Project Proposal (due 3/8)

- Team of 2-3 people
  - Please let me know if you need a partner
- Proposal Logistics:
  - Significant implementation, worth 50% of grade
  - Each person turns in the proposal (should be same as other team members)
- Proposal:
  - 3-4 page document (11pt, single-spaced)
  - Submit with handin program:

    “handin CS6235 prop <pdf-file>”

Project Parts (Total = 50%)

- Proposal (5%)
  - Short written document, next few slides
- Design Review (10%)
  - Oral, in-class presentation 2 weeks before end
- Presentation and Poster (15%)
  - Poster session last week of class, dry run week before
- Final Report (20%)
  - Due during finals – no final for this class

Project Schedule

- Thursday, March 8, Proposals due
- Monday, April 2, Design Reviews
- Wednesday, April 18, Poster Dry Run
- Monday, April 23, In-Class Poster Presentation
- Wednesday, April 25, Guest Speaker

Content of Proposal

I. Team members: Name and a sentence on expertise for each member

II. Problem description

- What is the computation and why is it important?
- Abstraction of computation: equations, graphic or pseudo-code, no more than 1 page

III. Suitability for GPU acceleration

- Amdahl's Law: describe the inherent parallelism. Argue that it is close to 100% of computation. Use measurements from CPU execution of computation if possible.
- Synchronization and Communication: Discuss what data structures may need to be protected by synchronization, or communication through host.
- Copy Overhead: Discuss the data footprint and anticipated cost of copying to/from host memory.

IV. Intellectual Challenges

- Generally, what makes this computation worthy of a project?
- Point to any difficulties you anticipate at present in achieving high speedup

Projects – How to Approach

- Some questions:
  1. Amdahl's Law: target bulk of computation

     and can profile to obtain key computations…
  2. Strategy for gradually adding GPU execution to CPU code while maintaining correctness
  3. How to partition data & computation to avoid synchronization?
  4. What types of floating point operations and accuracy requirements?
  5. How to manage copy overhead? Can you overlap computation and copying?