1-2. A perfect grading is when half of the students get Bs, one quarter As, and one quarter Cs. Let $L$ be the language of all the sequences of letters $A$, $B$, and $C$ that correspond to perfect grading. For example, $BABC \in L$ but $ABC \notin L$. Prove that this language is not context-free.

3. The following Turing machine deletes a binary number:

- start, $-$ $\to$ moving, R (here, $-$ means blank)
- moving, 0 $\to$ R
- moving, 1 $\to$ R
- moving, $-$ $\to$ deleting, L
- deleting, 0 $\to$ $-$, L
- deleting, 1 $\to$ $-$, L
- deleting, $-$ $\to$ halt

Trace it on the example of the word 10. Explain how each step will be represented if we interpret the Turing machine as a finite automaton with two stacks.

4. Arithmetic operations on Turing machines:

a Design a Turing machine that subtracts 2 from a binary number.

b Trace your Turing machine, step-by-step, on the example of the number 3.

c Why in Turing machines (and in most actual computers) the representation of a binary number starts with the least significant digit?

5. The following finite automaton describes binary strings that start with 1:

- the starting state $s$;
- the final state $f$ meaning that the first symbol was 1; and
- the error state $e$ meaning that the first symbol was not 1.
Transitions are as follows:

- from the state \( s \), symbol 1 leads to the state \( f \) and symbol 0 leads to the state \( e \);
- from the state \( f \), each symbol leads back to \( f \);
- from the state \( e \), each symbol leads back to \( e \).

Use the general algorithm to transform this finite automaton into a Turing machine. Show, step-by-step, how your Turing machine will accept the string 10.

6. Give the formal definition of a feasible algorithm, and an explanation of what practically feasible means. Give two examples different from what we had in class:

- an example of a computation time which is formally feasible, but not practically feasible, and
- an example of a computation time which is practically feasible but not formally feasible.

7. What is P? What is NP? What does it means for a problem to be NP-hard? NP-complete? Give brief definitions. Give an example of an NP-complete problem: explain what is the input, what is the desired output. Is P equal to NP?