Solution to Homework 9

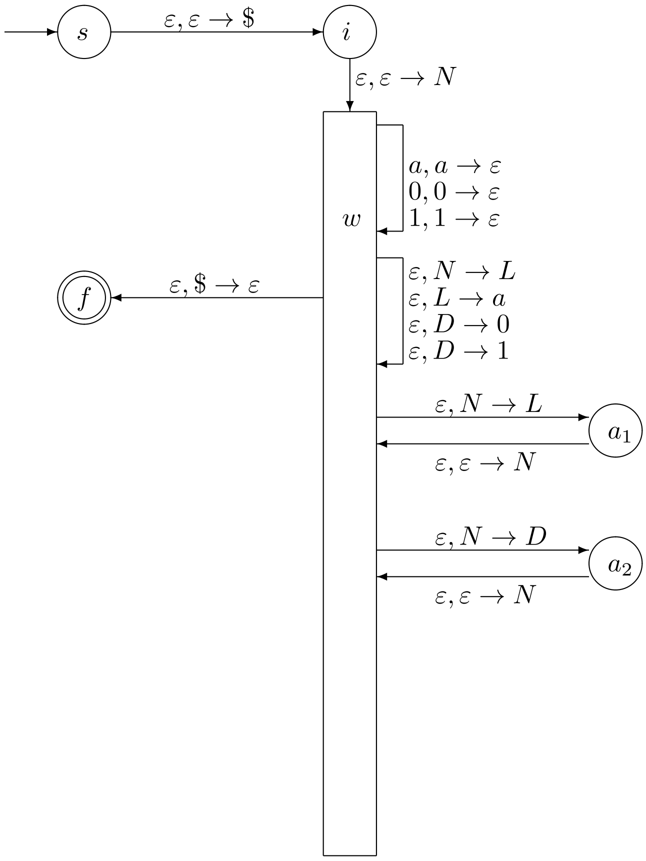
Background. In Problem 7, we considered a grammar with rules $N \to L$, $N \to NL$, $N \to ND$, $L \to a$, $D \to 0$, and $D \to 1$.

Tasks:

1. Use a general algorithm to construct a (non-deterministic) pushdown automaton that corresponds to context-free grammar described in Problem 7.

2. Show, step by step, how the word $a01$ will be accepted by this automaton.

Solution to Task 1. By using the general algorithm, we get the following pushdown automaton:



Solution to Task 2. Let us show how this is done on the example of the word +110 generated by the above automaton:

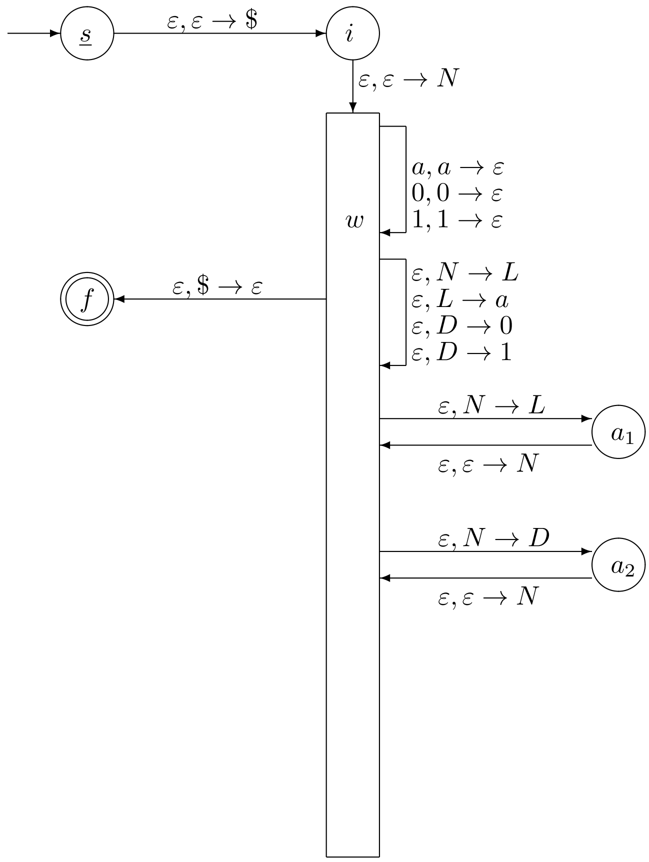
$$\underline{N} \to \underline{N}D \to \underline{N}DD \to \underline{L}DD \to a\underline{D}D \to a0\underline{D} \to a01.$$

To make this derivation clearer, let us mark the variables corresponding to different transitions by subscripts:

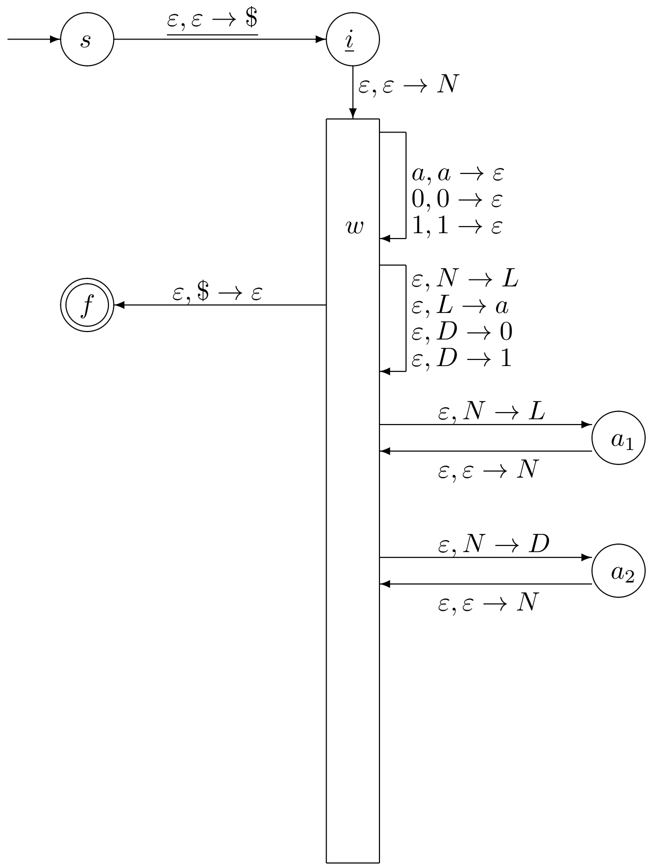
$$\underline{N_1} \to \underline{N_2}D_1 \to \underline{N_3}D_1D_2 \to \underline{L_1}DD \to a\underline{D_1}D_2 \to a0\underline{D_2} \to a01.$$

Let us now trace what our pushdown automaton will do.

We start in the state s with an empty stack:



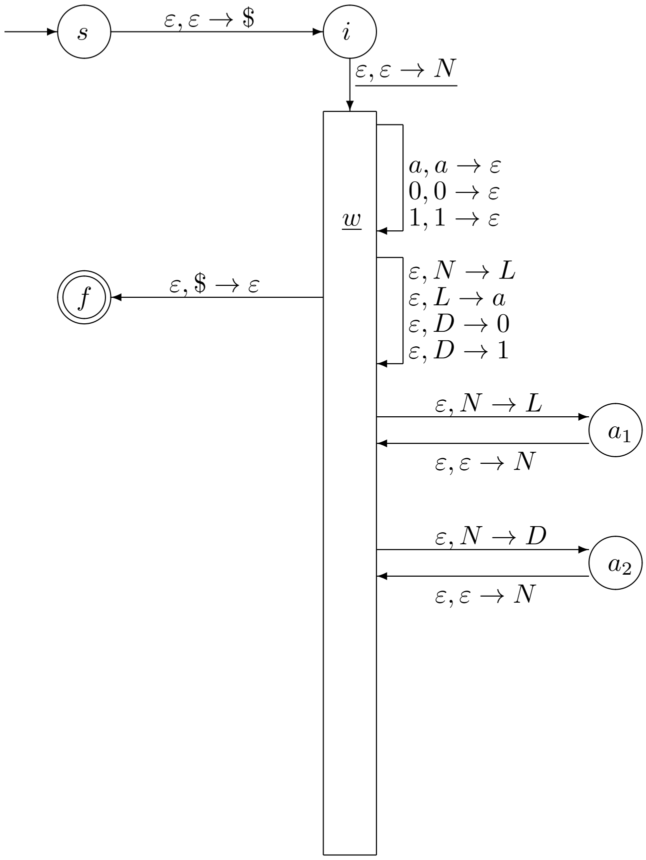
The only thing we can do when in the state s is push the dollar sign into the stack and get to the intermediate state i:



The contents of the stack is as follows:

$$\boxed{\$}$$

When we are in the state i, the only thing we can do is push the starting variable N into the stack and go into the working state w;



Now, the stack contains the starting variable on top of the dollar sign:

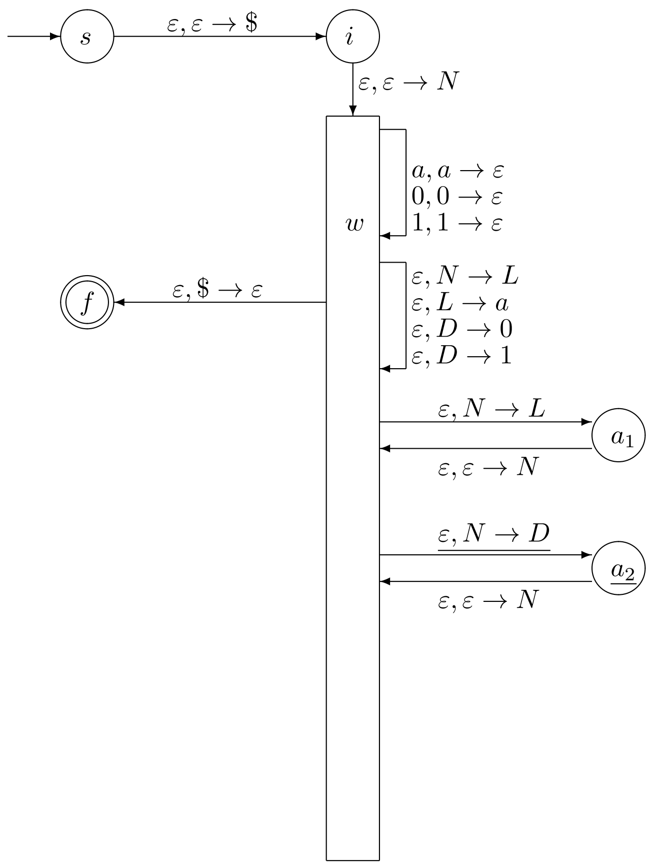
N
$\$$

Now that we are in the working state, we can start following the rules that were used to derive the word $a01$. The first rule was $N \to ND$, or, to be precise, $N_1 \to N_2 D_1$. As we have mentioned, this rule is implemented in two steps:

- first, we pop N and push the last symbol of the right-hand side – in this cases, the symbol D into the stack, getting into the auxiliary state a_2;
- then, we push N into the stack, and go back to the working state w.

Let us illustrate this step by step.

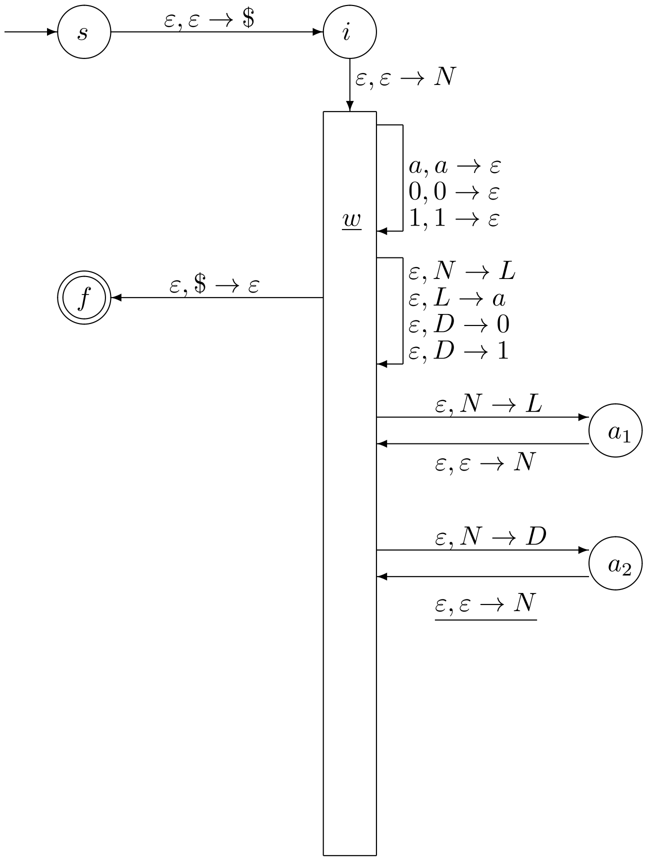
First, we pop N, push D, and go into the state a_2:



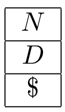
The stack will now have D instead of the original N:

D
$\$$

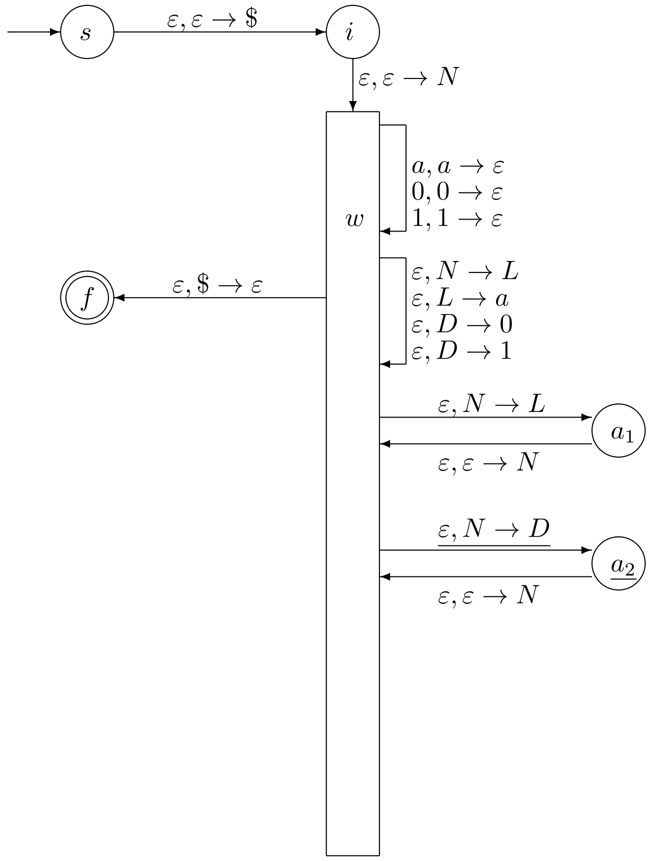
Then, we push N into the stack and go back to working state w:



The stack will now have N on top of its previous contents:



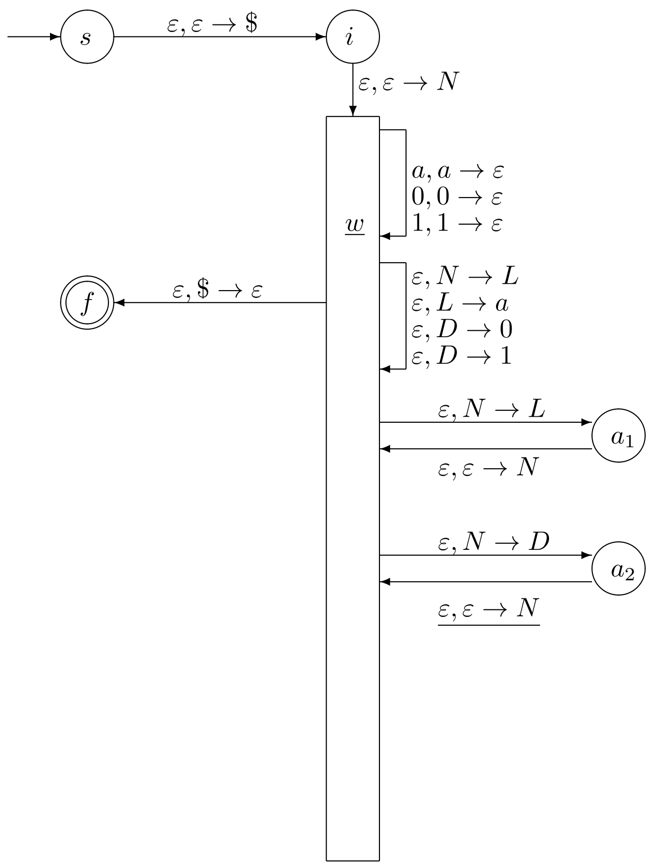
Now, the symbol N is on top of the stack, so we again use the rule $N \to ND$: first, we replace N by D and go to the state a_2, then push N and go back to the state w:



Now, the stack will have D instead of N:

D
D
\$

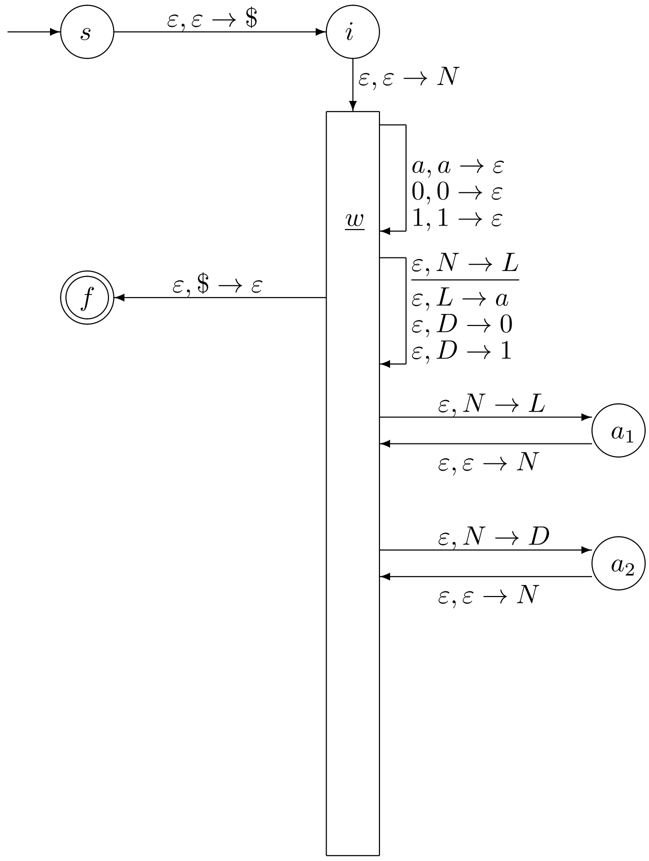
and



Now, the stack will have N on top:

N
D
D
$\$$

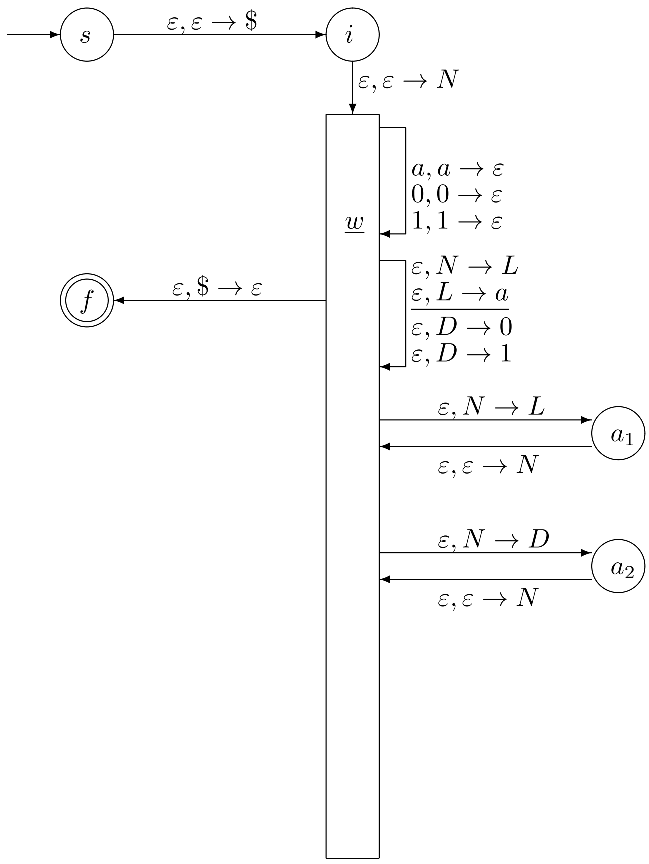
Next, we use the rule $N \to L$, i.e., replace N with L on top of the stack:



Now, the stack will have L on top:

L
D
D
\$

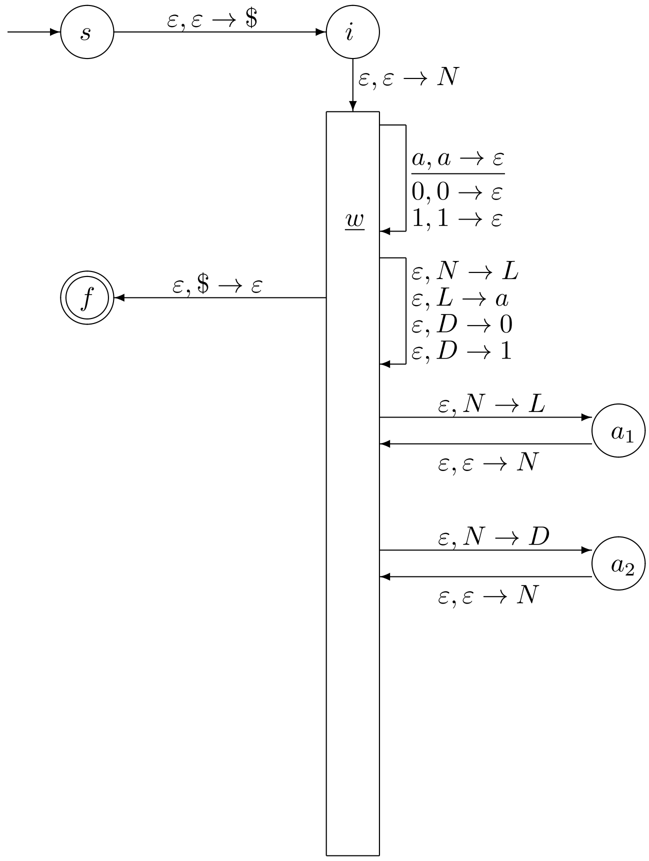
Next, we use the rule $L \to a$, i.e., replace L with a on top of the stack:



Now, the stack will have a on top:

a
D
D
$\$$

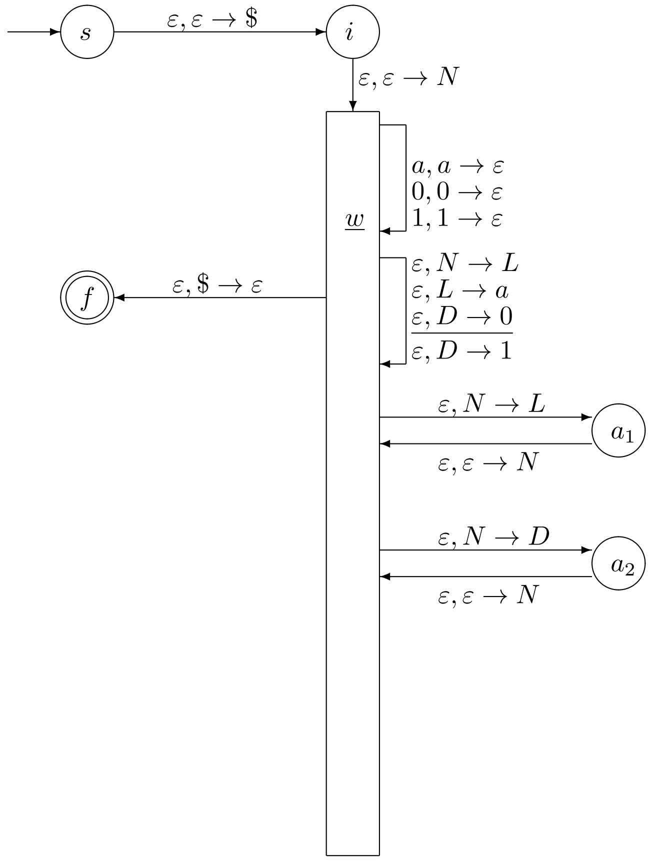
Now, we have a terminal symbol a on top of the stack. In this case, the only think we can do is use the rule $a, a \to \varepsilon$: read a and pop a from the top of the stack:



Now, the stack will have the following form:

D
D
$\$$

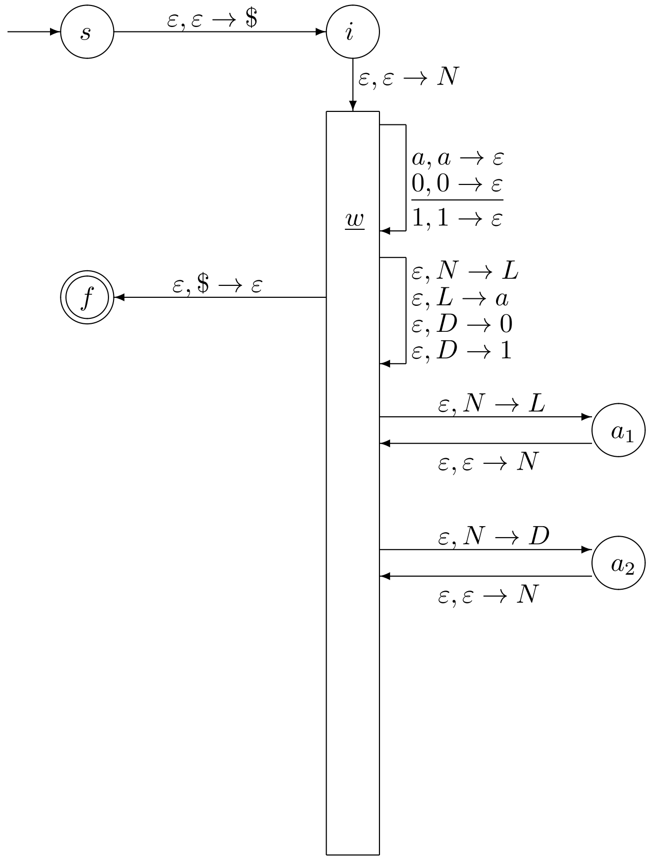
Next, we use the rule $D \to 0$, i.e., replace D with 0 on top of the stack:



Now, the stack will have 0 on top:

0
D
\$

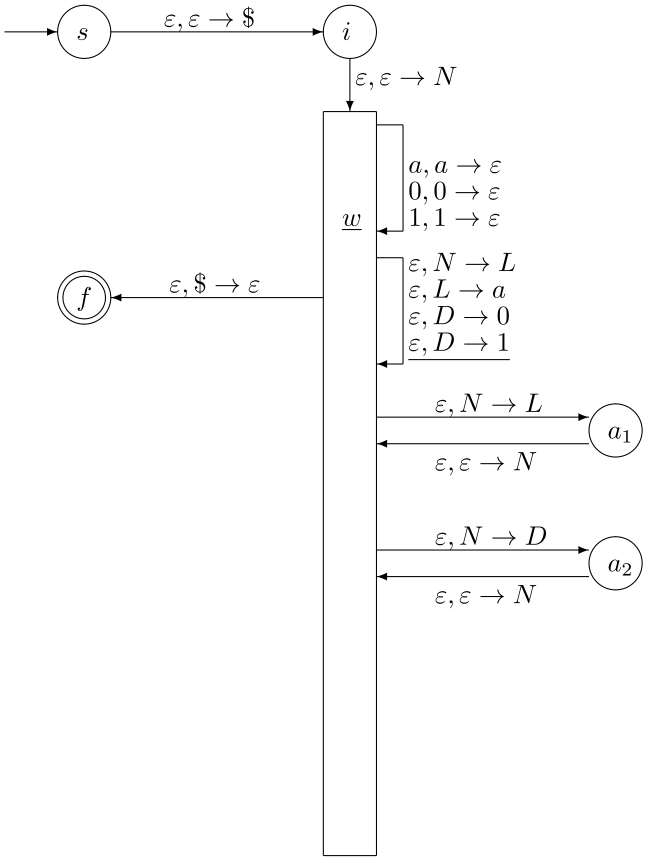
Now, we have a terminal symbol 0 on top of the stack, so the only thing we can do is to use the rule $0, 0 \to \varepsilon$, i.e., read 0 and pop 0 from the stack:



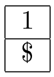
Now, the stack will have the following form:

D
$\$$

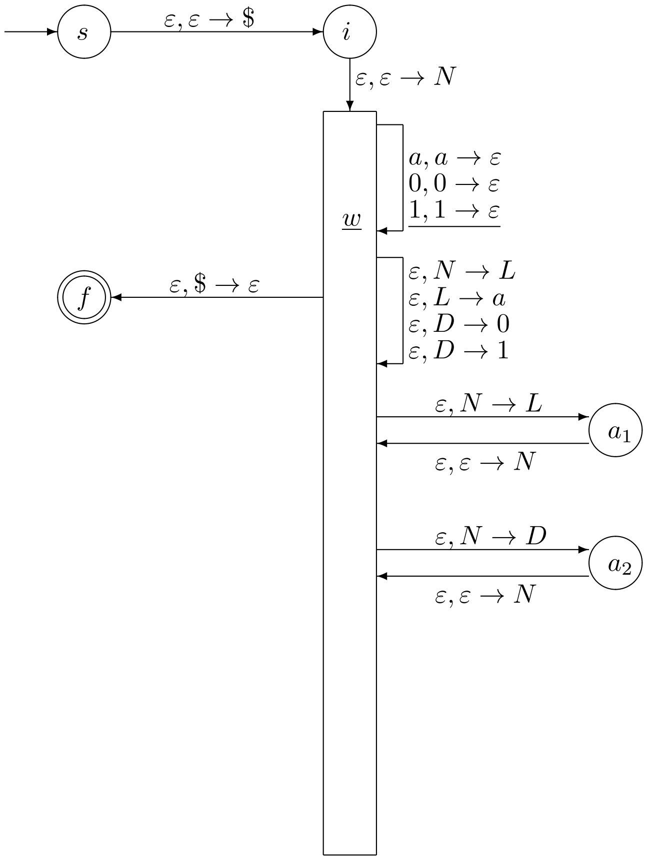
Then, we use the rule $D \to 1$, i.e., we replace D with 1 on top of the stack:



Now, the stack will have 1 on top:



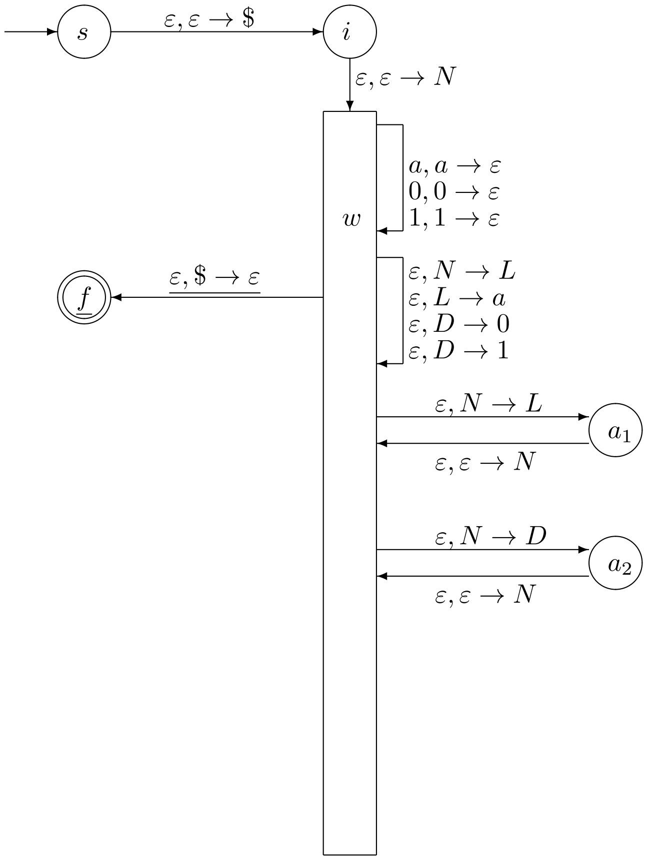
Now, there is a terminal symbol 1 on top of the stack, so we have no other choice but to use the rule $1, 1 \to \varepsilon$:



Now, the stack will only have the dollar sign:

$\boxed{\$}$

We read all the symbols, and the only symbol in the stack is the dollar sign. We can thus go to the final state:



The stack is now empty. We are in the final state with the empty stack, so the word $a01$ is accepted.

A graphical description of the transitions.

read									
state	s	i	w	a_2	w	a_2	w	w	w
stack		$\$$	N $\$$	D $\$$	N D $\$$	D D $\$$	N D D $\$$	L D D $\$$	a D D $\$$

read	a		0		1	
state	w	w	w	w	w	f
stack	D D $\$$	0 D $\$$	D $\$$	1 $\$$	$\$$	