COMPLEXITY: Exercise No. 1
due next week

1. Prove or disprove:

   (a) $f(n) = O(n) \Rightarrow 2^{f(n)} = O(2^n)$.

   (b) If $f = o(g)$ then there is h such that $f = o(h)$ and $h = o(g)$.

   (c) If $f_1, f_2, \ldots, f_8 \in O(n)$ then $\sum_{i=1}^{8} f_i(n) = O(n)$.

   (d) If $f_1, f_2, \ldots \in O(n)$ then $\sum_{i=1}^{n} f_i(n) = O(n^2)$.

   (e) There is a function f such that $f(n) = O(n^{1+\epsilon})$ for every $\epsilon > 0$ but $f(n) = \omega(n)$.

   (f) If f is monotone positive function, $f(n) = O(n)$ and $f(n) \neq o(n)$ then $f(n) = \Theta(n)$.

2. Describe explicitly (i.e., write down the transisition function δ) a 1-tape DTM with tape alphabet $\Sigma = (0, 1, \sqcup, \triangleright)$ which accepts all words where the number of 1's is at most three times the number of 0's and rejects all remaining words.

3. Describe explicitly an $O(\log k)$-space and $O(k)$-time 3-tape DTM that given a number k in binary encoding, it writes 1^k (i.e. k ones) on the output tape.

4. Sketch a DTM that accepts L_{pal} (the language of palindromes) with space $O(\log n)$ and time $O(n^2/\log n)$.

5. Let M be a k-tape DTM with input and output which always stops with time complexity $t(n)$ and space complexity $s(n)$. Prove that there is a constant c (which depends on M) such that $t(n) \leq nc^{s(n)}$ (assume $s(n) > 0$ for all n).