

# 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 its 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  $2^{\log_2 |I|}$  possibilities, 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)  $C \in \text{EXPSPACE}$  and
- 2)  $\forall L \in \text{EXPSPACE}, L \leq_r C$

Let  $C'$  be an arbitrary language in EXPSPACE.

$\exists TM T \exists k$  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  $C$ .  $C = \{(I, T, k) \mid T \text{ accepts } I \text{ in } \leq 2^{|I|^k} \text{ space}\}$ . 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.

For any language  $L$ , the Turing Machine  $T'$  would return the value of  $TM_C$  with input

$(I, T, k)$ .

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

Second we prove part 1. We need to show that  $C \in EXPSPACE$ .

There exists a  $TM_C$  that decides  $C$ .

This  $TM_C$  takes an input  $(I, T, k)$ .

This machine will either halt on accept or reject, or halt and reject if it runs for more than  $2^{|I|^k}$  steps.

**17. b)** A language  $C$  is complete for  $EXPSPACE$  iff

1)  $C \in EXPSPACE$  and

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

Let  $C'$  be an arbitrary language in  $EXPSPACE$ .

$\exists TM T \exists k$  such that

-  $\forall I T$  halts on  $I$  using space  $\leq (c/2)^{|I|^k} * 2^{|I|^k}$ .

-  $\forall I T$  accepts  $I$  iff  $I \in C'$ .

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

Now that we have established  $(c/2)^{|I|^k}$  as some constant factor being multiplied to  $2^{|I|^k}$ , we can dismiss the  $c$  since this constant multiplier will not affect it being in  $EXPSPACE$ .

The proof will then follow the same path as 17 (a).