

Due at the start of class Friday, June 12, 2009.

Problem 1. Assume your machine has 32 bit words. Assume you can multiply two n word numbers in time $3n^2$ with a standard algorithm. Assume you can multiply two n word numbers in time $20n^{\lg 3}$ with a “fancy” algorithm.

- (a) Approximately, how large does n have to be for the fancy algorithm to be better?
- (b) How many bits is that?
- (c) How many decimal digits is that?

Problem 2. Use the same assumptions as for problem (1), except assume you can multiply two n word numbers in time only $10n^{\lg 3}$ with a “fancy” algorithm.

- (a) Approximately, how large does n have to be for the fancy algorithm to be better?
- (b) How many bits is that?
- (c) How many decimal digits is that?

Problem 3. Selection Sort can be thought of as a recursive algorithm as follows: Find the largest element and put it at the end of the list (to be sorted). Recursively sort the remaining elements.

- (a) Write down the recursive version of Selection Sort in psuedocode.
- (b) Derive a recurrence for the exact number of comparisons the algorithm uses.
- (c) Use the iteration method to solve the recurrence. Simplify as much as possible.
- (d) Use mathematical induction to verify your solution.