Put your *name* and *section number* on your solution and (if more than one sheet) staple. Section 3 of the NP-completeness homework will be due Friday, October 7.

1. Assume your machine has 64 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 $10n^{\lg 3}$ 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?

2. Assume that we would like to multiply a two-digit number ab with a three-digit number cde. The standard algorithm would do six atomic multiplications. Explain how you can do fewer atomic multiplications by forming the product $(a+b)(c+d+e)$. How many atomic multipications do you use?

3. Consider an array of size nine with the numbers in the following order $50, 30, 10, 60, 80, 50, 90, 70, 20$.
   (a) Form the heap using the algorithm described in class. Show the heap as a tree. Show the heap as an array. Exactly how many comparisons did heap creation use?
   (b) Start with the heap created in Part (a). Show the *array* after each element sifts down *after heap creation*. How many comparisons does each sift use? What is the total number of comparisons *after heap creation*?

4. Assume that we start with a random array of size $n = 2^k - 1$ and form a heap.
   (a) What is the probability that the third largest element will be a child of the root? Justify.
   (b) Before you form a heap, you notice that none of the three smallest elements are near the top of the array, or more formally none of them are in any of the first $(n-3)/4$ locations of the array. What is the probability that the third smallest element will be the parent of a leaf? Justify.