Project

One of:

- Pick a sizable parallel problem to implement
  - Ok to form a team of 2 students
- Pick a new language and implement past exercises
- Present a new language for parallelism in class

Programming Project Artifacts

- Description of the problem
  - Include an an explanation of expected speedup (i.e., parallel versus inherently sequential)
- What you expected to learn and did learn from the project
  - Must not be so generic that it would apply to any HW
  - Must address *learning* about parallelism
- Implementation
- Measured speedup for implementation
  - Document measurements: platform, *P*, etc.
  - Speedup must be > **1**

Due Dates

Due Friday, December 5:

- Description of the problem
- What you expect to learn from the project

Due Friday, December 12:

- All 4 parts of the completed project — even the parts already submitted

Project Ideas from the Book

Implement existing parallel algorithms

e.g., Batcher's Sort

Re-implement existing parallel benchmarks

Parallelize some useful computation

Chess-End Games	3-Satisfiability Problems	Gene Sequence Alignment
Segmented Least-Squares	Video Motion Detection	A* Path-Finding for Games
Audio Analysis with a GPU	KD-Tree Construction	Image Convolution
Exact String Matching	MP3 Fast Fourier Transformation	Boid Simulation
Kohonen Maps	Ray Tracing	Galaxy Simulation
Prime Factorization	Rectangular Partitioning	Kenser-Ney Smoothing
Data Encryption	Checkers Min/Max Search	Artificial Neural Nets
Ray Casting	Julia Sets	Constraint Satisfaction
Sample Sort	Traveling Salesman Problem	Collaborative Filtering