Eigen library

The goal of the present exercises is to employ Eigen in order to perform linear algebra operations which can be helpful in computational sciences.

Exercise 1: *Wrapping*

- Declare an array with three columns with random coordinates. You are invited to read the Eigen::Array documentation. The result should be structured like:

$$\begin{bmatrix} p_{1x} & p_{1y} & p_{1z} \\ \vdots & \vdots & \vdots \\ p_{nx} & p_{ny} & p_{nz} \end{bmatrix} \tag{1}$$

- Loop over each point stored in the array and compute the cross product with the vector $[1, 1, 1]$. You should use an auto-range loop of the kind:

```
for (auto & e : array){
  // ...
}
```

  You should read the Reduction documentation and in particular have a care for the **rowwise()** method.

- Modify your code to store the result in another array.
- Reshape the result into an array with a single column (that task can be useful when needing to resolve a linear system)
- What was the storage convention ? (row-major or col-major ?)
- Create now an array with 12 columns. Make a loop over the rows and interpret for each row the first 9 columns as a 3×3 matrix M and the last 3 columns a vector V of size 3.
- Construct an array with each row containing the flattened product $M \cdot V$.

Exercise 2: *Solving a linear system*

- Construct a matrix with the following shape:

$$A_{n\times n} = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \tag{2}$$

- This system is the stiffness matrix of a system of n springs: Compute the determinant to show inversibility.
- Solve the system $Ax = b$ where b is a vector of your choice. Your should read Eigen::solving.

- Verify that the obtained result is the solution.
- Construct A as a sparse matrix.
- Solve again the system with a sparse solver. Eigen::sparseSolvers.
- Which of the two approaches is the fastest ? why ?