Homework 10

Joe Baker, Brett Schreiber, Brian Knotten

February 4, 2018

14

Let J_i be the i^{th} binary-encoded Java program, and MJ_i be the i^{th} binary-encoded mini-Java program.

Define TM D: For an input x, run $MJ_x(x)$ and return the opposite. (Note that all MJ programs halt on all inputs).

The language accepted by D cannot be accepted by any MJ_i by its construction. Any MJ_i will return the opposite of D on input i.

Since mini-java is a subset of Java, we can build a Java J program that: For an input x, run $MJ_x(x)$ and returns the opposite. Then J accepts the language that no MJ_i can accept.

15

a

A TM M can be defined as follows so that $L(M) = A$:
On input x:
    Instantiate $c = 0$ on the working tape.
    for each character $x_i \in x$:
        if $x_i =$ '(', then increment c.
        else, if $x_i =$ ')', then decrement c.
        if $c < 0$ reject. (There is a right paren before a left one).
    Accept if an only if the final value of $c = 0$.
$L(M) = A$ because $L(M)$ only accepts when the number of left parentheses matches the number of right parentheses. M runs in logspace because the in the worst case, the input x to M will be n number of left parentheses. So the working tape has to count up to n. But by using the standard base-2 binary encoding of n, the working tape will only use a maximum of $\log(n)$ cells.

b

A TM N can be defined as follows so that $L(N) = B$:
Assume N has two work tapes.
On input x:
    First pass: For each character x_i, x_{i+1} in x:
        If $x_i, x_{i+1} =$ '(', ']', reject,
        or if $x_i, x_{i+1} =$ '[', ')', reject.

  Set $c_1 = 0$ on the first working tape.
Set $c_2 = 0$ on the second working tape.
    Second pass: For each character x_i in x:
        If $x =$ '(', increment c_1
        or if $x =$ ')', decrement c_1
        or if $x =$ '[', increment c_2

or if $x =$ ']', decrement c_2
If $c_1 < 0$ or $c_2 < 0$, then immediately reject.
If $c_1 = 0$ and $c_2 = 0$, then accept. Otherwise, reject.

N runs in logspace, even though it uses two work tapes, since $2\log(n) = O(\log(n))$.