

## CS215 ASSIGNMENT 1

Problems 1 and 2 are due Tuesday, October 5

Problem 3 is for self-test and next proofs group meeting

**Problem 1.** Design a TM for a language given below. Give a high-level description of the algorithm (preferably in pseudocode, but plain English is ok too), and the complete transition function. The transition function should be submitted in the format defined in the next problem, or, if you prefer, drawn using a state diagram.

If your *first name* starts with A-L, design a TM for  $L_1$ , otherwise design a TM for  $L_2$ .

$$\begin{aligned}L_1 &= \{0^i 1^j : i, j \geq 2 \text{ \& } i, j \text{ are relatively prime}\} \\L_2 &= \{0^i 1^j : j = i^i\}\end{aligned}$$

**Problem 2.** Write a C++ program that simulates a given Turing Machine (and use it to test your TM from Problem 1). The program is called with two arguments given on the command line: the name of the file with the description of the Turing Machine, and the input string. The program prints one of the following messages: **Accept**, **Reject**, **Invalid TM**, or **Invalid input** (an input string is invalid if it contains symbols that do not belong to the input alphabet).

Assume that the input alphabet is  $\Sigma = \{0, 1\}$ , and that the states are alpha-numeric words (words using only non-whitespace, printable characters). Let “accept” be the accept state and “reject” be the reject state. The tape symbols (other than the input symbols) are  $\_$ , A, B, C ..., Z. (symbol  $\_$  denotes the blank). The transition function is given by a sequence of instructions, one per line (empty lines are allowed). Each instruction has the following format:

```
q a p b m      <comment>
```

where  $q, p$  are states,  $a, b$  are tape symbols, and  $m \in \{L, R\}$  is the head move. The rest of the line can contain comments. Leading spaces or tabs are allowed. (For readability, you *must* use indentation to format loops.) There is no end marker, the instructions end with the eof. When the transition function is not specified for given arguments (state, tape symbol), assume that the TM enters the rejecting state.

Assume that the machine is deterministic. If more than one transition is specified for a given pair (state, tape symbol), the program must print an error message.

**Problem 3.** (don't turn in for grading) From text: exercises 3.2 and 3.8, problems 3.13, 3.14.

**Collection:** Turn in both the TM description and the TM simulator by email no later than Tuesday, Jan. 12, 10AM. Send me the following files: a pdf (or other) file for Problem 1, the TM description for Problem 1 (in the format specified in Problem 2), the source code and the makefile for Problem 2. Send all files as email attachments. You also need to turn in the hardcopies (except for the makefile).

*Important:* Make sure to read the assignment carefully and to follow exactly the program specification and the turn-in instructions. Points will be taken off for homeworks that deviate from the homework instructions. Principles of good programming apply, both to the simulator and to the TM code.