NTIN071 A&G: Tutorial 11 – Turing machines

Solve 1, 2, 3a-d, 4 first (the rest is for practice).

Problem 1 (A Turing machine)**.** Consider the following TM.

	B	a	b	c
$\to q_0$	(q_1, B, L)	(q_0, a, R)	(q_0, c, R)	(q_0, c, R)
q_1	(q_2, B, R)	(q_1, c, L)		(q_1, b, L)
$*q_2$				

(a) Draw the state diagram.
(b) Describe the computation (by a sequence of configurations) for $w = aabca$.
(c) Describe the operation it performs.
(d) What is the language recognized by the machine?

Problem 2 (Erase all 1s)**.** Design a TM over the alphabet $\{0, 1\}$ which will erase all 1's from the input and then return to the beginning (e.g. if it starts in the configuration $q_0 0011010$, then it will halt in the configuration $q_F 0000$ for some $q_F \in F$).

Problem 3 (Programming TMs)**.** Design a TM which will accept the language L. Write down the sequence of configurations that shows that the given word w is accepted.

(a) $L = \{0^n 1^n \mid n \geq 0\}$, $w = 0011$
(b) $L = \{0^n 1^n 2^n \mid n \geq 0\}$, $w = 001122$
(c) $L = \{0^i 1^j \mid i \leq j\}$, $w = 00111$
(d) $L = \{w \in \{0, 1\}^* \mid |w|_0 = |w|_1\}$, $w = 100110$
(e) $L = \{ucu^R \mid u \in \{0, 1\}^*\}$, $w = 10c01$
(f) $L = \{uu^R \mid u \in \{0, 1\}^*\}$, $w = 101101$
(g) $L = \{ucu \mid u \in \{0, 1\}^*\}$, $w = 110c110$
(h) $L = \{uu \mid u \in \{0, 1\}^*\}$, $w = 110110$

Problem 4 (Predecessor)**.** Construct a Turing machine T that for a given input natural number $x > 0$ in binary encoding outputs its predecessor, i.e., $x - 1$ (in binary encoding as well) and returns the head to the beginning of the output.

(a) Draw the state diagram of T.
(b) Write a sequence of *configurations* that the machine goes through during some accepting computation for the input word $w = 10100$.

Construct a deterministic, single-tape, single-track machine. (If you want e.g. a two-track machine, program it yourself.) A number in binary encoding must not start with 0, unless it is equal to 0. Examples of input and output configurations:

- from the configuration $q_0 1$ the machine should finish in $f0$ for some $f \in F$,
- from the configuration $q_0 1001$ the machine should finish in $f1000$ for some $f \in F$,
- from the configuration $q_0 100$ the machine should finish in $f11$ for some $f \in F$.

Problem 5 (Reverse)**.** Design a TM which will create the reverse of the input word.

Problem 6 (Memory blocks)**.** Design a TM which will switch the contents of two memory blocks. Specifically, if it starts in the configuration $q_0u\#v\#w\#x\#y$ (where $u, v, w, x, y \in \Sigma \setminus \{\#\}$), then it halts in the configuration $fu\#x\#w\#v\#y$ for some $f \in F$. Try to construct a small and efficient machine.

Problem 7 (Nondeterministic test of non-primeness)**.** Design a nondeterministic TM which will accept the language $L = \{1^n \mid n \text{ is not a prime number}\}$.

Problem 8 (One-way infinite tape)**.** Describe how to convert a Turing machine with a (single) two-way infinite tape to a Turing machine whose tape is only infinite in one direction, to the right. (You can assume that the second TM's tape contains a special delimiter $\triangleright$ in its first field.)

Problem 9 (Head moves)**.** Consider modifications of Turing machines in which the allowed moves of the head are the following. What class of languages they recognize?

(a) left (L) and right (R),
(b) stay (N) and right (R),
(c) stay (N) and left (L),
(d) left (L), right (R), and stay (N).

Problem 10 (Only two actions at once)**.** Show that any single-tape Turing machine M can be converted to a Turing machine M' which is allowed to execute only two of the three actions at one step, that is, any instruction either

- changes state and head position, or
- changes state and tape symbol, or
- changes head position and tape symbol,

but no instruction can perform all three of these actions.

Problem 11 (Right or restart)**.** Consider a Turing machine model where the tape is only one-way infinite (to the right) and the head can only perform two types of movement: right (R) or RESTART (that is, return to the first field of the tape). Show how to convert a single-tape Turing machine to a Turing machine of this kind.

Problem 12 (Rewrite at most once)**.** Consider a single-tape Turing machine which is allowed to change any field (i.e., it can rewrite the symbol with a different symbol) on the tape at most once. Show that this model is equivalent to a regular single-tape TM.

Problem 13 (Don't rewrite input)**.** Explain why if a single-tape Turing machine is forbidden to modify the fields containing the input, it is equivalent to a finite automaton. (And therefore such TMs only recognize regular languages. It is enough to give the main idea, not a detailed construction.)

Problem 14 (Closure properties)**.** Show that both the class of all *decidable* languages and the class of all *partially decidable* languages are closed under:

(a) *union*, (b) *intersection*, (c) *concatenation*, (d) *Kleene star*.

Moreover, show that

(e) decidable languages are closed under *complementation*, but
(f) partially decidable languages are not.