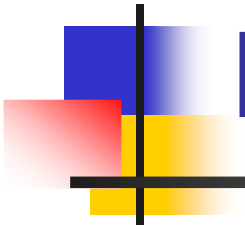
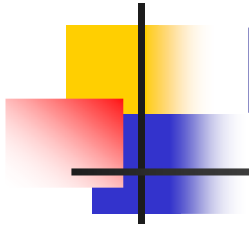


Introduction to Parallel Computing

Alexandre David

B2-206

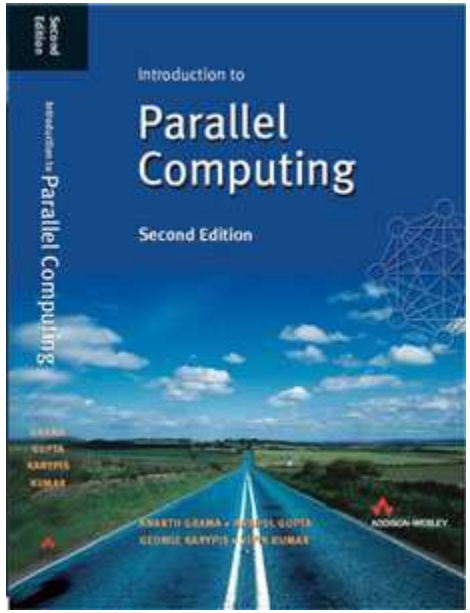
http://www.cs.aau.dk/~adavid/teaching/MTP-06/



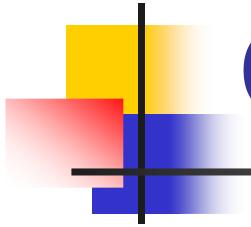
Presentation of the Course

- Parallel Computing
  - Little on parallel hardware
  - Mostly on parallel algorithms and design
- Models for Parallelism (PRAM…)
- Tools for Parallelism (MPI, pthreads…)
- 15 lectures, 2x45 min + exercises
- TA: Martin Clemmensen clemme@cs.aau.dk

The Course Book

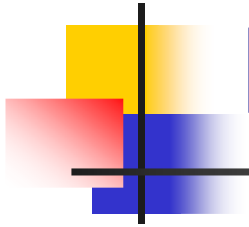


- *Introduction to Parallel Computing*
- Covers many aspects on parallel computing.
- Both basic and advanced topics.
- I will follow the book but not cover it all.



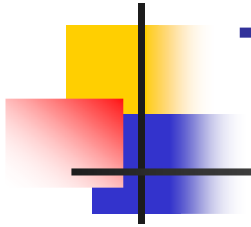
Goals of the Course

- Design, analysis, and implementation of parallel algorithms.
  - Principles of parallel algorithm design.
  - Analytical modeling of parallel programs.
  - Tools such as MPI & pthreads.
  - Some examples.



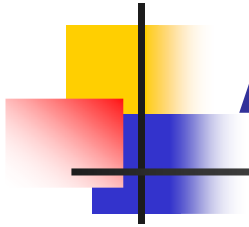
Do We Need Parallelism?

- Complexity of parallel programs: How to *specify* and *coordinate* **concurrent** tasks?
- Question of portable algorithm standards?
- Need to accelerate applications?



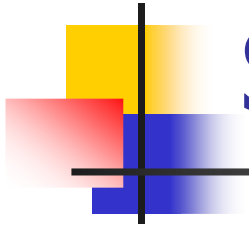
Trends in Hardware

- Everything points towards parallelism from multi-core, hyperthreading, multi-threads, superscalar, … technologies.
- Because
  - Limits to continue to increment performance with uniprocessors.
  - Other constraints like heat, complexity, yields, etc…



Arguments for Parallelism

- Computational power:
  - Moore's law.
  - Translating transistors into useful OPS.
- Memory/disk speed:
  - Performance/yr: CPU +40%, DRAM +10%.
  - How to feed data?
- Parallel platforms: larger aggregate cache+bandwidth+IPC…



Scope of Parallel Algorithms

- Engineering & design.
- Scientific computing.
- Commercial (web) applications.
- Embedded systems.
- Gaming industry.