Assignment 4 Solutions **CAS705 Fall 2005** **Due on November 7th**

1. For part (a), use essentially the same machine that decides L in time n^6 to decide $\text{pad}(L, n^2)$ in time n^3 by making it "ignore" the trailing junk[1] (note that $|\text{pad}(x, |x|^2)| = |x|^2 = m$, and x is decided in time $|x|^6 = m^3$). For part (b) prove the contrapositive: assume $\mathbf{P} = \mathbf{NP}$, and suppose that L is in **NEXPTIME**. Then $\text{pad}(L, 2^{n^k})$ is in **NP**, and so it is in **P**, and hence L is **EXPTIME**.

2. Same idea as the **Space Hierarchy Theorem** done in class. This proof is technically more difficult as it has to use a counter to keep track of time, and managing the counter takes a little bit of time as well .... Hence we lose the $\log(n)$ factor.

3. For any oracle A define the language $L(A) = \{x| \ \{0,1\}^{|x|} \cap A = \emptyset \ \}$. Note that an **NP** machine with oracle A can decide $\overline{L(A)}$ by guessing y of length $|x|$ and checking if $y \in A$. So, a **coNP** machine with oracle A can decide $L(A)$. Now construct A iteratively so that no oracle **NP** machine can decide $L(A)$: let $M_1^?, M_2^?, M_3^?, \ldots$ be a list of all **NP** Turing machine where $M_i^?$ takes n^i many steps on inputs of length n. At step 0, $A_0 = \emptyset$, and at step i, we choose an n such that all strings examined (for membership in A) so far are of length $< n$ and furthermore, $n^i < 2^n$. Simulate $M_i^?$ on input 0^n, and each time $M_i^?$ queries the oracle with some w, if w's membership in A has already been established, answer accordingly, and otherwise declare w *not* in A. If at the end $M_i^?$ rejects (all paths are rejecting), declare the remaining $\{0,1\}^n$ *not* in A, and if $M_i^?$ accepts, find an accepting path and some $y \in \{0,1\}^n$ not queried on it, and declare it in A.

4. **Nepomnjascij's Theorem:** Let $0 < \varepsilon < 1$ be a rational number, and let a be a positive integer. Then,
$$\text{NTIMESPACE}(n^a, n^\varepsilon) \subseteq \mathbf{LTH}$$
Suppose M is a nondeterministic TM running in time n^a and space n^ε. Then, M accepts an input x iff
$$\exists \mathbf{y}[\ \mathbf{y} \text{ represents an accepting computation for } x\ ] \tag{1}$$
So $\mathbf{y} = y_1 y_2 \ldots y_{n^a}$ where each $|y_i| = n^\varepsilon$, and so $|\mathbf{y}| = n^{a+\varepsilon}$. So $\mathbf{y}$ is too long to verify in linear time (in $n = |x|$). So we use a trick: let $\mathbf{z} = z_1 z_2 \ldots z_{n^{1-\varepsilon}}$ where the z_i represent every $(n^{a-1+\varepsilon})$-th string in $\mathbf{y}$. So now, (1) can be restated as follows:
$$(\exists \mathbf{z})(\forall i)(\exists \mathbf{u})[\ \mathbf{u} \text{ shows } z_{i+1} \text{ follows from } z_i \text{ in } n^{a-1+\varepsilon} \text{ steps } \& \ z_n \text{ is accepting }]$$
Note that $|\mathbf{z}| = n^{1-\varepsilon} n^\varepsilon = n$, so this is OK, but $|\mathbf{u}| = n^{a-1+\varepsilon} n^\varepsilon = n^{a-1+2\varepsilon}$. But note that $(a - 1 + 2\varepsilon) < (a + \varepsilon)$ because $0 < \varepsilon < 1$. So we have reduced $\mathbf{u}$ with respect to $\mathbf{y}$, by a factor of $n^{1-\varepsilon}$, so to know how many times we have to repeat the above nesting of quantifiers, we solve for i in the following equation:
$$\frac{n^{a+\varepsilon}}{(n^{1-\varepsilon})^i} = n$$
and solving, we get $i = (a + \varepsilon)/(1 - \varepsilon)$.

[1] Note quite ignore, as it has to check that the input string $w = \text{pad}(x, |x|^2)$, and reject if not.