This is HNot the title.
it can be split on several lines.

RaymonHIIIII

May 24, 2010

Lecture 1

Levels of abstraction

- **Application Level:** February 8, 2011Systematic procedure for solving the problem is developed, based on some discrete data domain. In many cases, the problem itself cannot be solved directly, so have to solve a related problem, this is often discrete and reflects the discrete nature of computer storage. Parallelism arises in two forms, **data-parallelism** and **task-parallelism**.

- **Specification Level:** Here the algorithm is expressed as a program, moving concerns further towards the restrictions associated with real computer storage. Programming languages (which reflects the hardware architecture) often leads to unnatural constraints on the algorithm. See notes on how MPI forced the parallel algorithm in a certain direction. [1]

- **Algorithm Level:**

- **Program Level:**

- **Architecture Level:**

		Primes				
		2	3	5	7	
Powers	504	3	2	0	1	
	540	2	3	1	0	
Powers	gcd	2	2	0	0	min
	lcm	3	3	1	1	max

Table 1: Results for bilinear (Q_2) approximation, where $u(0,0) = 0.298393205714$.

Lecture 2

Lecture 3

Lecture 4

Lecture 5

Lecture 6

Lecture 7

Lecture 8

Lecture 9

Lecture 10

Lecture 11

Lecture 12

Lecture 13

Lecture 14

References

[1] Daniel J. Duffy. Finite difference methods in financial engineering: A partial differential equation approach, March 2006.