به نام او

Quiz 6

Question:

Prove the followings:

a) If an algorithm makes at most a constant number of calls to polynomial-time subroutines and performs an additional amount of work that also takes polynomial time, then it runs in polynomial time.

b) A polynomial number of calls to polynomial-time subroutines may result in an exponential-time algorithm.

Answer (100pts): Part a (40 pts), part b (60 pts)

a) If subroutines are called k times (a constant) plus an additional polynomial amount of work, the time complexity is at most
$\mathcal{O}\left(\left(((n^{d_1})^{d_2})\dots\right)^{d_k} + n^c\right) = \mathcal{O}(n^{d_1+d_2+\cdots+d_k} + n^c) = \mathcal{O}\left(n^{k\times\max(d_i)} + n^c\right).$

This is still polynomial time.

b) When k is not constant, it becomes a function of the input size and hence $\mathcal{O}\left(n^{k\times\max(d_i)} + n^c\right)$ is not polynomial. To understand this better, suppose each subroutine simply doubles the size of its input in its output. Then, then the algorithm runtime would be at least $\mathcal{O}(2^k)$.