# Solution to Homework 8

Tasks: In the corresponding lecture, we described an algorithm that, given a finite automaton, produces a context-free grammar – a grammar that generate a word if and only if this word is accepted by the given automaton.

1. On the example from the automaton B from Homework 1.4, show how this algorithm will generate the corresponding context-free grammar. Similarly to Homework 3, assume that we only have symbols a, r, and A.

2. On the example of a word Aaa accepted by this automaton, show how the tracing of acceptance of this word by the finite automaton can be translated into a generation of this same word by your context-free grammar.

Reminder. The automaton has the following form:
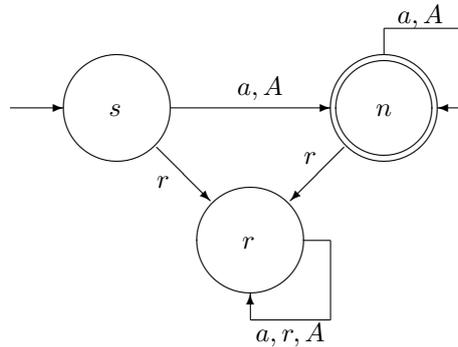
Solution to Task 1. The general algorithm for transforming FA into CFG is as follows:

- To each state q of the FA, introduce a new variable Q.

- The variable corresponding to the starting state will be the starting variable of the new CFG.

- For each transition of the finite automaton
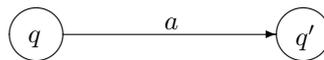
we add a rule $Q \to aQ'$.

- For each final state f of the FA, we add a rule $F \to \varepsilon$.

By applying this general algorithm to this FA, we get a CFG with 3 variables S, N, and R, three terminal symbols a, r, and R, the starting variable S and the following rules:

$$S \to rR$$
$$S \to aN$$
$$S \to AN$$
$$N \to rR$$
$$N \to aN$$
$$N \to AN$$
$$R \to rR$$
$$R \to aR$$
$$R \to AR$$
$$N \to \varepsilon$$

Solution to Task 2. Derivations in this grammar follow, step-by-step, the way the original finite automaton accepts a word. The word Aaa is accepted by the original finite automaton as follows:

- we start in the start state s; this corresponds to the starting variable S;

- then, we use the fact that once we are in the state s and we see the symbol A, then we move to the state n; this transition corresponds to the rule $S \to AN$, so the generation so far is:

$$\underline{S} \to AN;$$

- then, we use the fact that once we are in the state n and we see the symbol a, then we go back to the state n; this transition corresponds to the rule $N \to aN$, so generation so far is

$$\underline{S} \to A\underline{N} \to AaN;$$

- then, we again use the fact that once we are in the state n and we see the symbol a, then we go back to the state n; this transition corresponds to the same rule $N \to aN$, so generation so far is

$$\underline{S} \to A\underline{N} \to Aa\underline{N} \to AaaN;$$

- we have read all the symbols of the word, and we are in the final state n; for the FA, this means that the word Aaa is accepted; for CFG, we need to use the rule $N \to \varepsilon$ corresponding to the final state n; thus, we get the following derivation of the word Aaa:

$$\underline{S} \to A\underline{N} \to Aa\underline{N} \to Aaa\underline{N} \to Aaa.$$

So, we have indeed derived the word Aaa in the grammar.