CS 1511 Homework 9

Mathew Varughese, Justin Kramer, Zach Smith

Monday, Feb 18

16. We wish to show that if a Turing Machine S accepts B in LOGSPACE then there is a Turing Machine U that accepts A in LOGSPACE.

We know that we have a Turing Machine T that takes an input so that $x \in A \iff T(x) \in B$

The idea of U is to run the output of T on S and if S accepts, U accepts. Otherwise it rejects. However, simulating T would take linear space since it's output takes linear space. It is linear in its output for the following reason.

The output space of T is linearly bounded by the length of the input (I). This is because T is LOGSPACE. So, it has a total of x maximum positions it can move on the tape. (Say length of alphabet is 2, then there are $log_2|I|$ spots, so $2^{log_2|I|}$ possibilites, which simplifies to $|I|$).

We construct U as follows:

U takes in input x. U also has a counter variable. This counter variable will take log space to hold. Run S, and whenever S needs to read a value from the output of T store the position of T's output it needs to read in the counter variable. Then rerun T until this space is outputted and generated and use that value on S. Repeat this process until S is finished. Then U accepts if S accepts and reject otherwise.

17. a) A language C is complete for EXPSPACE iff

1) $C \in EXPSPACE$ and

2) $\forall L \in EXPSPACE, \quad L \leq_r C$

Let C' be an arbitrary language in EXPSPACE.

$\exists\, TM\ T \quad \exists k$ such that

- $\forall I\ T$ halts on I using space $\leq 2^{|I|^k}$.
- $\forall I\ T$ accepts I iff $I \in C'$.

First we prove part 2. We have a language C. $C = \{(I, T, 2^{|I|^k}) \mid T \text{ accepts } I \text{ in } \leq 2^{|I|^k} \text{ space}\}$

There exists a turing machine TM_C that decides C.

For every L in EXPSPACE, there exists a TM (T') that performs the reduction $L \leq_r C$ in poly time with (I, T, k) as an input.

For any language L, the Turing Machine T' would return the value of TM_C with input $(I, T, 2^{|I|^k})$.

This TM_T and integer k would be baked into the algorithm for T'.

The encoding of k into $2^{|I|^k}$ will only take polynomial time.

Simulating TM_C with a decider TM_M will take $2^{I,T,2^{|I|^k}}$ all raised to k' space. In all cases of k', this simulation will always be in EXPSPACE, showing that TM_C is in EXPSPACE. This is because TM_M will always have a maximum space requirement that is equal to or greater than our space requirement of $2^{|I|^k}$ space.

17. b) Our proof will be exactly the same as above, except with one change.

Our language C will now be defined as such:

$C = \{(I, T, c^{|I|^k}) \mid T \text{ accepts } I \text{ in } \leq c^{|I|^k} \text{ space } \}$

The encoding of k into $c^{|I|^k}$ will still only take polynomial time and our simulation will always take a maximum space requirement that is equal to or greater than our space requirement of $c^{|I|^k}$ space.