

HOMEWORK 2, CPSC 421-101, FALL 2015

JOEL FRIEDMAN

Copyright: Copyright Joel Friedman 2015. Not to be copied, used, or revised without explicit written permission from the copyright owner.

1. Exercise 7.2.6 of the handout (you have to prove this “from scratch,” i.e., without relying on any theorem we have learned; you can adapt a proof that we have seen).

2. Describe a Turing machine that decides the language

$$\{x \in \{0, 1\}^* \mid x \text{ has an odd number of 0's}\},$$

i.e., the language consisting of strings over the alphabet $\{0, 1\}$ that have an odd number of zeros (for example, 1011 and 000). You should give: (1) a high level description (the rough algorithm), (2) an implementation level description (describe the set of states, the alphabets, and the tape movement), and finally (3) a formal description (an entire description of the δ function).

3. Describe a Turing machine that decides the language

$$\{x \in \{0, 1\}^* \mid x \text{ has more 0's than 1's}\}.$$

You should give: (1) a high level description (the rough algorithm), (2) an implementation level description (describe the set of states, the alphabets, and the tape movement), and finally (3) a formal description (an entire description of the δ function).

4. Give a high level description (the rough algorithm) and an implementation level description (describe how things are organized on the work tape, some rough idea about what the state set will look like) of a Turing machine that decides the language

$$\{x \in \{0, 1\}^* \mid x \text{ in binary represents a prime number}\}.$$

DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, BC V6T 1Z4, CANADA, AND DEPARTMENT OF MATHEMATICS, UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, BC V6T 1Z2, CANADA.

E-mail address: jf@cs.ubc.ca or jf@math.ubc.ca

URL: <http://www.math.ubc.ca/~jf>