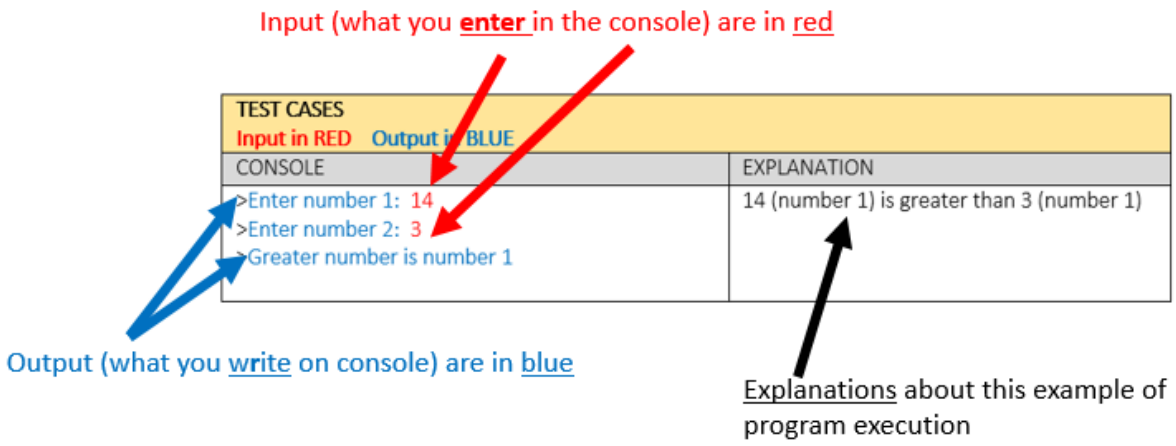
# BEFORE STARTING!

Manage your time: you can do everything in 2 days, or follow the plan day by day.
But: don't do everything at the last minute!

Think by yourself: don't ask the answer to other students
Your facilitator is here to answer your questions – **not the students!**

Read the **instructions**!
Also **read** the **examples**, they will help you to understand what we ask you to do

NEW!! Now we display input and output together, but using different colors…



# MONDAY

## THE INDEX OF 7

| WHAT YOUR PROGRAMM SHALL DO |
|---|
| - Enter 1 number (numberOfValues) in the console<br>- Enter < numberOfValues> numbers in the console<br><br>-Print the **index** of the **first** 7 found among the entered values:<br>- If no 7 is entered, print "No 7 entered" |

Example: if will enter 4 values: 5, 7, 6, 8 then the result to print is 1, since the first 7 is at index 1


| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >Number of values: 4<br>>5<br>>5<br>>7<br>>5<br>>7 index is: 2 | Here we enter 4 values : 5, 5, 7, 5<br>The first 7 found is at index 2 in this list of numbers<br><br>So we print 2 |
| >Number of values: 3<br>>6<br>>5<br>>2<br>>No 7 entered | Here we enter 3 values : 5, 5, 2<br>There is no 7 in this list |
| >Number of values: 5<br>>6<br>>5<br>>7<br>>1<br>>7<br>>7 index is: 2 | Here we enter 5 values : 6, 5, 7, 1, 7<br><br>The first 7 found is at index 2 in this list<br>There is another 7 at index 4 but we take the first one!<br><br>So we print 2 |

# CORRECTION

```
numberValues = int(input("Number of values:"))

sevenIsFound = False
```

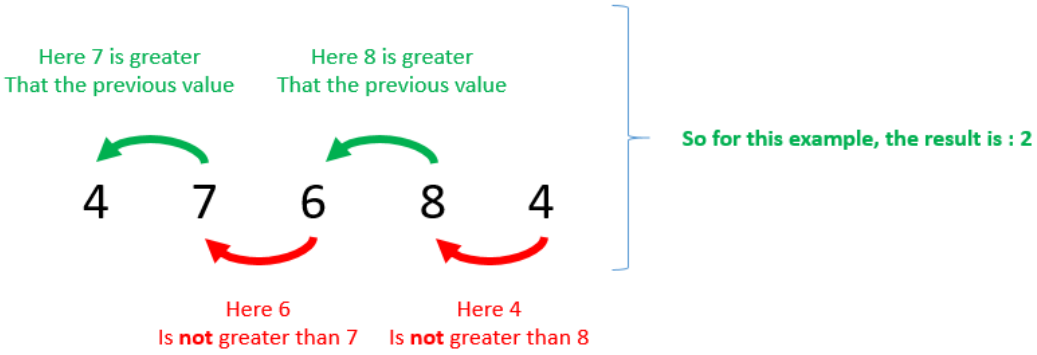
```
sevenIndex = 0

for index in range(numberValues):
    value = int(input())

    if value == 7 and not sevenIsFound:
        sevenIsFound = True
        sevenIndex = index

if sevenIsFound:
    print("7 index is: " + str(sevenIndex))
else:
    print("No 7 entered")
```

# TUESDAY

## GREATER THAN PREVIOUS ONE!

| WHAT YOUR PROGRAMM SHALL DO |
|---|
| - Enter 1 number (numberOfValues) in the console<br>- Enter < numberOfValues> numbers in the console<br><br>-Print the **number** of the **time** a value is greater than the previous value on the list |

Example: if will enter 5 values: 4, 7, 6, 8, 4 then the result to print is 2, since we have 2 times a number greater than the previous number:




| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >Number of values: 3<br>>4<br>>1<br>>3<br>>Result is : 1 | Here we enter 3 values : 4, 1, 3<br>- 1 is NOT greater than 4 (0 found)<br>- 3 is GREATER than 1 (1 found)<br><br>So we print 1 |
| >Number of values: 4<br>>1<br>>2<br>>3<br>>5<br>>Result is : 3 | Here we enter 4 values : 1, 2, 3, 5<br>- 2 is greater than 1 (1 found)<br>- 3 is greater than 2 (1 found)<br>- 5 is greater than 3 (3 found)<br><br>So we print 3 |

| CAN I HAVE SOME HELP? |
|---|
| Here you need to find a way to “remember” the previous value<br><br>Also be careful about when do you start to compare the values: first iteration? Second iteration? Third iteration? |

# CORRECTION

```
numberValues = int(input("Number of values:"))

numberOfGreater = 0
previousValue = 0

for index in range(numberValues):
    currentValue = int(input())

    if index > 0 and currentValue > previousValue:
        numberOfGreater = numberOfGreater+1

    previousValue = currentValue

print("Result is: " + str(numberOfGreater))
```

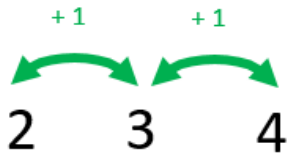
Notes for teachers:

- Students should be careful: during the first iteration (index 0) there is no previous value!
- Here we are using 1 variable to keep the value as “previous value” before going to the next step

# WEDNESDAY

What is a sequence of 3 consecutive numbers?

- Consecutive numbers means the next number is equal to the previous + 1
- Therefore a sequence of 3 consecutive number N1, N2, N3 means : N2 = N1 +1 and N3 = N2 +1
    - Example : 2, 3, 4 or 10, 11, 12




WEDNESDAY EXERICICE 1

| WHAT YOUR PROGRAMM SHALL DO |
|---|
| - Enter 3 numbers (n1, n2, n3) in the console<br><br>- Print "SEQUENCE" if the 3 numbers are a **sequence of 3 consecutive numbers**<br>- Otherwise print "NOT SEQUENCE" |

| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >5<br>>6<br>>7<br>>SEQUENCE | 5 6 7 is a consecutive sequence of number :<br><br>5 +1 = 6<br>6 + 1= 7 |
| >5<br>>6<br>>8<br>>NOT SEQUENCE | |

# CORRECTION

```
n1 = int(input("N1:"))
n2 = int(input("N2:"))
n3 = int(input("N3:"))

isConsecutive = (n2 == n1 + 1) and (n3 == n2 + 1)
if isConsecutive:
    print("SEQUENCE")
else:
    print("NOT SEQUENCE")
```

Note: as you guess, it's just an appetizer for the next exercise ☺

WEDNESDAY EXERICICE 2

| WHAT YOUR PROGRAMM SHALL DO |
|---|
| - Enter 1 number (numberOfValues) in the console<br>This number must be equal to 3 or greater<br>- Enter < numberOfValues> numbers in the console<br><br>- Print the number of **sequence of 3 consecutive numbers** you found in the list of numbers |

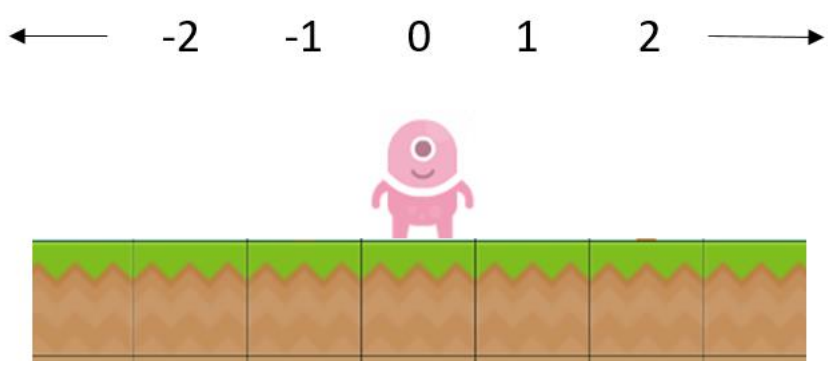
| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >Number of values: 4<br>>10<br>>2<br>>3<br>>4<br>>1 | Here we enter 4 values : 10, 2, 3, 4<br><br>This is only 1 sequence : 2, 3, 4<br><br>So we print 1 |
| >Number of values: 5<br>>6<br>>2<br>>3<br>>4<br>>5<br>>2 | Here we enter 5 values : 6, 2, 3, 4, 5<br><br>This is only 2 sequences<br>2, 3, 4<br>3, ,4 5<br><br>So we print 2 |
| >Number of values: 3<br>>6<br>>2<br>>3<br>>0 | Here this is not sequence,<br><br>So we print 0 |

```
numberValues = int(input("Number of values:"))
numberOfSequences = 0
n1 = 0
n2 = 0

for index in range(numberValues):
    n3 = int(input())

    # Check if n1, n2, n3 are consecutive (only from iteration 2)
    if index >= 2:
        isConsecutive = (n2 == n1 + 1) and (n3 == n2 + 1)
        if isConsecutive:
            numberOfSequences = numberOfSequences + 1

    # Update n2
    if index >= 1:
        n1 = n2

    # Update n1
    n2 = n3

print(numberOfSequences)
```

# THURSDAY

## LET'S MOVE BALOOK!

Do you remember Balook? The robot who can go left and right… Let's start to move it a bit.




You can enter the following actions on the console:

- L (left) will move Balook to the **left**
- R (right) will move Balook to the **right**
- S (stop) stop the program

Balook is at position 0 at the beginning

- If you move left, position is decreased of 1
- If you move right, position is increased of 1

| WHAT YOUR PROGRAMM SHALL DO |
|---|
| - Enter a string ( the action ) : which shall be only : L (for left), R (for right) , S (for stop)<br>- Regarding the action, move the position of Balook<br>- Continue to enter action, and stop program when the action is S (stop)<br><br>-Print the **final position of Balook** |

| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >Action: L<br>>Action: L<br>>Action: L<br>>Action: S<br>>Balook position is: -3 | We entered 3 times the action L (left)<br>So the position at the end is -3 (0 -1 -1 -1) |
| >Action: L<br>>Action: R<br>>Action: L<br>>Action: R<br>>Action: S<br>>Balook position is: 0 | We entered 2 LEFT and 2 RIGHT<br>So the position at the end is 0 (0 +1 -1 +1 -1) |

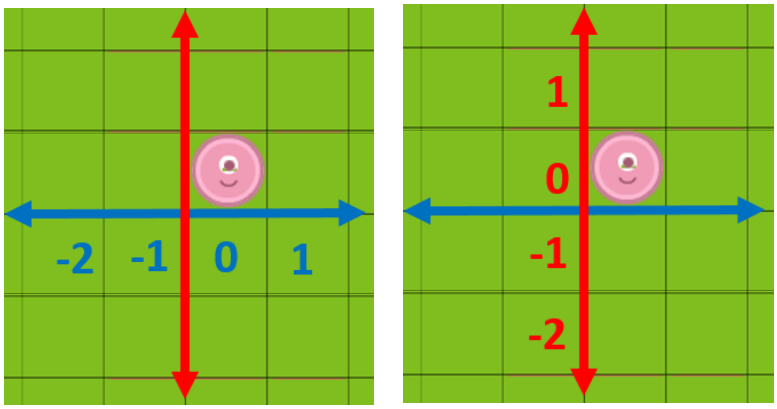
# CORRECTION

```
canContinue = True
balookPosition = 0

while canContinue:
    action = input("Action:")

    if action == "R":
        balookPosition = balookPosition + 1
    elif action == "L":
        balookPosition = balookPosition - 1
    elif action == "S":
        canContinue = False

print("Balook position is: " + str(balookPosition))
```

## FRIDAY

### LET'S MOVE BOSOK!

Bosok… This robot can move right, left, up and down. Do you remember?




*Right now, Bosok is at position (0, 0) which mean 0 on X and 0 on Y*

- So Bosok has 2 positions: horizontal (on X axis) and vertical (on Y axis)
- We represent Bosok position like this: (3, 4) which means: 3 on X and 4 on Y

<u>You can enter the following actions on the console:</u>

- L (left) will move Bosok **left**
- R (right) will move Bosok **right**
- U (up) will move Bosok **up**
- D (down) will move Bosok **down**
- S (stop) stop the program

Bosok is at position (0, 0) at the beginning

**WHAT YOUR PROGRAMM SHALL DO**

- Enter a string ( the action ) : which shall be only : L, R, R, U, D, S
- Regarding the action, move the position of Balook
- Continue to enter action, and stop program when the action is S (stop)
-Print the **final position of Bosok**

| EXAMPLES | |
|---|---|
| CONSOLE | EXPLANATION |
| >Action: L<br>>Action: U<br>>Action: L<br>>Action: S<br>>Bosok position is: (-2, 1) | We entered 2 LEFT and 1 UP<br>- position on X is : 0 -1 -1 = -2<br>- position on Y is : 0 + 1 = 1<br>So the position at the end is (-2, 1) |
| >Action: D<br>>Action: D<br>>Action: D<br>>Action: S<br>> Bosok position is: (0, -3) | We entered 3 D<br>- position on X is : 0<br>- position on Y is : 0 -1 -1 -1 = -3<br>So the position at the end is (0, -3) |

# CORRECTION

```
canContinue = True
bosokX = 0
bosokY = 0

while canContinue:
    action = input("Action:")

    if action == "R":
        bosokX = bosokX + 1
    elif action == "L":
        bosokX = bosokX - 1
    elif action == "U":
        bosokY = bosokY + 1
    elif action == "D":
        bosokY = bosokY - 1
    elif action == "S":
        canContinue = False

print("Balook position is: (" + str(bosokX) + ", " + str(bosokY) + ")")
```