

# Computational Complexity

## Class 2

### Turing Machines: Computation

**Exercise 1.** We consider a variant of Turing machines where the head can start in any place on the input word. A word is accepted if there exist at least one input position such that the word is accepted with the head starting in this position. Is this model equivalent to the original one?

**Exercise 2.** We call a configuration  $c$  of a Turing machine *reversible* if there is at most one configuration  $d$ , such that  $c$  is reached from  $d$  in one step. A Turing machine is said to be *weakly reversible* if any configuration reachable from some initial configuration is reversible. Note that if it is the case then we can trace the computation back. For a given Turing machine, construct a weakly reversible machine recognizing the same language. Estimate the time overhead in your construction.

### Turing Machines: Computable and Recursively Enumerable Sets

**Exercise 3.** Prove that if a language and its complement are recursively enumerable, then it is computable. (Turing-Post Theorem)

**Exercise 4.** (a) For a Turing machine, consider a mapping  $S_M : \{0, 1\}^* \rightarrow \mathbb{N} \cup \{\infty\}$ , where  $S_M(w)$  is the exact amount of space used by  $M$  for an input  $w$  (possibly infinity). Show that this function is in general not computable. On the other hand, if  $S_M(w)$  never takes the value  $\infty$  then it is computable.

- (b) If  $S : \mathbb{N} \rightarrow \mathbb{N}$  is any computable function then the set  $\{w \mid S_M(w) \leq S(w)\}$  is computable.
- (c) Formulate and prove analogous properties for time complexity.

**Exercise 5.** Show that the following conditions are equivalent for all non empty languages  $L$ :

- (a)  $L$  is recursively enumerable,
- (b)  $L$  is the domain of some partial computable function,
- (c)  $L$  is the range of some partial computable function,
- (d)  $L$  is the range of some (total) computable function.

**Exercise 6.** Show that a language is computable if and only if it is finite or is the range of some computable function that is monotone (with respect to lexicographic ordering).

**Exercise 7.** Let  $L$  be a computable set of words. Is the following language computable:

$\{w \mid \text{there is } v \in L \text{ such that } v \text{ and } w \text{ are equal up to at most two positions}\} ?$

**Exercise 8.** Are the following problems decidable, recursively enumerable ?

- (a) Does a given Turing machine over  $\{0, 1\}$  accept only words that do not contain any 1?
- (b) Does a given Turing machine over  $\{0, 1\}$  accept only words that contain a letter 1?
- (c) Does a given Turing machine over  $\{0, 1\}$  accept some word that contains the letter 1?

## Turing Machines: Complexity

**Exercise 9.** A function  $f : \mathbb{N} - \{0\} \rightarrow \mathbb{N}$  is *space constructible* if there is an off-line Turing machine which, for an input of length  $n \geq 1$ , writes in exactly  $f(n)$  cells of the auxiliary tapes. Show that the following functions are space constructible:  $n, 2n, n^2, n^2 + n, 2^n, \lceil \log_2(n) \rceil$ .

A function  $f : \mathbb{N} - \{0\} \rightarrow \mathbb{N}$  is *time constructible* if there is a Turing machine which, for an input of length  $n \geq 1$ , makes exactly  $f(n)$  steps and halts. Show that the following functions are time constructible:  $n, 2n, n^2, 2^n, 2^{2^n}$ .

**Exercise 10.** Assuming that natural numbers  $k, m, n$  are given in binary representations, estimate the time to compute  $m + n, m \bmod n, mn, mn \bmod k$ .

**Exercise 11.** Estimate the computation time of the Euclidean algorithm implemented on Turing machine.