COMP 3803 — Fall 2025 — Problem Set 5

Question 1: Construct a Turing machine with one tape, input alphabet $\Sigma = \{a, b, c\}$, and tape alphabet $\Gamma = \{a, b, c, \Box\}$ that accepts the language

$$\{a^m b^n c^{m+n} : m \geq 0, n \geq 0\}.$$

You may assume that the input string belongs to the language described by the regular expression $a^*b^*c^*$. At the start of the computation, the tape head is at the **leftmost** symbol of the input string. If the input string is empty, then the tape head is at an arbitrary position.

Start by explaining your algorithm in plain English, then mention the states that you are going to use, then explain the meaning of these states, and finally give the list of instructions.

Question 2: Construct a Turing machine with one tape and input alphabet $\Sigma = \{a\}$, that "doubles" a non-empty string of a's.

Start of the computation: The tape contains a string of the form a^n, for some $n \geq 1$, and the tape head is at the **leftmost** a. The Turing machine is in the start state.

End of the computation: The tape contains the string a^{2n} and its head is at the **rightmost** a. At the end, the Turing machine is in the final state.

The Turing machine in this question does not have an accept state or a reject state; instead, it has a final state. As soon as this final state is entered, the Turing machine terminates.

Start by explaining your algorithm in plain English, then mention the states that you are going to use, then explain the meaning of these states, and finally give the list of instructions.

Question 3: Construct a Turing machine with one tape, input alphabet $\Sigma = \{a, b\}$, and tape alphabet $\Gamma = \{a, b, \Box\}$ that accepts the language

$$\{w \in \{a, b\}^* : \ w \text{ contains the substring } aa\}.$$

At the start of the computation, the tape head is at the **leftmost** symbol of the input string. At termination, the tape head can be at an arbitrary cell.

Start by explaining your algorithm in plain English, then mention the states that you are going to use, then explain the meaning of these states, and finally give the list of instructions.

Question 4: The *complement* of a string $w \in \{a, b\}^*$ is the string w' obtained from w by replacing every a by b, and replacing every b by a. For example, the complement of *abba* is *baab*, and the complement of ε is ε.

Construct a Turing machine with one tape, input alphabet $\Sigma = \{a, b\}$, and tape alphabet $\Gamma = \{a, b, \Box\}$ that computes the complement of any input string.

Start of the computation: The tape contains a string w in $\{a, b\}^*$, and the tape head is at the **leftmost** symbol of w. The Turing machine is in the start state.

End of the computation: The tape contains the complement w' of w and its head is at the **leftmost** symbol of w'. At the end, the Turing machine is in the final state.

The Turing machine in this question does not have an accept state or a reject state; instead, it has a final state. As soon as this final state is entered, the Turing machine terminates.

Start by explaining your algorithm in plain English, then mention the states that you are going to use, then explain the meaning of these states, and finally give the list of instructions.

Question 5: Consider the language consisting of all strings of a's whose length is a positive square, i.e.,

$$A = \{a^{n^2} : n \geq 1\}.$$

Describe, in plain English, a two-tape Turing machine that enumerates all strings in A.

- At the start of the computation,
  - the Turing machine is in the start state,
  - the first tape contains the symbol a and its tape head is at the $\Box$ to the right of this symbol,
  - the second tape contains the symbol a and its tape head is at this symbol.
- The start of the computation is called the start of step 1.
- At the start of step n,
  - the Turing machine is in the start state,
  - the first tape contains the string a^{n^2} and its tape head is at the leftmost $\Box$ to the right of this string,
  - the second tape contains the string a^n and its tape head is at its leftmost symbol.
- Explain how to transition from the start of step n to the start of step $n+1$.
- Since A is an infinite language, the Turing machine will not terminate.
- Explain the states that you are going to use, and explain the meaning of these states. Do **not** give the instructions!