

Episod II : Conditional branches and iterative structures


EXERCISE 1

Write a VBA script (Sub oddsquares()) that asks the user to enter an integer between 1 and 100 and then calculates the sum of the squares of odd numbers between 1 and this number.
We can make sure that the user enters a number in the requested interval.

EXERCISE 2

Write a VBA script (Sub guessthrow()) that will oppose you to your program. The computer simulates the roll of a 6-sided die ($x = \text{Int}(6 * \text{Rnd}() + 1)$). You have to guess this number in at most 3 tries. If you guess the number you are looking for, the program congratulates you. If not, the program suggests that you try again until you reach the maximum 3 tries allowed.

INDICATION

 To randomly obtain an integer x in the range $[a, b]$, one can use the formula $x = \text{Int}(((b-a)+1) * \text{Rnd}() + a)$.


EXERCISE 3

Write a program (Sub calculatesum()) that will oppose you to your program. The computer asks you to calculate the sum of two random integers between 1 and 100. You must calculate this sum in no more than 5 tries. If you find the number you are looking for, the program congratulates you. If not, the program suggests that you start again until you reach the maximum 5 attempts allowed.

EXERCISE 4

Write a program (Sub palindrome()) that indicates if a word given by the user is a palindrome. A word is a palindrome if the order of its letters remains the same when read from left to right or vice versa.
Example : radar, kayak, Laval.

INDICATION

 In VBA, the `StrReverse(aString)` function allows to invert the order of characters in a string passed as argument. It returns the string resulting from this operation. String comparison is by default case sensitive, because the binary representation of each character is considered. For example, Laval is not a palindrome in this context. To make the comparison case insensitive, use the "Option Compare Text" statement at the beginning of the module of this exercise or use `LCase(aString)` to lowercase the string ...

EXERCISE 5

We assume a word and a letter entered by the user. Write the program (Sub guesaletter()) that checks if the given letter is a letter of the word. The program stops as soon as it is verified that the letter is a letter of the word.



INDICATION



To know the length of the word we will use `len("word")` and to get the *i*th letter `mid("word",i,1)`.



EXERCISE 6

This program (`Sub password()`) will oppose two players. The first player chooses a password and the second one tries to guess it. The number of attempts can be limited and Player 1 can be declared the winner if the password is not discovered after these authorized attempts.



EXERCISE 7

Write a program (`Sub numberofvowels()`) that returns the number of vowels contained in a string entered by the user.



EXERCISE 8

Write a program (`Sub pascal()`) that asks the user to enter an integer number *n* and then displays on Sheet1 of the Excel workbook the first *n* lines of Pascal's triangle.

Lexique des variables

<i>n</i>	(entier)	le nombre de lignes à afficher	DONNÉE
<i>i</i>	(entier)	numéro de ligne courant	INTERMÉDIAIRE
<i>j</i>	(entier)	numéro de colonne courant	INTERMÉDIAIRE
<i>x</i>	(entier)	le nombre courant dans le triangle	INTERMÉDIAIRE

Algorithme

```
pour i de 0 à n - 1 faire  
    x ← 1  
    écrire x  
    pour j de 1 à i faire  
        x ← (x × (i - j + 1)) / j  
        écrire " ", x  
    fpour  
    écrire "\n"  
fpour
```