Handout #7: Evaluating a solution

EXERCISE #1

Mark the following statements as true or false:

A. Most real-world software solutions are shown be correct empirically rather than mathematically. **True**

B. Reducing the number of failing tests doesn't strengthen your claim to a solution's correctness. **True**

C. An inefficient solution cannot be considered correct. **False**

D. An elegant solution maximises both effectiveness and simplicity at the same time. **True**

E. A usability evaluation begins by explaining to the test subject how to use the system. **False**

EXERCISE #2

Look back at the two examples in the section 'Is it elegant?' Write generic versions of both the obvious solution and Gauss's solution, both of which sum up numbers from 1 to N.

Obvious solution:

input upper_number from user

let total = 0

for each number, n, between 1 and upper_number:

    let total = total + n

Gauss's solution:

input upper_number from user

let pair_sum = 1 + upper_number

if upper_number is even, then

    let number_of_pairs = upper_number / 2

    let total = number_of_pairs x pair_sum

else

    let number_of_pairs = (upper_number - 1) / 2

    let total = number_of_pairs x pair_sum

    let total = total + (pair_sum / 2)