Task 2.1. Use the general algorithm that we learned in class to design a non-deterministic finite automaton that recognizes the language $(+ \cup -)(0 \cup 1)^*$. 

Reminder:

• $\hat{0}$ is a language consisting of only one 1-symbol word $0$;
• $\hat{A}\hat{B}$ means concatenation.

Solution. We start with the standard non-deterministic automata for recognizing:

• the language $+ -$ that consists of a single word $+$, and
• the language $- -$ that consists of a single word $-$:

Then, we use the general algorithm for the union to design a non-deterministic automaton for recognizing the language $+ \cup -$:

Similarly, we get a non-deterministic automaton for recognizing the language $0 \cup 1$:
Now, we apply a standard algorithm for the Kleene star, and we get the following non-deterministic automaton for $(0 \cup 1)^*$:

Now, we use the algorithm for concatenation for combine them: final states of the automaton for $+ \cup -$ are no longer final, and from each of them, we add a jump to the starting state of the automaton for $(0 \cup 1)^*$:
Task 2.2. Transform the resulting non-deterministic finite automaton into a
deterministic one.

Solution. Let us first enumerate the states of the resulting non-deterministic
automaton.

In the beginning, before we see any symbol, we are in state 5, and we can also
jump to states 1 and 8. Thus, before we see any symbols, we can be in one of
the states 1, 5, and 8. This set \{1, 5, 8\} is thus the starting state of the desired
deterministic finite automaton. Checking where we can go from this state and
from the resulting states when we see one of the symbols +, −, 0, or 1, we arrive
at the following deterministic automaton.