Solution to Homework 11

Task. Use the general algorithm to transform the pushdown automaton from Problem 6 into a context-free grammar. Show, step-by-step, how the resulting grammar will generate the word ABAA.

Solution. Let us recall how the word ABAA is accepted by this automaton:

<table>
<thead>
<tr>
<th>read</th>
<th>A</th>
<th>B</th>
<th>A</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>s</td>
<td>w</td>
<td>a₁</td>
<td>w</td>
</tr>
<tr>
<td>stack</td>
<td>$</td>
<td>$</td>
<td>A</td>
<td>$</td>
</tr>
</tbody>
</table>

We start with the state s, we end up in the final state f. Thus, the first rule we apply is the rule $S \rightarrow A_s f$;

$$ S \rightarrow A_{sf} $$

The first symbol we push is the dollar sign, this dollar sign is popped at the end. Thus, we have the following combination of pop=push rules:

$$ s \xrightarrow{\varepsilon, \varepsilon \rightarrow \$} w \xrightarrow{\$} f $$

In general, we have the two transitions

$$ p \xrightarrow{x, \varepsilon \rightarrow t} q \quad r \xrightarrow{y, t \rightarrow \varepsilon} s $$

What do we need to plug in instead of p, q, etc. in the general 2-rule picture to come up with this particular picture:

- instead of p, we place s;
• instead of \( q \), we place \( w \);
• instead of \( s \) and \( t \), we place \( f \);
• instead of \( x \) and \( y \), we place \( \varepsilon \).

If we make these substitutions in the general rule:

\[
A_{ps} \rightarrow xA_{qr}y,
\]

we get the rule

\[
A_{sf} \rightarrow \varepsilon A_{wf} \varepsilon.
\]

Since concatenation with the empty string does not change anything, this means

\[
A_{sf} \rightarrow A_{wf}.
\]

Thus, the derivation so far takes the following form:

\[
\]

We covered how we push the dollar sign and how we pop it. Let us underline what we have covered:

<table>
<thead>
<tr>
<th>read</th>
<th>A</th>
<th>B</th>
<th>A</th>
<th>A</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>s</td>
<td>w</td>
<td>a₁</td>
<td>w</td>
<td>a₁</td>
</tr>
<tr>
<td>stack</td>
<td>$</td>
<td>$</td>
<td>A</td>
<td>$</td>
<td>$</td>
</tr>
</tbody>
</table>

If we ignore the dollar signs – since we already took care of them – then we see that we have three intermediate states with the empty stack. So, we need to use the transitivity rule, which in this case takes the form

\[
A_{wf} \rightarrow A_{wa_1}A_{a_1}A_{wa_1}A_{a_1}f.
\]

Thus, the derivation tree takes the following form:
**Transition** $A_{wa_1}$. In the first transition from $w$ to $a_1$ we have only one rule, so we add a fictitious rule to form a pair. We ignore the dollar signed that do not change:

$$w \xrightarrow{A, \varepsilon \to \varepsilon} a_1 \quad a_1 \xrightarrow{\varepsilon, \varepsilon \to \varepsilon} a_1$$

This combination leads to the rule $A_{wa_1} \to AA_{a_1 a_1} \varepsilon$, i.e., $A_{wa_1} \to AA_{a_1 a_1}$. Here, the remaining transition between $a_1$ and $a_1$ does not include any additional steps, so we can use the rule $A_{a_1 a_1} \to \varepsilon$. So, we get the rule $A_{wa_1} \to \Lambda$.

**Transition** $A_{a_1 w}$. For the transition between $a_1$ and $w$, we push $A$ and then pop $A$:

$$a_1 \xrightarrow{\varepsilon, \varepsilon \to A} w \quad w \xrightarrow{B, A \to \varepsilon} w$$

So, we have a transition $A_{a_1 w} \to \varepsilon A_{w w} B$, i.e., $A_{a_1 w} \to A_{w w} B$. Here, the remaining transition between $w$ and $w$ does not include any additional steps, so we can use the rule $A_{w w} \to \varepsilon$. So, we get the rule $A_{a_1 w} \to B$.

**Transition** $A_{wa_1}$. The next transition $A_{wa_1}$ is the same as the first transition from $w$ to $a_1$, so it corresponds to the rule $A_{wa_1} \to \Lambda$.

**Transition** $A_{a_1 f}$. In the last transition $A_{a_1 f}$, we first push $A$, then pop $A$:

$$a_1 \xrightarrow{\varepsilon, \varepsilon \to A} w \quad f \xrightarrow{\varepsilon, A \to \varepsilon} f$$

So, we get the transition $A_{a_1 f} \to \varepsilon A_{wf} \varepsilon$, i.e., $A_{a_1 f} \to A_{wf}$. This takes care of the second $A$, so we get:
Now, if we ignore the second A, we get an intermediate state $a_2$ with an empty stack, so we need to use transitivity $A_{wf} \rightarrow A_{wa_2}A_{a_2f}$. For $A_{waz}$, there is only one rule, so we pair it with a fictitious trivial rule:

$$w \xrightarrow{A, \varepsilon} A \xrightarrow{a_2} A_{a_2}$$

So, we get $A_{waz} \rightarrow \varepsilon A_{a_2}A_{a_2}$, i.e., $A_{waz} \rightarrow \varepsilon A_{a_2}$. There are no rules for the transition $A_{a_2}$, so we get $A_{a_2z} \rightarrow \varepsilon$ and thus, $A_{waz} \rightarrow \varepsilon$.

Finally, for the transition $A_{a_2f}$, we push and pop the last A by using the following rules:

$$a_2 \xrightarrow{\varepsilon, \varepsilon} A \xrightarrow{w} A_{a_2w} \xrightarrow{A, \varepsilon} A_{a_2}$$

Here, we get the rule $A_{a_2f} \rightarrow \varepsilon A_{wz},$ i.e., $A_{a_2f} \rightarrow A_{wz}$. There are no rules for the transition $A_{wz}$, so we get $A_{wz} \rightarrow \varepsilon$ and thus, $A_{wz} \rightarrow \varepsilon$. So, the transition takes the following form: