Lecture #1, day 1, 1/12/98

1 Matrix Multiplication

1.1 Naive method

The naive method for doing matrix multiplication is the method typically taught in math class. Multipy 2 matrices $A * B = C$ by setting $C_{ij} = \sum_{k=1}^{n} A_{ik} B_{kj}$. The number of multiplications and additions is $O(n^3)$

1.2 Slightly less Naive method

Divide the matrices into submatrices, multiply the submatrices, then recombine the resulting matrices.

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} * \begin{bmatrix} E & F \\ G & H \end{bmatrix} = \begin{bmatrix} AE + BG & AF + BH \\ CE + DG & CF + DH \end{bmatrix}$$

- avoids thrashing the cache, but otherwise has the same time complexity

1.3 Strassen's algorithm

Strassen's algorithm starts by subdividing the matrices in the same manner as the slightly less naive method, but then uses a different series of manipulations to construct the result matrix with only 7 recursive calls, instead of the 8 used in the slightly less naive method. Complexities:

- Addition: $A(n) = 18\frac{n^2}{4} + 7A(\frac{n}{2})$ with $A(1) = 0$. Solution: $A(n) = 6n^{\log_2 7}$
- Multiplication: $M(n) = 7M(\frac{n}{2})$ with $M(1) = 1$. Solution: $M(n) = n^{\log_2 7}$

One important note is that the constant 6 in $A(n)$ for Strassen's algorithm is much larger than the constant 1 in $A(n)$ for the naive algorithms. This means there is some size of matrix s.t. forall larger matrix multiplications strassen should be used and for all smaller matrices, one of the naive methods should be used.