# Introduction to FastFlow programming

Massimo Torquati    <torquati@di.unipi.it>

Computer Science Department, University of Pisa - Italy

# ClassWork5: comments

- Let's comment on two possible solution contained in the ClassWork5 folder:

    ~spm1501/public/ClassWork5/maponstream*.cpp

# High-level Patterns in FastFlow

- Divide and Conquer (D&C)   (dc.hpp)

- Macro Data Flow  (mdf.hpp)

- TaskF  (taskf.hpp)

- IterativeStencil computations (stencilReduce*.hpp)

- PoolEvolotuon (poolEvolution*.hpp)


Let's have a look at the D&C pattern

# Divide and Conquer (D&C)

- Simple interface modelling Divide and Conquer computations (DC)

- The programmer using the D&C pattern needs to provide:

  - two data type as template parameters:

    - ProblemType: input problem type
    - ResultType: output result type

  - the input as an object and the object where the final result has to be stored

  - A number of functions that implement::

    - the divide operation
    - the combine operation
    - the base case operation
    - the test condition operation

# Divide and Conquer (D&C)

- Let's have a quick look at the interface contained in the file

    <fastflow-home>/ff/dc.hpp


- Fibonacci example

    <fastflow-home>/examples/fibonacci/fib_dac.cpp

# ClassWork6

- Implement the Merge Sort algorithm for a std::vector of integer values.

- Compare the performances for small and large vector sizes when varying the number of "worker" threads.