# Assignment 2        [30 points max]

**General reminder**: the code needs to be written in some standard programming language (e.g., Python, C/C++, C#, Java, …) following the conventions of that language and commented so that others can understand it. Do not use AI tools.

## Compulsory        [5 points max]

1. Create a program (in any programming language) to calculate numerically the derivative of $f(x) = \sin x$ by definition: $f'(x) = \lim\limits_{h \to 0} \frac{f(x+h)-f(x)}{h}$.

    a. Calculate the derivative for $x = \pi$ and $h = 1e - 16$ **using** IEEE 754 **[2 points]** **double data type**. Compare the result with the correct, analytically obtained, value and explain why they differ.

      Hint: perform a thorough analysis to identify the true source of the problem; it will help you significantly to with the task 1b.

    b. Without exploiting the knowledge that the derivative of $\sin x$ is $\cos x$, **[3 points]** try to modify the calculation to minimize the error. Do not change $h$. Why is the new solution preferable?

      Hint: use goniometric formulas.

## Elective        [up to 15 points]

2. Create a program (in any programming language) to multiply a band matrix $A$ **[10 points]** by a vector $x$. The band matrix is a sparce matrix $n \times n$ with non-zero entries in a diagonal band of bandwidth $m$, i.e., for $m = 1$, the matrix will have non-zero entries on the main diagonal only, for $m = 2$, non-zero elements are also in one diagonal on either side, etc.

$$
\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} =
\begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1m} & 0 & \cdots & 0 \\
a_{21} & \ddots & & & \ddots & & \vdots \\
\cdots & & & & & & 0 \\
a_{m1} & & & & & & a_{n-m,n} \\
0 & \ddots & & & & & \vdots \\
\vdots & & & & & & a_{n-1,1} \\
0 & \cdots & 0 & a_{n,n-m} & \cdots & a_{n,n-1} & a_{nn}
\end{pmatrix}
\cdot
\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}
$$

The program must contain two versions: brute-force (storing all entries and preforming all multiplications) and optimised one that should be as fast and memory efficient as possible. Always use IEEE 754 **double data type** to store the values.

Hint: For an optimised version, it suffices to work with a much smaller array of ($2m$-1) x $n$ elements do instead of a two-dimensional array $n$ x $n$.

Test your program on band matrices with $n = 5\ 000$ and $m = 5$. Non-zero ele-

ments of matrix *A* and elements of vector *x* should be generated randomly. Verify the optimised version by comparing its outputs (vector *y*) with those produced by the brute-force one. Finaly, compare both versions in terms of time and memory saving and write the results of the comparison into a readme text file (or as a code comment).

3. Register on https://www.hackerrank.com/ and solve any **medium** or **hard** of the challenges in ProjectEuler+ contest. Full points will be given if the code is sufficiently efficient and easy to follow.  **[5 points]**

   The proof of the solution (e.g., in the form of screenshot from your HackerRank profile) must be submitted (together with the code) as a part of this assignment by the deadline.

# Elective – BONUS                                   [up to 10 points]

**Rules**: You may choose either block A, which is designed for working alone, or block B, which is designed for working in teams of 2-4 students. Mixing tasks from these blocks is not allowed, the student needs to specify clearly, which block they chose.

## Block A
BONUS 1                                                                **[5 points]**

Solve any unsolved **hard, advanced** or **expert** challenge in ProjectEuler+ contest. 2 points are guaranteed, the rest depends on the quality of your code. All other rules remain the same.

BONUS 2                                                                **[5 points]**

Solve another unsolved **advanced** or **expert** challenge in ProjectEuler+ contest. 2 points are guaranteed, the rest depends on the quality of your code. All other rules remain the same.

## Block B (TEAM)
BONUS 1                                                               **[10 points]**

Register with a team for Ludum Dare 57, JAM or COMPO, and create your own game following the rules of the organizers. 5 points are guaranteed for any game completed. Full points are guaranteed, if your game will be ranked as the first, second or third, in any category. In all other cases, the number of points depend on the code.

Note: for the submission, please submit the confirmation of your registration plus the list of team members.