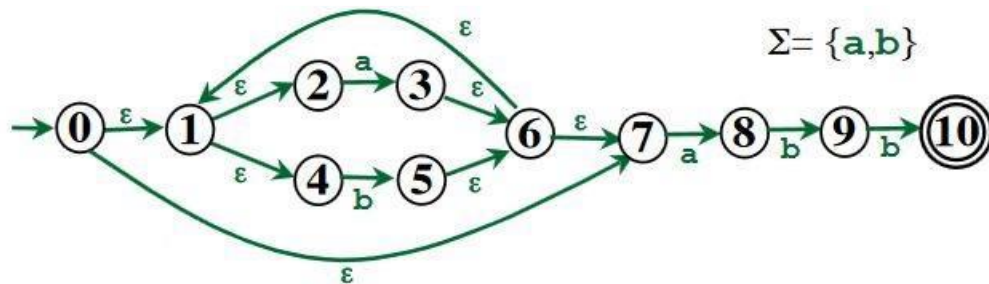


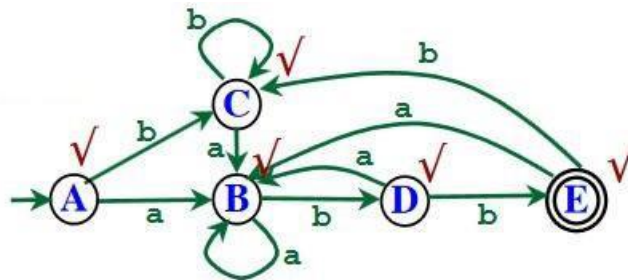
Proiectul 2

1. Transformare lambda-NFA \rightarrow AFD

Input: un automat finit nedeterminist cu λ/ε -deplasari oarecare

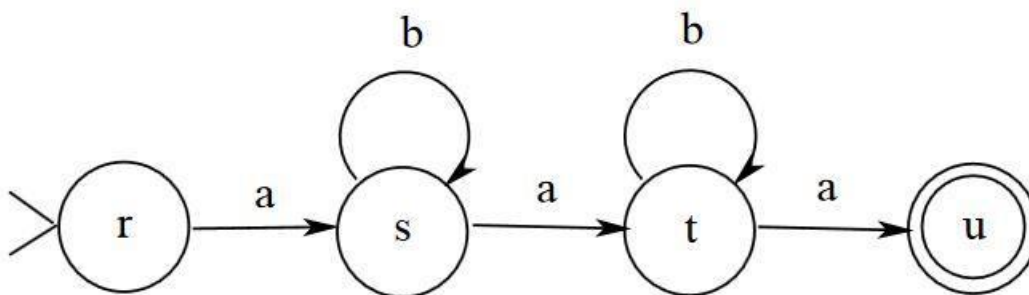


Output: automatul finit determinist rezultat in urma aplicarii algoritmului de conversie asupra automatului dat ca input



2. Transformare automat_finit \rightarrow expresie_regulata

Input: Un automat finit oarecare



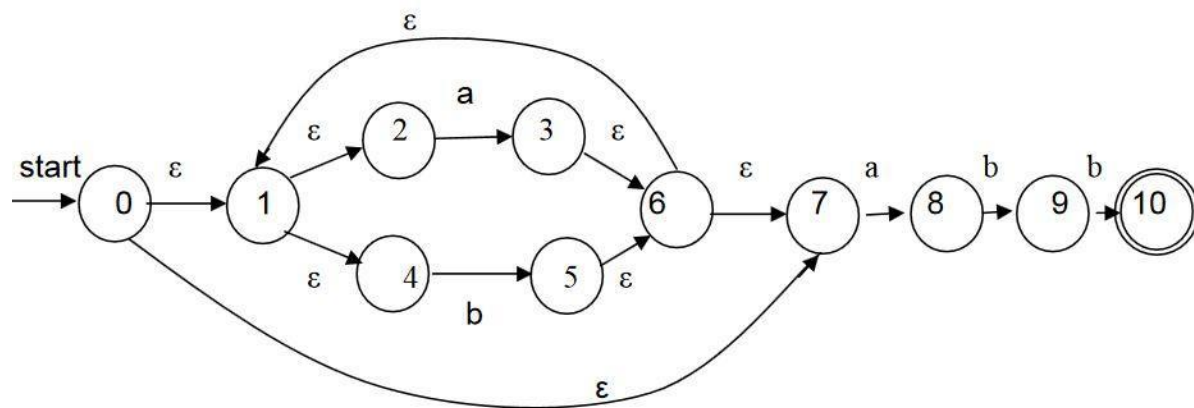
Output: Expresia regulata echivalenta cu automatul dat ca intrare
 ab^*ab^*a

3. Transformare expresie_regulata \rightarrow automat_finit

Input: o expresie regulata oarecare
 $E = (a|b)^*abb$

Output: automatul finit echivalent expresiei regulate date la intrare (automatul nu trebuie sa fie neaparat determinist). Intrucat un automat poate avea diverse forme echivalente, este acceptata orice solutie echivalenta cu automatul finit determinist minimal corespunzator expresiei regulate.

Automatul rezultat:



4. Pentru o gramatica regulata, sa se genereze toate cuvintele avand o lungime maxima data.

Input: O gramatica regulata oarecare si un numar natural, n .

G :

$N = \{S, A\}$

$\Sigma = \{a, b, c\}$

$S \rightarrow aS$

$S \rightarrow bA$

$A \rightarrow \epsilon$

$A \rightarrow cA$

Output: Lista cuvintelor de dimensiune n recunoscute de gramatica

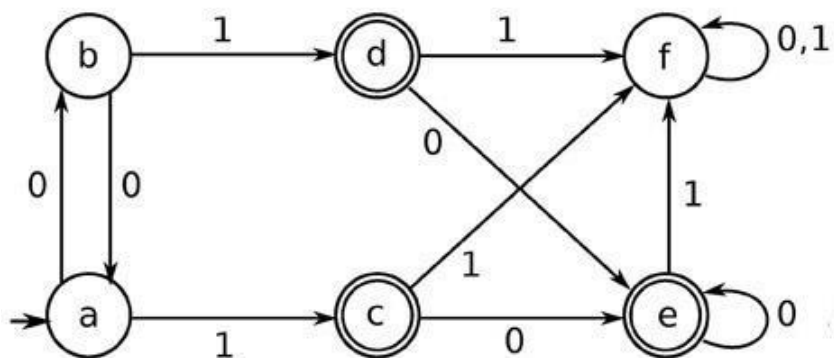
$n = 2 \rightarrow ["ab", "bc"]$

$n = 3 \rightarrow ["aab", "abc", "bcc"]$

$n = 4 \rightarrow ["aaab", "aabc", "abcc", "bccc"]$

5. Automatul minimal

Input: un automat finit determinist oarecare



Output: automatul finit determinist minimal rezultat in urma aplicarii algoritmului

