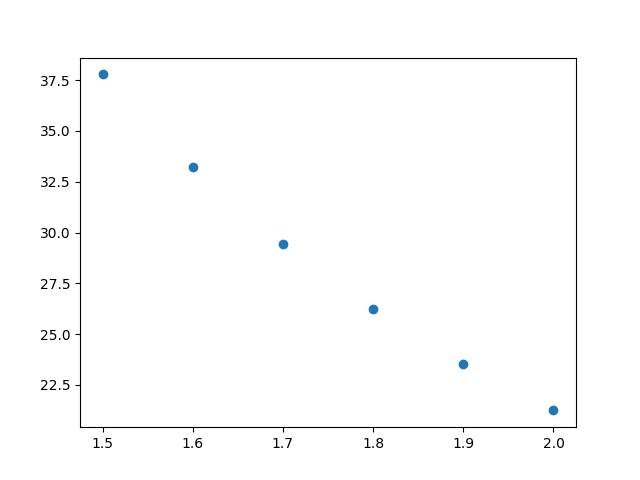


Figure 1

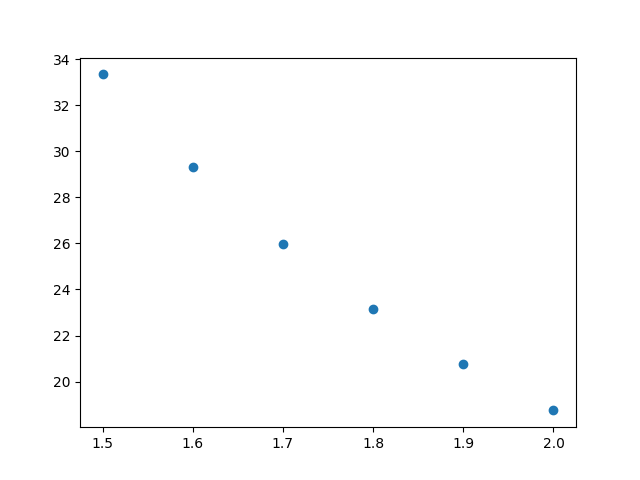


Figure 2

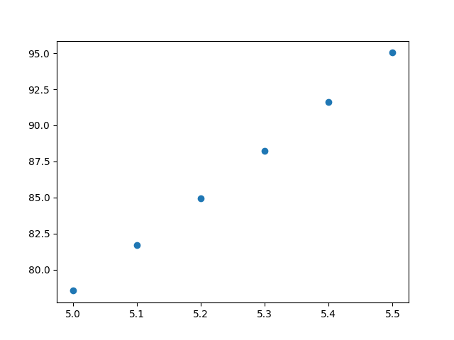


Figure 3

Python Code:

# a) Declare the variable m w`hich you set as the mass value 75 [kg].

m = 75

# b) Declare 6 variables t1, . . . , t6 which you set as height values between 1.5 and 2.0 [m].

t1 = 1.5

t2 = 1.6

t3 = 1.7

t4 = 1.8

t5 = 1.9

t6 = 2.0

# c) Declare 6 variables imc1, . . . , imc6 for which you calculate the BMI values using m and the corresponding values t1,...,t6 and then set them as those values. Use the formula defined above for the BMI.

imc1 = m / (t1 \* t1)

imc2 = m / (t2 \* t2)

imc3 = m / (t3 \* t3)

imc4 = m / (t4 \* t4)

imc5 = m / (t5 \* t5)

imc6 = m / (t6 \* t6)

# d) Create a graph of the variation of the BMI as a function of the height for the mass given and insert it into your solution document. Utilize the python code below.

import matplotlib.pyplot as plt

plt.scatter([t1, t2, t3, t4, t5, t6], [imc1, imc2, imc3, imc4, imc5, imc6])

plt.show()

# e) (optional) Redefine m as a new value 85 [kg], then execute your code again to automati- cally recalculate the values imc1, . . . , imc6 and creating a new graph which you insert in your solution document as well.

m = 85

imc1 = m / (t1 \* t1)

imc2 = m / (t2 \* t2)

imc3 = m / (t3 \* t3)

imc4 = m / (t4 \* t4)

imc5 = m / (t5 \* t5)

imc6 = m / (t6 \* t6)

plt.scatter([t1, t2, t3, t4, t5, t6], [imc1, imc2, imc3, imc4, imc5, imc6])

plt.show()

# f) (optional) On the basis of the previous exercises, calculate the area of 4 circles with values for radius between 5 and 5.5 [cm]. As a reminder, the formula for the area of a circle is: Areaofacircle = π ∗ r2. To use an exact value for π, you may import the ’math’ library and get π with math.pi. Create another graph of the variation of the area of the circles as a function of the radius and insert it into your document.

import math

r1 = 5

r2 = 5.1

r3 = 5.2

r4 = 5.3

r5 = 5.4

r6 = 5.5

area1 = math.pi \* (r1 \* r1)

area2 = math.pi \* (r2 \* r2)

area3 = math.pi \* (r3 \* r3)

area4 = math.pi \* (r4 \* r4)

area5 = math.pi \* (r5 \* r5)

area6 = math.pi \* (r6 \* r6)

plt.scatter([r1, r2, r3, r4, r5, r6], [area1, area2, area3, area4, area5, area6])

plt.show()