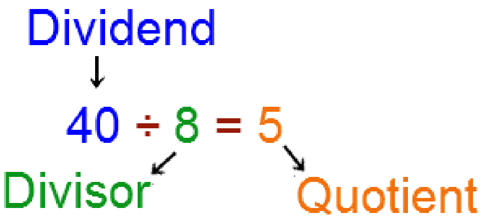
TUESDAY

EXERCICE 1



WHAT YOUR PROGRAM SHALL DO

- Enter 2 numbers (dividend and divisor) in the console
  CONSOLE : console shall display Enter dividend :
  CONSOLE : console shall display Enter divisor :
- Display the quotient as float
  CONSOLE : console shall display Quotient (float) : <quotientAsFloat>
- Display the quotient as integer (the rounded value as integer)
  CONSOLE : console shall display Quotient (integer) : <quotientAsInteger>
- If divisor is equal to 0, console shall display : Division by 0 not allowed

EXAMPLES

INPUT	EXPLANATION
>Enter dividend : 35 >Enter divisor : 2 >Quotient (float) : 17.5 >Quotient (integer) : 17	17.5 is the decimal (float) result of 35/2 17 is the rounded integer of this division 35/2
>Enter dividend : 35 >Enter divisor : 0 >Division by 0 not allowed	

HELP ME!!

To convert a float number to a integer number, you can use the instruction:

```
int( <your_number> )
```

CORECTION

```
dividend = int(input("Enter dividend:"))
divisor = int(input("Enter divisor:"))
if divisor == 0:
    print("Division by 0 not allowed")
else:
    quotien = dividend / divisor
    print("Quotien  (float)  :"+str(quotien))
    print("Quotien (integer) :"+str(int(quotien)))
```

EXERCICE 2

$$\underbrace{14}_{\text{INTEGER PART}}\,.\,\underbrace{32}_{\text{DECIMAL PART}}$$

WHAT YOUR PROGRAM SHALL DO

- Enter 1 float number **as a string** in the console
  CONSOLE : console shall display Enter float:

- Display the **INTEGER** part of this number
  CONSOLE : console shall display Integer part : <integerPart>

- Display the **DECIMAL** part of this number
  CONSOLE : console shall display Decimal part : <decimalPart>

- *Note : if NO decimal part, print only the integer part*

EXAMPLES	
INPUT	EXPLANATION
>Enter float : 35.5 >Integer part : 35 >Decimal part : 5	35.5 has a integer part equal to 35 and a decimal part equal to 5
>Enter float : 12 >Integer part : 12	Here we have only the integer part

HELP ME!!

Here it’s easier to enter the float number as a string, to manipulate it character by character

We use the point “.” to separate the integer part from the decimal part.

CORRECTION

```
floatString = input("Enter float:")

# 1 - Check if the number if decimal and get the dot position
hasDot = False
dotPosition = 0
for index in range(len(floatString)):
    if floatString[index] == ".":
        hasDot = True
        dotPosition = index

# 1 - Print the integer and the decimal parts
if hasDot:
    integerString = floatString[0:dotPosition]
    decimalString = floatString[dotPosition+1:]

    print("Integer part : " + integerString)
    print("Decimal part : " + decimalString)
else:
    print("Integer part : " + floatString)
```

WEDNESDAY

(abc)	GOOD
((abc)d)	GOOD
(abc))	NOT GOOD – ONE EXTRA CLOSING PARATHENSIS
(((abc)d)	NOT GOOD – ONE EXTRA OPENING PARATHENSIS

WHAT YOUR PROGRAM SHALL DO

- Enter a text in the console

  CONSOLE : console shall display Enter a text :

- If each opening parenthesis "(" is well closed by a ")", display GOOD FORMAT
- Else display BAD FORMAT

➔ Each opening parenthesis should be well closed by a closing parenthesis

EXAMPLES	
INPUT	EXPLANATION
>Enter text : ((Ronan)Hugo) >GOOD FORMAT	Here we have 2 opening parenthesis and 2 closing parenthesis

>Enter text : (a) b) c (>BAD FORMAT	Here the losing parenthesis is green is not correct, since There is not more opening parenthesis to close
>Enter text : ((Ronan)Hugo))) >BAD FORMAT	Here we have 2 opening parenthesis and 4 closing parenthesis This is a BAD format

CORRECTION

```
text=str(input("Enter text:"))
count=0
formatStillGood=True
for i in range(len(text)):

 # 1- Update the counter
  if text[i] =="(":
    count=count+1

  if text[i]==")":
    count=count-1

 # 2- Check that counter is not negative
  if count < 0:
      formatStillGood=False

# 3- Print result
if count==0 and formatStillGood:
  print("good format")
else:
  print("bad format")
```

THURSDAY

WHAT YOUR PROGRAM SHALL DO

- Enter a number of hours
  CONSOLE : console shall display Number of hours:

- Enter a number of minutes
  CONSOLE : console shall display Number of minutes:

- Enter a number of seconds
  CONSOLE : console shall display Number of seconds:

- Display the **total number of seconds of this time**
  CONSOLE : console shall display Total seconds: <result>

- Note : You must check that :
  - hours number is in range [0, 24[
  - minute number is in range [0, 60[
  - second number is in range [0, 60[

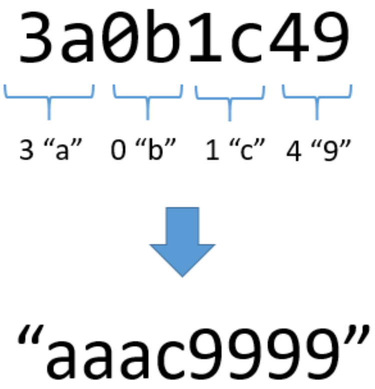
- Otherwise print : Time format error

EXAMPLES	
INPUT	EXPLANATION
>Number of hours: 4 >Number of minutes: 35 >Number of seconds: 10 >Total seconds: 16510	4 * 3600 + 35* 60 +10 = 16510 seconds
>Number of hours: 75 >Number of minutes: 35 >Number of seconds: 10 >Time format error	hours number is NOT in range [0, 24] This is a BAD format

CORRECTION

```
hours=int(input("Number hours :"))
minutes=int(input("Number minutes :"))
seconds=int(input("Number seconds :"))
if hours<0 or hours>=24 or minutes<0 or minutes>=60 or seconds<0 or seconds>=
60:
  print("Time format error")
else:
  total=hours*3600+minutes*60+seconds
  print("Total number of seconds: ",total)
```

FRIDAY



WHAT YOUR PROGRAM SHALL DO

- Enter a code as a string
  CONSOLE : console shall display Enter code:

- This code is composed of pair of 2 values :
  - The first value of the pair is the number of repeats
  - The second value is the character to repeat

- As an example : “3a4b” code shall produce aaabbbb (3 a and 4 b)

- Print the created string
  CONSOLE : console shall display String is:

EXAMPLES

INPUT	EXPLANATION
>Enter code: 3a0b1c49 >String is: aaac9999	We need to print: 3 “a” 0 “b” 1 “c” 4 “9”

HELP ME!!

Note for this exercise, we don’t ask you to check if the input is in the good format

- We suppose the user enter the right input

CORRECTION

```
enterString = input("Enter code: ")
charRepetition = 0
result = ""
for n in range(len(enterString)):
    char = enterString[n]
    if n % 2 == 0:
        charRepetition = int(char)
    else:
        for m in range(charRepetition):
            result = result + char

print("string is: ", result)
```