Problem 1
Consider the alphabet $\Sigma = \{a, b, (, ), \cup, *, \emptyset\}$. Construct a context-free grammar that generates all strings in $\Sigma^*$ that are regular expressions over $\{a, b\}$.

Problem 2
Write a Turing machine (high level only, pseudo code) that does the following: given a positive integer $n$ written in binary written on the left end of the tape, the Turing machine replaces it with the positive integer $n + 1$ written in binary, moves its head to the left end of the tape, and then halts.

Problem 3
Write a Turing machine (at the implementation level, i.e. define $Q$, $\Sigma$, $\delta$, $\Gamma$ etc.) that completes the same task as the machine in problem 2. You may assume the input is given in the correct format, i.e. no need to check that the input string is a positive integer written in binary.

Problem 4
Assume that RIGHT and LEFT are replaced by DOUBLE-RIGHT and DOUBLE-LEFT in the Turing machine, its transition function now has the form: $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{RR, LL\}$. At each point, the machine can move its head right two steps, or move its head left two steps. Is this Turing machine variant equivalent to the standard Turing machine? Namely, can any language recognized by one be recognized by the other? Prove your answer.

Problem 5
Let $M$ be a Turing machine and $w$ be a string in the input alphabet of $M$. Consider the problem
$$A = \{(M, w) | M \text{ moves left at some point during its computation on } w\}$$
Show that $A$ is decidable.