1. [21 points] Consider function \( f: \{a,b,c\} \rightarrow \{a,b\} \) with the following definition:

\[
\begin{align*}
    f(a) &= b \\
    f(b) &= b \\
    f(c) &= a
\end{align*}
\]

a) What is the domain of \( f \)?

b) What is the codomain of \( f \)?

c) What is the image of \( f \)?

d) Is \( f \) one-to-one (an injection)?

e) Is \( f \) onto (a surjection)?

f) Is \( f \) one-to-one onto (a bijection)?

g) Is \( f \) a total function?

2. The Turing Machine below accepts the language \( L = (ab)^*a \) (that is, \( L = \{ \varepsilon, a, aba, ababa, abababa, \ldots \} \)), with input alphabet \( \Sigma = \{a, b\} \). State 0 is the start state.
a. [10 points] How many steps does this Turing Machine take to accept a string $s$ of length $n$, if $s \in L$? Express your answer as a function of $n$. (Give an exact answer; do not use big-O notation in your answer.)

b. [19 points] Add transitions to the Turing Machine so that the new Turing Machine will recognize $L$ (i.e., your Turing Machine should end with 1 on an otherwise blank tape if the input string is in $L$, and should end with 0 on an otherwise blank tape if the input string is not in $L$).

3. [25 points] Design a single-tape Turing Machine with the following behavior: If the first and last characters of the input are the same, then replace all the "b"s in the input by "a"s; otherwise, leave the string unchanged. In either case, the machine should halt scanning the leftmost symbol of the output. Assume input alphabet $\Sigma = \{a, b\}$.

For example, if the input string is $abaa$, then the output string should be $aaaa$; if the input string is $bb$, then the output string should be $aa$; and if the input string is $baa$, then the output string should be $baa$. 
4. [25 points] Design a 3-tape Turing Machine which accepts the language $\{a^x b^y a^x b^y \mid x, y \geq 0\}$. The input alphabet is $\{a, b\}$.

Your machine should have one input tape (read-only), one work tape, and one output tape (write-only). The machine will start with the input written on the input tape, scanning the leftmost symbol of the input. The work and output tapes will be initially blank. At the end of the computation, if the input string is in the language, the machine should halt scanning a 1 on an otherwise blank output tape. If the input string is in the language, the machine may exhibit any behavior except for halting with a 1 on the output tape.

Describe your algorithm in pseudocode, and then give the state diagram for your Turing Machine. Use this notation:

```
1
  a:R
  b:a
  B:B

2
```

means "When in state 1 reading $a$ on the input tape, $b$ on the work tape, and $B$ on the output tape, change state from 1 to 2, move right on the input tape, write an $a$ on the work tape, and write a $B$ on the output tape."