I. Goal

The goal of this lab is to learn how to use JFLAP to design a Turing Machine to decide a language. Fortunately, JFLAP is fairly intuitive. Once the TM has been created, the “Input” button can be used to test a set of strings to see if they are in the language. You can also “Print” the TM.

II. Languages

Create a TM that accepts the following languages (\( \Sigma = \{a, b\} \)). Use the “Input” button to test a number of strings the TM accepts and some strings it rejects (turn in a screen shot of this for each). You may assume the input is bracketed by “*” if you would like. For example: “*abba*” for “abba”. On a separate sheet of paper describe each language as a 7-tuple.

The machines get progressively harder, but if done in order each should help with the following problems.

1) (25 pts) \( w : w = a^+b^* \)
2) (25 pts) \( w : w = a^n \) where \( n \) is even
3) (25 pts) \( w : w \) has an odd number of “a”s followed by an even number of “b”s
4) (25 pts) \( w : w \) has the same number of “a”s as “b”s in any order and the empty string is not accepted. \( |a| = |b| > 0 \).

Challenge Turing Machine

A Turing Machine that decides palindromes with an even number of characters. “abba” is accepted but “bab” is not.