Lab 1

The purpose of this lab is, on the one hand, to make some reminders on algorithms, programming and complexity; and, on the other hand, to learn how to measure the execution time of a program. We use the C programming language.

I. Part I (Sum of the first N natural numbers)

In this part, we make some reminders on algorithms, programming and complexity with a first problem of calculating the sum of integers. This problem is to be solved using the technique of iterative programming.

1. Develop an iterative algorithm, denoted Sum_1, which allows the calculation of the sum, denoted S, of the first n natural numbers :

$$S = \sum_{i=1}^{n} i = 1 + 2 + 3 + \cdots + n$$

The natural integer n is to be read as input (n>=1). Use the loop while.

2. Calculate the time complexity of this algorithm.

3. Calculate the space complexity of this algorithm.

4. Develop the corresponding iterative program, noted PSum_1, with the C language.
Ind : To be able to use large values of n, use double type.

II. Part II (Measuring Execution Time)

In this part, we learn how to measure the execution time of a program using the time management functions of the C programming language (time.h).

1. Take the previous program and include the instructions which allow to measure the execution times T for the sample of values of n given in the table below (the new program is noted PSum_2).

n	…	10^6	$2*10^6$	10^7	$2*10^7$	10^8	$2*10^8$	10^9	$2*10^9$
T									

n	10^{10}	$2*10^{10}$	10^{11}	$2*10^{11}$	10^{12}	$2*10^{12}$	…	…	…
T									

2. Plot the curves of the evolution of the theoretical and experimental execution time in Excel.
3. Compare the curves and interpret the obtained results.