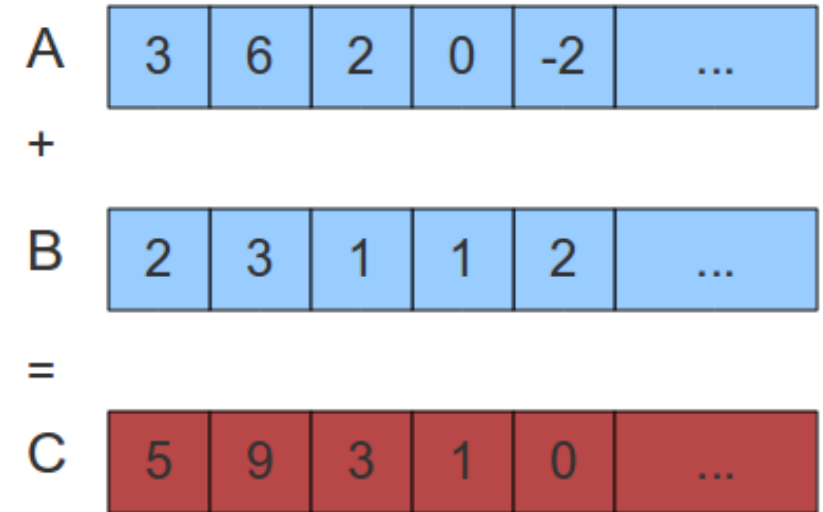


Exercise 1

Write a program that creates, initializes and sums two vectors $A[N]$ e $B[N]$ in a third vector $C[N] = A[N] + B[N]$.

Guidelines:

1. Vector A must be initialized with increasing values from 0 to N --> $A[0] = 0$; $A[1] = 2$; ... ; $A[N-1] = N-1$.
2. Vector B must be initialized with decreasing values from N to 1 --> $B[0] = N$; $B[1] = N-1$; ... ; $B[N-1] = 1$.
3. The initializations of vector A and vector B must be made using two different OpenCL Kernels.
4. Sum Kernel must wait the end of the vectors initializations using the OpenCL Events mechanism.
5. Reading data from the Sum Kernel must wait for the completion of the Sum Kernel execution.



Esercizio 2

Create a program that initializes a vector of N elements and performs the reduction in one element equal to the sum of all the elements.

The array must be initialized with values from zero to N-1.

Help: look at the picture, you have to call the kernel recursively

