

## Contenido para la interrogación 2 de IMT2112 semestre 2019-2

A la continuación hay una descripción del contenido para la interrogación 2 del curso IMT2112 Algoritmos Paralelos en Computación Científica, el jueves 24 de octubre de 2019.

1. Todos los apuntes de clases 1 hasta 20; disponible en Siding.
2. Todos los materiales de las ayudantías 1 hasta 9; disponible en Github.
3. Las tareas 1, 2 y 3; disponible en Siding.
4. Libro: Victor Eijkhout (2019), "Introduction to High Performance Scientific Computing." (<http://www.tacc.utexas.edu/~eijkhout/istc/istc.html>)
  - Todo el contenido especificado para la interrogación 1
  - 2.6.1 Parallel programming
    - Conceptos importantes: thread, heap, stack, private/shared variables, fork-join mechanism, race condition, thread safety
    - Fuera del contenido: secciones 2.6.1.3, 2.6.1.7 y 2.6.1.8
  - 2.6.2 OpenMP
    - Conceptos importantes: OpenMP, task scheduling
  - 2.6.3 Distributed memory programming through message passing
    - Conceptos importantes: the SPMD model, deadlocks, blocking and non-blocking communication, one-sided and two-sided communication, the MPI library
  - 2.6.4 Hybrid shared/distributed memory computing
  - 4.2 Finite difference methods
    - Conceptos importantes: stencil notation, discretization matrix
  - 5.4.1 Storage of sparse matrices
    - Conceptos importantes: compressed row storage, diagonal storage
  - 6.5 Parallel sparse matrix-vector product
  - 6.6 Parallel iterative linear solvers
    - Outside content: 6.6.2.
  - 6.7 Parallel preconditioners
    - Conceptos importantes: Jacobi preconditioning, ILU preconditioning, block ILU preconditioning
  - 6.8.2 Variable reordering and colouring
    - Conceptos importantes: red-black colouring, multi-colour parallel ILU
5. Libro: Victor Eijkhout (2017), "Parallel Programming in MPI and OpenMP." (<https://bitbucket.org/VictorEijkhout/parallel-computing-book/src>)
  - 1 Getting started with MPI
    - Important concepts: the data model, MPI library, language bindings
  - 2.1 The SPMD model
  - 2.2 Starting and running MPI processes
  - 2.3 Processor identification
  - 3.1 Working with global information
  - 3.2 Reduction
  - 3.3 Rooted collectives: broadcast and reduce

- 3.4 Rooted collectives: gather and scatter
- 3.7 Barrier
- 3.11 Non-blocking collectives
- 4.1 Distributed computing and distributed data
- 4.2 Blocking point-to-point operations
- 4.3 Non-blocking point-to-point operations
- 6.1 Communicator basics
- 6.2 Subcommunications

Observaciones sobre los ejemplos de código en el libro

1. No es necesario entender el syntax de Fortran o Python, sólo C++.
2. No es necesario recordar todas las funciones. Sin embargo, dado un ejemplo de un código o la descripción de una función, deben ser capaz de usar esta información y aplicar o un código corto.