Problem 1: (15%)
Provide a Deterministic Finite Automaton that matches all strings of “a”s and “b”s containing an odd number of “a”s.

Problem 2: (15%)
Provide a context free grammar for the language described in Problem 1.

Problem 3: (20%)
Consider the expression language from the typing lectures, without functions, products, or subtypes, as summarized below. Define the typing context \( \Gamma = [y : \text{ref int}, b : \text{bool}] \) and the expression \( e \) by

\[
\text{let } x = 3 \text{ in if } (x < y) \lor b \text{ then alloc } (x + 1) \text{ else let } z = y := 8 \text{ in 4 end end.}
\]

Is there some type \( \tau \) such that \( \Gamma \vdash e : \tau \) is derivable using the rules? If no, say why not, i.e. show where an attempt to construct a typing derivation fails. If yes, give a suitable typing derivation.
Problem 4: (20%)

Provide a simple grammar that resides in position A in the above figure. Prove that your grammar resides in position A in an organized manner.

Problem 5: (20%)

Provide a simple grammar that resides in position B in the above figure. Prove that your grammar resides in position B in an organized manner.

Problem 6: (10%)

Provide a simple grammar that resides in position C in the above figure. Prove that your grammar resides in position C in an organized manner.