**Compiler Design Assignment**

Read the following Problem Description first:

Consider the problem of identifying a valid number by the Lexical Analyzer of a Compiler. Since, there are an infinite number of valid numbers, it would not be a *good* idea for you to list all these numbers and then hand it over to the Lexical Analyzer. The alternate idea is to represent the language of all the valid numbers using an appropriate Regular Expression, then convert the Regular Expression to an equivalent Non Deterministic Finite Automaton (NFA), then convert the Non Deterministic Finite Automaton (NFA) to an equivalent Deterministic Finite Automaton (DFA), then convert the Deterministic Finite Automaton (DFA) to an equivalent computer program and then finally hand over the program to the Lexical Analyzer so that it may use the program to check whether a number is a valid number or not.

Now, answer the following questions:

1. Write down a Regular Expression that can be used to represent the Language of all the valid numbers.
2. Convert the Regular Expression from part (a) to an equivalent Non Deterministic Finite Automaton (NFA).
3. Convert the Non Deterministic Finite Automaton (NFA) to an equivalent Deterministic Finite Automaton (DFA).
4. Convert the Deterministic Finite Automaton (DFA) to an equivalent computer program using the [DFA to Program Tutorial](https://docs.google.com/document/d/13puyfQ-JTbQvtHpYfTmoNVHPL5yn7jTB5eihtFZXyos/edit?usp=sharing)