Automata, Computability, and Formal Languages
Spring 2023, Test 3

1-2. Let L be language of all the words that contain equal number of A's and B's and twice fewer C's. Prove that this language is not context-free.

3. The following Turing machine replaces a binary number with 0:

- start, – → moving, R (here, – means blank)
- moving, 0 → R
- moving, 1 → R
- moving, – → deleting, L
- deleting, 0 → –, L
- deleting, 1 → –, L
- deleting, – → R, final
- final, – → 0, back, L
- back, – → halt

Trace it on the example of the word 01. 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 adds 2 to 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 0:

- the starting state s;
- the final state f meaning that the first symbol was 0; and

- the error state e meaning that the first symbol was not 0.

Transitions are as follows:

- from the state s, symbol 0 leads to the state f and symbol 1 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 01.

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?

8. Prove that the square root of 12 is not a rational number.

9. Formulate the halting problem. Prove that it is not possible to check whether a given program halts on given data.

10. Formulate Church-Turing thesis. Is it a mathematical theorem? Is it a statement about the physical world?