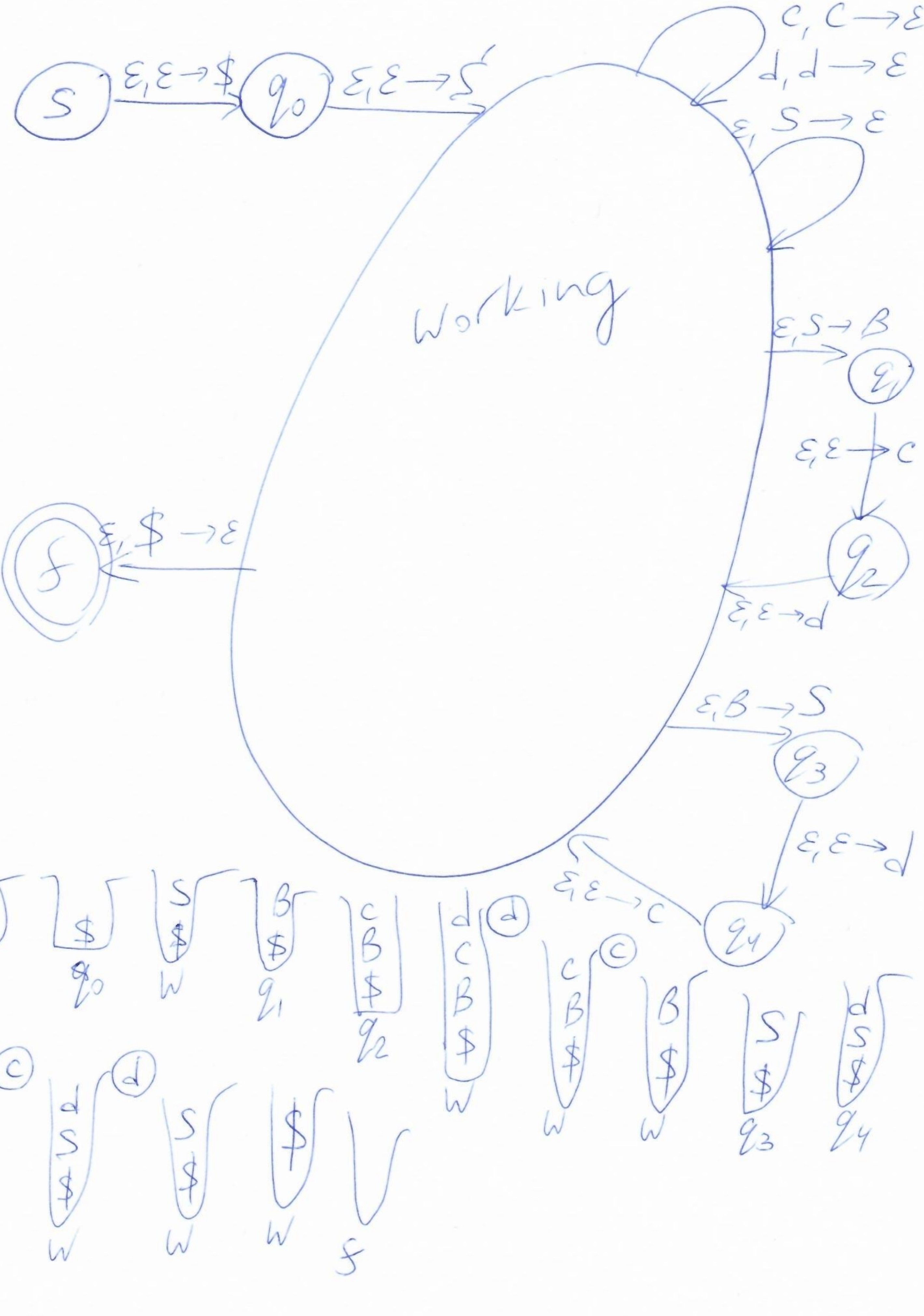
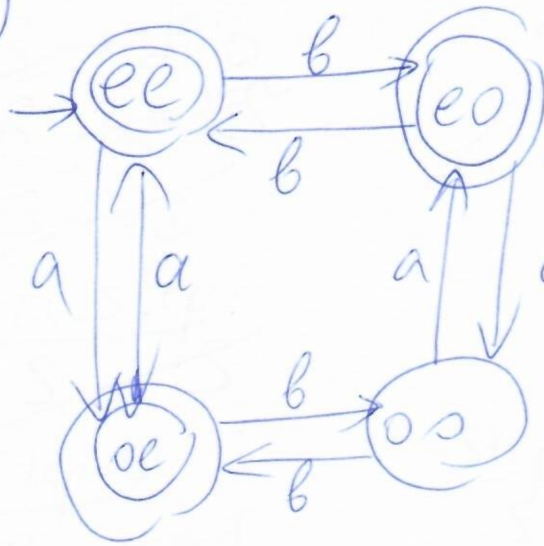
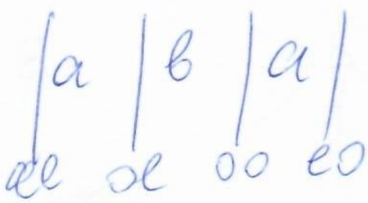
1


S
q0
ε, ε → $
ε, ε → S
c, c → ε
d, d → ε
ε, S → ε
Working
ε, S → B
q1
ε, ε → c
q2
ε, ε → d
ε, $ → ε
f
ε, B → S
q3
ε, ε → d
q4
ε, ε → c
S
q0
W
q1
q2
W
W
W
q3
q4
W
W
W
W
f

(2)





$E_e \to aO_e$

$E_e \to bE_o$

$O_e \to bO_o$

$O_e \to aE_e$

$E_o \to bE_e$

$E_o \to aO_o$

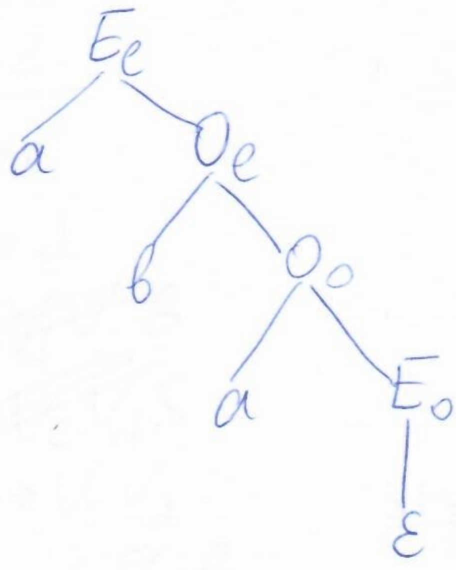
$O_o \to aE_o$

$O_o \to bO_e$

$E_e \to \varepsilon$

$E_o \to \varepsilon$

$O_e \to \varepsilon$



(3) $\cancel{S \rightarrow dcB}$

$\cancel{B \rightarrow cdS}$

$\cancel{S \rightarrow \varepsilon}$

Prel. step:

$\cancel{S_0 \rightarrow S}$

Step 0

$\cancel{B \rightarrow cd}$

$S_0 \rightarrow \varepsilon$

Step 1:

$\cancel{S_0 \rightarrow dcB}$

Step 2:

$V_c \rightarrow c$

$V_d \rightarrow d$

$\cancel{S \rightarrow V_d V_c B}$

$\cancel{B \rightarrow V_c V_d S}$

$B \rightarrow V_c V_d$

$\cancel{S_0 \rightarrow V_d V_c B}$

Step 3:

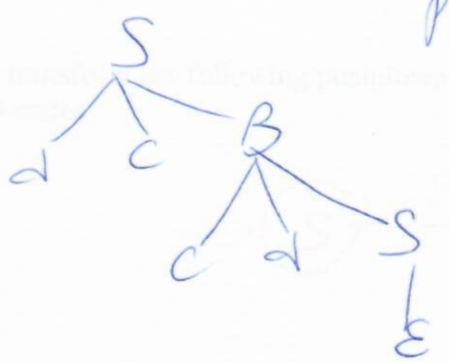
$V_{cd} \rightarrow V_c V_d$

$V_{dc} \rightarrow V_d V_c$

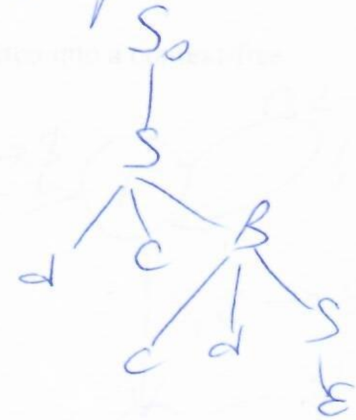
$S \rightarrow V_{dc} B$

$B \rightarrow V_{cd} S$

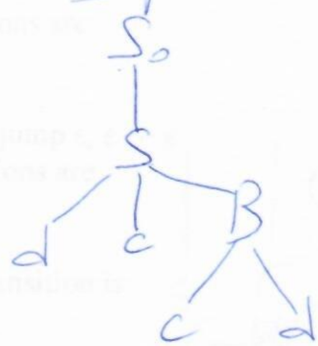
$S_0 \rightarrow V_{dc} B$



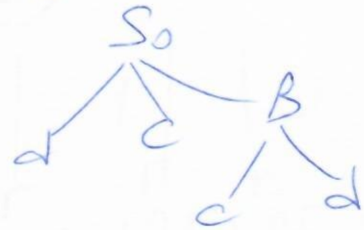
Prel. step



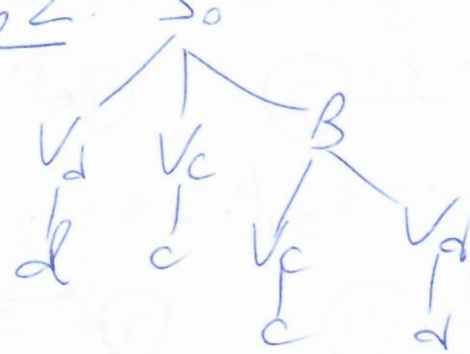
Step 0:



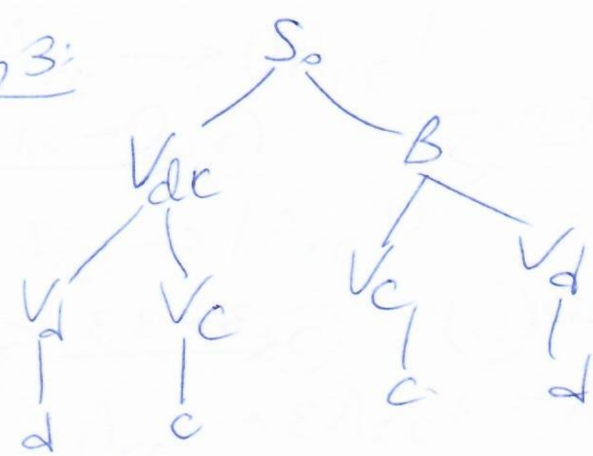
Step 1:



Step 2:

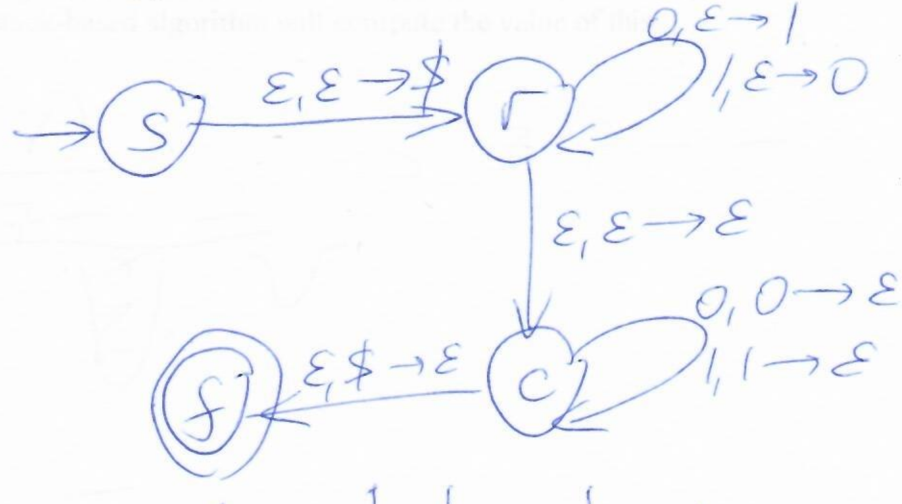


Step 3:



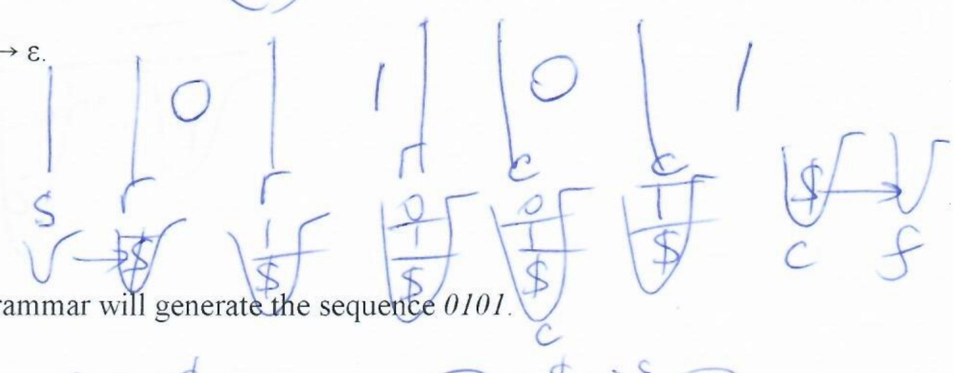
4. Use the general algorithm to transform the following pushdown automaton into a context-free grammar. This automaton has 4 states:

- the starting state s,
- the reading state r,
- the checking state c, and
- the final state f.



The transitions are as follows:

- From s to r, the transition is:
  - ε, ε → $;
- From r to r, the transitions are:
  - 0, ε → 1;
  - 1, ε → 0;
- From r to c, we have a jump ε, ε → ε.
- From c to c, the transitions are:
  - 0, 0 → ε
  - 1, 1 → ε
- From c to f, the only transition is:
  - ε, $ → ε.



Show, step-by-step, how the resulting grammar will generate the sequence *0101*.

$s \xrightarrow{\varepsilon,\varepsilon\to\$} r$ $\quad c \xrightarrow{\varepsilon,\$\to\varepsilon} f$

$S \to A_{sf}$

$A_{sf} \to \varepsilon A_{rc} \varepsilon = A_{rc}$

$r \xrightarrow{0,\varepsilon\to 1} r$ $\quad c \xrightarrow{1,1\to\varepsilon} c$

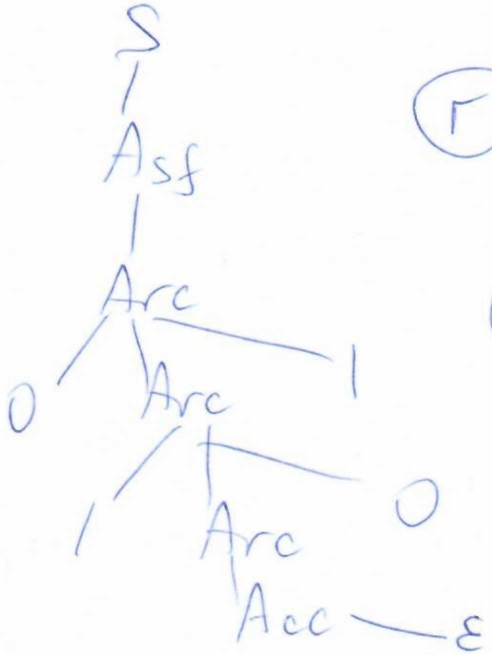
$A_{rc} \to 0 A_{rc} 1$

$r \xrightarrow{1,\varepsilon\to 0} r$ $\quad c \xrightarrow{0,0\to\varepsilon} c$

$A_{rc} \to 1 A_{rc} 0$

$r \xrightarrow{\varepsilon,\varepsilon\to\varepsilon} c$ $\quad c \xrightarrow{\varepsilon,\varepsilon\to\varepsilon} c$

$A_{rc} \to \varepsilon A_{cc} \varepsilon = A_{cc}$



5. Show, step by step, how the stack-based algorithm will transform the expression 5 − (3 − 7) into a postfix expression, and then how a second stack-based algorithm will compute the value of this postfix expression.

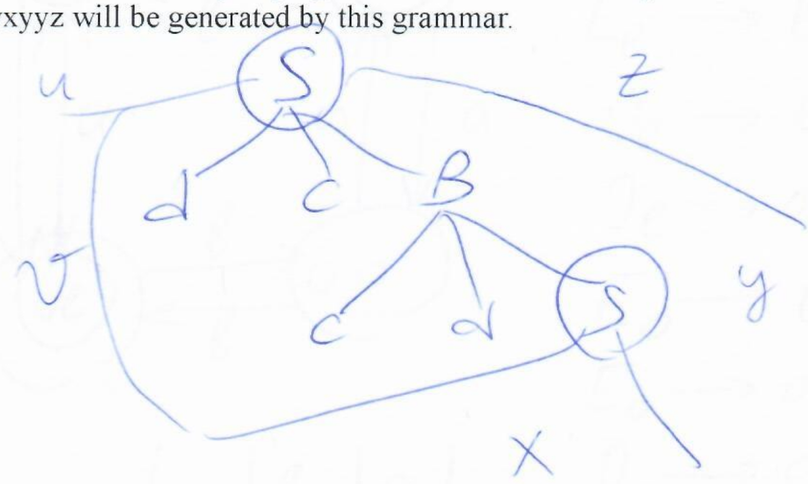
5 − (3 − 7)

5 3 7 − −

5 3 7 − −

5 3 7 − −

6. For the grammar from Problem 1, show how the word dccd can be represented as uvxyz in accordance with the pumping lemma for context-free grammars. Show that the corresponding word uvvxyyz will be generated by this grammar.



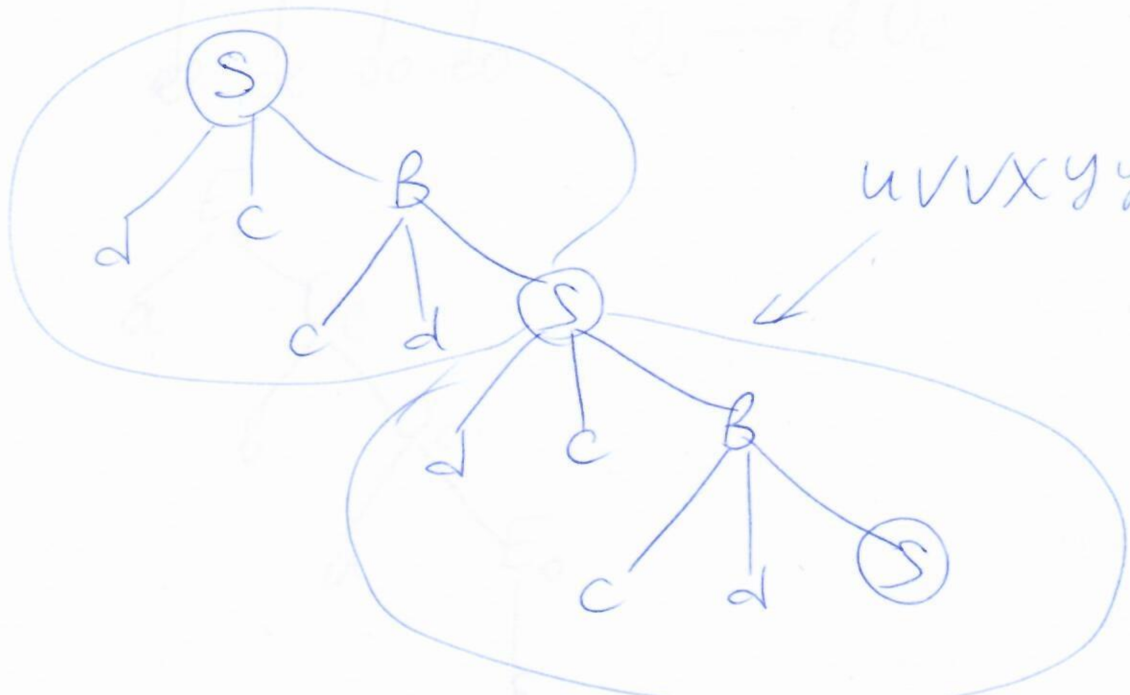
$u = \varepsilon$

$v = dccd$

$x = \varepsilon$

$y = \varepsilon$

$z = \varepsilon$



uvvxyyz