

Svolgere i seguenti esercizi, regole generali:

- Si usa ciascuno il proprio cervello e relativa materia grigia 😊
- Non si parla
- Se c'è un dubbio su quanto richiesto seguite i seguenti passaggi:
 - Sforzatevi rileggendo bene la domanda
 - Saltate provvisoriamente la domanda
 - Se avete ancora dubbi alzate la mano → chiedete in modo che tutti possano sentire

1 Azzeramento

Exercizio 1.1: “The interpreter”

Aprirete la console Python – inserite le seguenti espressioni (premete invio) – spiegate l'output:

- (a) `3 + 1`
- (b) `3 * 3`
- (c) `2 ** 3`
- (d) `"Hello, world!"`

Exercizio 1.2: “Scripts & GO!”

Copiate il contenuto del testo sopra in un file denominato `script.py`, cosa succede se lo avviate?
Provate a renderlo funzionante.

Esercizio 1.3: “More interpreter”

Spiegare l’output dei seguenti comandi:

- (a) `'py' + 'thon'`
- (b) `'py' * 3 + 'thon'`
- (c) `'py' - 'py'`
- (d) `'3' + 3`
- (e) `3 * '3'`
- (f) `a`
- (g) `a = 3`
- (h) `a`

Esercizio 1.4: “Booleans”

Spiegare l’output dei seguenti comandi:

- (a) `1 == 1`
- (b) `1 == True`
- (c) `0 == True`
- (d) `0 == False`
- (e) `3 == 1 * 3`
- (f) `(3 == 1) * 3`
- (g) `(3 == 3) * 4 + 3 == 1`
- (h) `3**5 >= 4**4`

Esercizio 1.5: “Integers”

Spiegare l’output dei seguenti comandi:

- (a) `5 / 3`
- (b) `5 % 3`
- (c) `5.0 / 3`
- (d) `5 / 3.0`
- (e) `5.2 % 3`
- (f) `2001 ** 200`

Esercizio 1.6: “Floats”

Spiegare l’output dei seguenti comandi:

- (a) `2000.3 ** 200` (comparare con sopra)
- (b) `1.0 + 1.0 - 1.0`
- (c) `1.0 + 1.0e20 - 1.0e20`

Esercizio 1.7: “Variables”

Scrivete uno scrive nella quale la variabile “nome” contiene una stringa con il vostro nome. Assumendo ora che il vostro nome sia “Mario Rossi”, fate in modo di ottenere l’output “Ciao, Mario Rossi”. (Non usare print(“Ciao, Mario Rossi”) 😊)

Esercizio 1.8: “Type casting”

Molto spesso si vuole convertire o imporre il tipo (di dato n.d.r.) di una variabile.

In Python è possibile farlo usando tipodidatodesiderato(x), esempio: int(x)

Spiegare l’output dei seguenti comandi:

- (a) `float(123)`
- (b) `float(' 123')`
- (c) `float(' 123.23')`
- (d) `int(123.23)`
- (e) `int(' 123.23')`
- (f) `int(float(' 123.23'))`
- (g) `str(12)`
- (h) `str(12.2)`
- (i) `bool(' a')`
- (j) `bool(0)`
- (k) `bool(0.1)`

2 “Flussi di controllo”

Esercizio 2.1: “Range”

Scrivete `range(5)` nella console, cosa ottenete? Perciò cosa significa `for i in range(5)`?

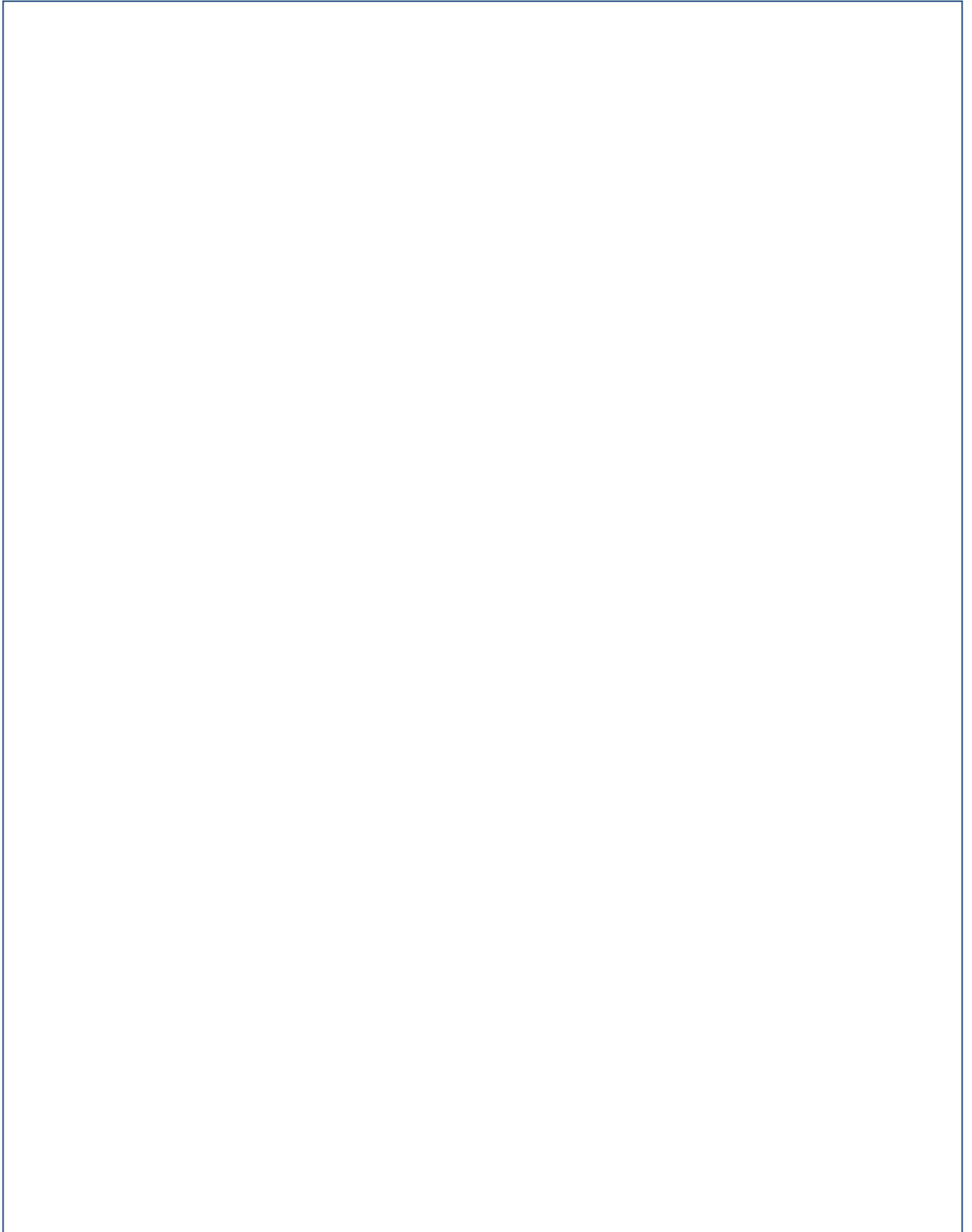
Cerchiamo di avere qualche informazione in più: cosa succede scrivendo nella console:

`type(range(5))`?

Esercizio 2.2: “For loops”

Usate un ciclo `for` per:

- (a) Visualizzare a schermo i numeri da 0 to 100
- (b) Visualizzare a schermo i numeri da 0 to 100 che sono divisibili per 7
- (c) Visualizzare a schermo i numeri da 1 to 100 che sono divisibili per 5 ma non per 3
- (d) Visualizzare a schermo per ognuno dei numbers $x = 2, \dots, 20$, tutti i numeri che sono divisori di x , escludendo 1 e x . Quindi, per 18, deve stampare: 2 3 6 9.



Esercizio 2.3: “Simple while loops”

Invece di un ciclo for utilizzare un ciclo while per:

- (a) Stampare i numeri da 0 to 100
- (b) Stampare i numeri da 0 to 100 divisibili per 7

Esercizio 2.4: “While loops”

Usare un loop `while` per trovare e stampare i primi numeri divisibili per 5, 7 e 11. Utilizzare una variabile per salvare i numeri trovati progressivamente.

Pseudo-codice:

```
number found = 0
x = 11
while number trovato minore di 20: if
    x divisibile per 5, 7 e 11:
        print x
        incrementare il numero trovati
        incrementare il numero di x
```


Esercizio 2.3: “More while loops”

Il numero più piccolo divisibile per 2, 3 e 4 è 12. Trovate il numero più piccolo divisibile per 1 e 10.

3 “Functions”

Esercizio 3.1: “Hello!”

- (a) Scrivete una funzione `hello_world` che stampi `'Hello, world!'`
- (b) Scrivete una funzione `hello_name(name)` che stampi `'Hello, name!'` `name` è di tipo `string`.
- (c) Spiegare la differenza tra `print` e `return`. Cosa succede se invece di `print` usate `return`?

Esercizio 3.2: “Maximum”

Scrivete una funzione denominata `my_max(x, y)` che ritorni il massimo tra `x` e `y`. Non usate la funzione `max` 😊, ma `if` ne seguenti modi:

- (a) Usare sia `if` e `else`.
- (b) Usate `if` ma non `else` (e neanche `elif`).