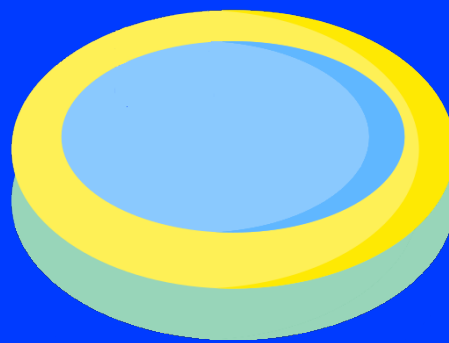




WADING POOL

< DAY 04 />



WADING POOL



OH MY GIT!

In addition to the tasks below, you must go as far as possible in [this game](#).

Work on it as soon as you have a bit of time, or whenever you need a break in you day!



Conditionals



From now on, if ever you are stuck on a task, we warmly recommend you to try a no-code tool, such as [Flowgorithm](#), that generates code from graphical flowcharts.

Task 1.1



Evaluate and explain the following lines:

- ✓ `(42 > 12)`
- ✓ `(12 = 12)`
- ✓ `(12 == 12)`
- ✓ `("hello" == "world")`
- ✓ `(218 >= 118)`
- ✓ `("a".upper() == "A")`
- ✓ `(1 * 2 * 3 * 4 <= 9)`
- ✓ `("z" in "azerty")`

Task 1.2



Ask an integer to the user. If it's equal to 42, display "This is correct!".

Task 1.3



Prompt the user for an integer. Then, if it is:

- ✓ odd, display "This integer is odd" ;
- ✓ even, display "This integer is even".

Task 1.4



Prompt the user for a string:

- ✓ if it's "open sesame", display "access granted" ;
- ✓ if it's "will you open, you goddamn !@&/°", display "access fucking granted" ;
- ✓ else, display "permission denied".



Flowgorithm



Task 1.5



Prompt the user for an integer:

- ✓ if it's 42, display "a" ;
- ✓ if it's smaller or equal than 21, display "b" ;
- ✓ if it's even, display "c" ;
- ✓ if this integer divided by 2 is smaller than 21 (excluded), display "d" ;
- ✓ finally, if it's odd and greater or equal than 45, display "e" ;
- ✓ in any other cases, display "f".



Flowgorithm

Task 1.6



Dig the following code.

Then, run it.

Finally, fix it.

```
a == 42
b == 41
if a = b
    print("A and B is the sames")
if b =< a
    print("B is equal or lower as A")
if b != a
    print("B his diferent from A")
```



As you are at it, also correct the grammar and orthographic mistakes...

what are other
words for
mistakes?



faults, errors, omission,
misstep, aberration, fault,
lapse, miscalculation, blunder,
inaccuracy



Thesaurus.plus

Loops

Task 2.1



Write a snippet of code that prints all integers from 1 to 1000.

Task 2.2



Prompt the user for a string.
Display all the characters of this string twice.



For instance `taxi` will return `ttaaxxi`.

Task 2.3



Print all integers, decreasingly from 10 000 to 1, that are divisible by 7.

2 THANK	3 YOU	4 ALL
5 FOR	6 HAVING	7 NOT YOU
8 EASY	9 DIVISIBILITY	10 RULES

Task 2.4



For all integers from -30 to 30:

- ✓ if it's a multiple of 3, display "Fizz" ;
- ✓ if it's a multiple of 5, display "Buzz" ;
- ✓ if it's a multiple of 3 and 5, display "FizzBuzz" ;
- ✓ if it does not meet any of the previous conditions, just print the integer itself.



Task 2.5



Generate the lyrics of the song "99 bottles of beer".
The song ends when there is no more bottles on the wall.



1 bottle is singular...

Task 2.6



Write a program that takes an integer n as input. Then, for each integer from 2 to $n/2$, display the list of its multiples strictly smaller than n , in descending order.



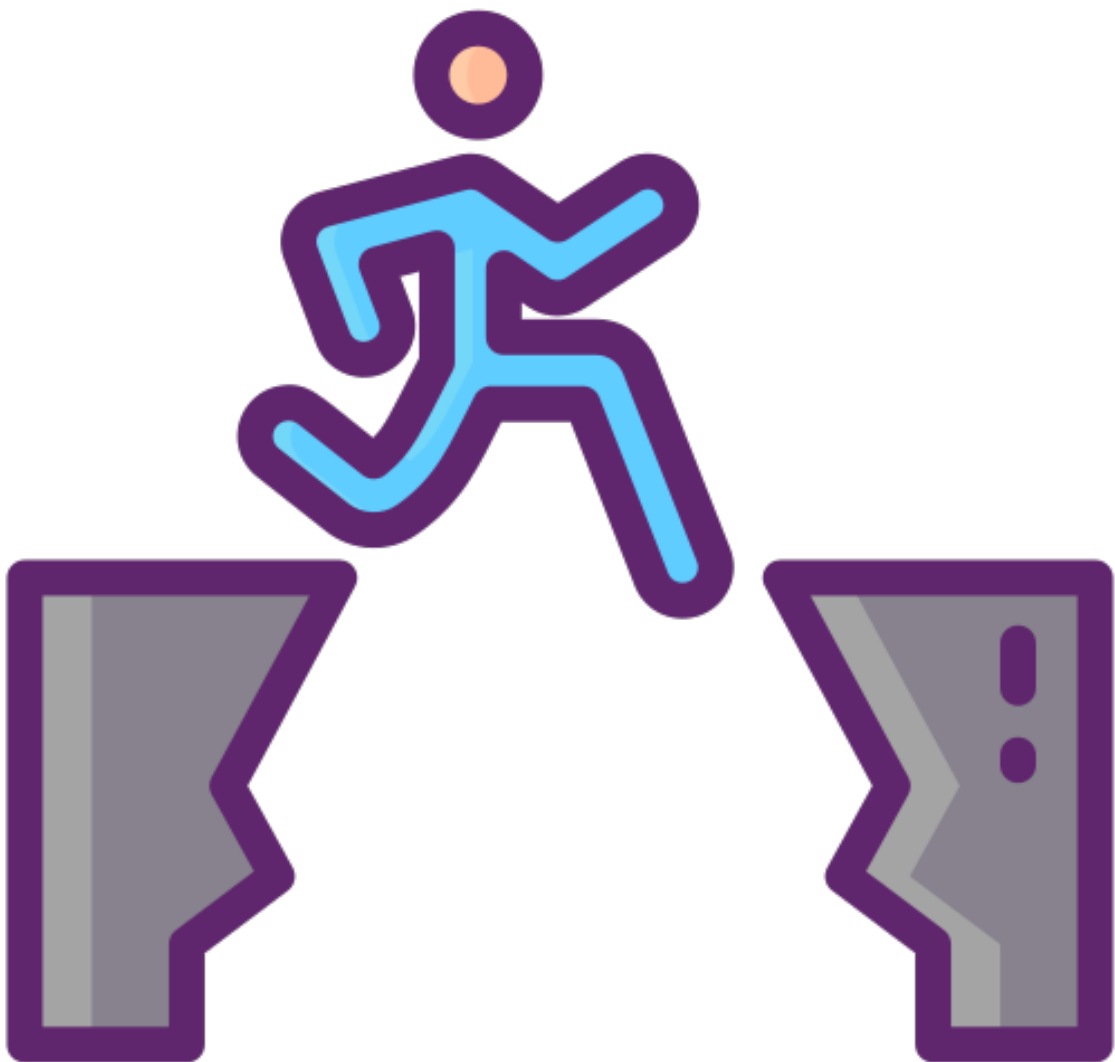
For example, for $n = 14$, your program should display:

```
12 10 8 6 4 2
12 9 6 3
12 8 4
10 5
12 6
7
```

CHALLENGE

Write the shortest code possible that realizes the following:

- ✓ prompt the user simultaneously for an integer and a string ;
- ✓ if the integer is 0, then quit ;
- ✓ if the string contains a vowel, display the integer ;
- ✓ if the integer is greater or equal than 42, display the integer ;
- ✓ else display the string.



Encryption

Task 3.1



The **Caesar Cipher** is a simple encryption technique named after Julius Caesar, who used it to encrypt messages of military significance.

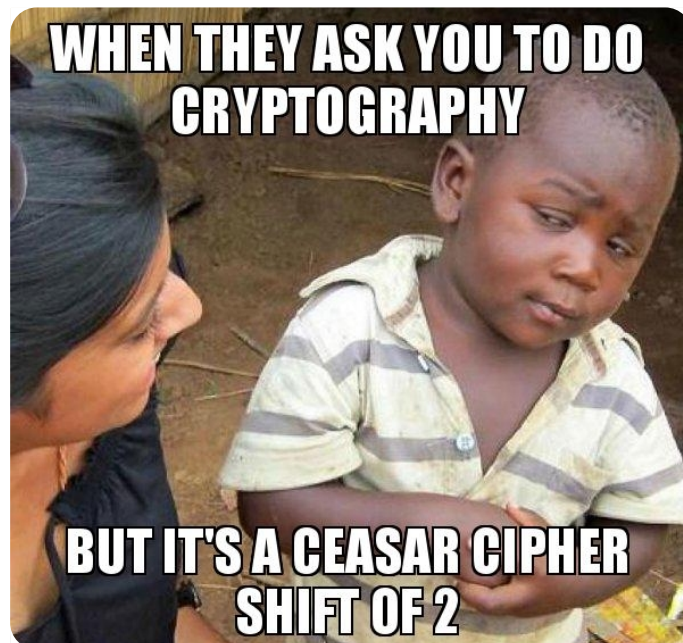
It is a type of substitution cipher in which each letter of a plaintext is replaced by a letter with a fixed number of positions down the alphabet.

The encryption key is an integer between 1 and 25 that determines the shift of the letters. To decrypt the message, the recipient simply uses the same key to shift the letters back.

As examples:

- ✓ `ab` with a key of 3 gives `de` ;
- ✓ `hello world` with a key of 4 gives `lipps asvph` ;
- ✓ `the zen of python` with a key of 13 gives `gur mra bs clguba`.

Prompt the user for a clear message and a key. Then, use Caesar cipher to encrypt the clear message with the key. Finally, print the encrypted message.



Website, such as [dcode](#) may help you.

Is that not enough for you?



Task 3.2



Write a program that can decrypt any Caesar-ciphered text.

Task 3.3



Write a program that can encrypt and decrypt a text using a Vigenère code.

Task 3.4



Write a program that can decrypt an English Vigenère-ciphered text given the length of the key, but without knowing the key.



Day 03 Task 3.5

v2

{EPITECH}