CPSC 511 — Fall 2014
Solutions for Question #4 on Midterm Test

In this question you were asked to prove that $\mathcal{NP} \subseteq$ EXPTIME.

Solution: Recall that a language $L \subseteq \Sigma^\star$ is in $\mathcal{NP}$ if and only if there exists an alphabet $\Sigma_C^\star$ used to encode ***certificates*** and a polynomial time ***verifier*** for L — that is, a deterministic Turing machine

$$M_V = (Q, \Sigma_V, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})$$

where $\# \notin \Sigma \cup \Sigma_C$, $\Sigma_V = \Sigma \cup \Sigma_C \cup \{\#\}$, and M_V satisfies the following additional properties:

(a) When executed on an input $\mu\#\nu$ such that $\mu \in \Sigma^\star$ and $\nu \in \Sigma_C^\star$, M_V ***halts*** after a number of steps that is at most polynomial in the length of μ (rather than in the length of the longer string $\mu\#\nu$) in the worst case.

In particular (increasing the value of the polynomial function to simplify it, if necessary) we may assume there there are integer constants c_0, c_1 and c_2 such that M_V halts on an input $\mu\#\nu$, as above, after at most $c_0|\mu|^{c_1} + c_2$ steps in the worst case.

(b) If $\mu \in L$ then there exists a string $\nu \in \Sigma_C^\star$ such that M_V accepts $\mu\#\nu$.

Note: There will also exist a string $\nu \in \Sigma_C^\star$ such that that M_V accepts $\mu\#\nu$ and the length of ν is at most $c_0|\mu|^{c_1} + c_2$ — because it follows from the above that M_V never reads past the prefix of ν with this length. Thus deleting the rest of ν (if ν is longer than this) will not change the behaviour of M_V.

(c) On the other hand, if $\mu \in \Sigma^\star$ and $\mu \notin L$, then M_V ***rejects*** $\mu\#\nu$ for very string $\nu \in \Sigma_C^\star$.

Note that if $L = c_0|\mu|^{c_1} + c_2$ then the number of strings $\Sigma_C^\star$ with length at most L is at most

$$L + 1$$

if $|\Sigma_C| = 1$, or at most

$$\sum_{i=0}^{L} |\Sigma_C|^i = \frac{|\Sigma_C|^{L+1} - 1}{|\Sigma_C| - 1}$$

if $|\Sigma_C| \geq 2$. In either case, since $|\Sigma_C|$ is an integer constant (for any choice of the language $L \in \mathcal{NP}$), there exists another integer constant k such that the above value is in $O(2^{|\mu|^k})$ — that is, this value is (only) ***exponential*** in $|\mu|$.

An exponential-time deterministic Turing machine (or algorithm) can now be used to decide membership of an input string $\mu \in \Sigma^\star$ in L, by carrying out a brute force search of all possible certificates with length at most L, using the polynomial-time verification algorithm M_V to check every possible certificate:

On input $\mu \in \Sigma^\star$:

1. `L :=` $c_0|\mu|^{c_1} + c_2$
2. `for` (every string $\nu \in \Sigma_C^\star$ such that $|\nu| \leq L$) `{`
3. `  if` (M_V accepts $\mu\#\nu$) `{`
4. `    `***accept***
   `  }`
   `}`
5. ***reject***

The algorithm terminates after a number of steps that is exponential in $|\nu|$ — indeed, using a number of steps that is in $O(|\mu|^{k+1})$ for k as described above. It should reasonably clear that this algorithm accepts μ if $\mu \in L$ and that it rejects it otherwise — as needed so show that $L \in$ EXPTIME.