Automate pushdown

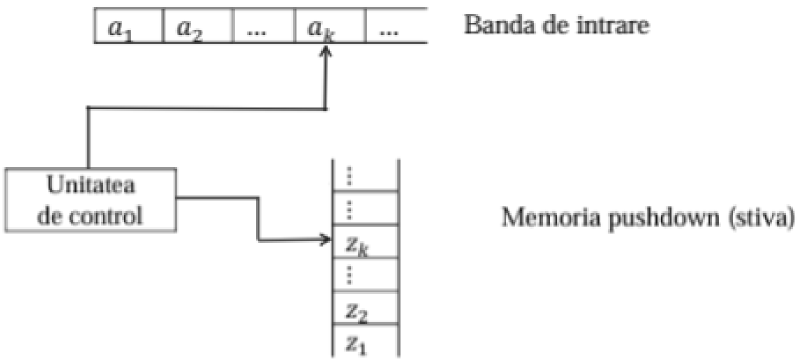
Un caz particular de automat finit determinist este automatul numit **pushdown,** care are memoria externă organizată sub formă de stivă, adică o bandă verticală nemărginită în jos, alcătuită din celule, în fiecare celulă putându-se memora câte un simbol dintr-un anumit alfabet.

Datorită stivei, automatul pushdown (***APD***), poate reține orice cantitate de informație, pe care o poate accesa în modul **ultimul intrat-primul ieșit (LIFO)**. Prima celulă din stivă, aflată în partea de sus, permite atât introducere cât și extragerea informației. Introducând un simbol în prima celulă, conținutul celorlalte celule ale stivei se deplasează automat în jos cu o celulă, ceea ce explică denumirea de pushdown. Ultimul simbol introdus în stivă, devine automat primul simbol care poate fi extras, iar primul simbol introdus în stivă este ultimul care poate fi extras. Cuvântul vid λ poate fi introdus în locul unui simbol aflat deja în prima celulă a stivei, ceea ce echivalează cu ștergerea/eliminarea acelui simbol din stivă.

Un ***APD*** funcționează în pași, adică **discret**, fiind alcătuit din trei componente:

- **Banda de intrare**, alcătuită din celule ce pot să conțină un simbol al unui alfabet finit ***Σ***, astfel încât celulele ocupate să fie în număr finit;
- **Unitatea de control** sau **dispozitivul de comandă,** care se află într-o anumită stare din mulțimea ***S*** a stărilor automatului și care posedă două capete:
  - un cap de citire a benzii de intrare
  - un cap de citire și de scriere din și în stivă.
- **Stiva** sau **memoria pushdown** care conține simboluri dintr-un alfabet finit ***Γ.***

Reprezentarea unui ***APD*** este dată în următoarea figură:



Definiția formală a ***APD*** este următoarea.

Definiția 1. Se numește **automat pushdown** 7-tuplul $\boldsymbol{P = (\Sigma, K, \Gamma, \delta, s_0, Z_0, F)}$, unde:

- ***Σ*** se numește **alfabetul de intrare** și este o mulțime finită nevidă;
- ***K*** se numește **mulțimea stărilor** și este o mulțime finită nevidă;
- ***Γ*** se numește **alfabetul memoriei pushdown** sau **alfabetul stivei** și este o mulțime finită nevidă;

Lect. univ. dr. TATIANA-CORINA DOSESCU
Facultatea de Informatică
Universitatea Titu Maiorescu

- $\boldsymbol{\delta}$ este **funcția de tranziție** a automatului;
- $\boldsymbol{s_0} \in \boldsymbol{K}$ este **starea inițială** a automatului;
- $\boldsymbol{Z_0} \in \boldsymbol{\Gamma}$ se numește **simbolul inițial/simbolul de start** al memoriei pushdown;
- $\boldsymbol{F} \subseteq \boldsymbol{K}$ este mulţimea **stărilor finale** ale automatului.

Funcția de tranziție se definește astfel:

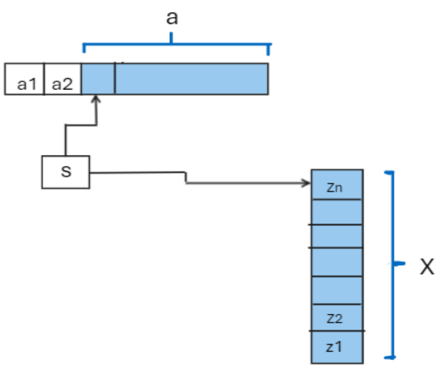
$$\boldsymbol{\delta} : \boldsymbol{K} \times (\boldsymbol{\Sigma} \cup \{\boldsymbol{\lambda}\}) \times \boldsymbol{\Gamma} \rightarrow 2^{\boldsymbol{K} \times \boldsymbol{\Gamma}^*} = \boldsymbol{\mathcal{P}}(\boldsymbol{K} \times \boldsymbol{\Gamma}^*),$$

$$\boldsymbol{\delta} \underbrace{(s, a, X)}_{\substack{configurația \\ curentă}} = \{(t, \gamma) |\ t \in \boldsymbol{K}, \gamma \in \boldsymbol{\Gamma}^*\}.$$

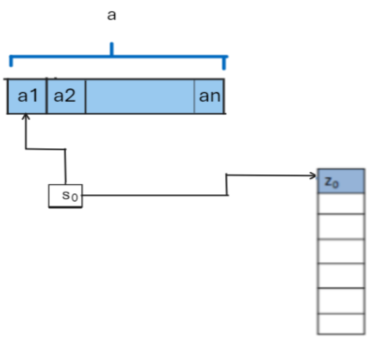
Configurația curentă (s, a, X) are următoarea semnificaţie:

- ***P*** se află în starea *s* ∈ ***K****;*
- banda de intrare a lui ***P*** conține *a*;
- *X* este în capul stivei.

Modelul fizic al ei este:



Configurația (s_0, a, Z_0) se numește **inițială** și are următorul model fizic:



Începerea activității unui ***APD*** se face din configurația inițială și după citirea unui simbol de intrare din $\boldsymbol{\Sigma} \cup \{\lambda\}$, unitatea de control efectuează două acțiuni:

1) trece într-o nouă stare din ***K*** sau rămâne în aceeași stare;

2) înscrie în stivă un cuvânt $w \in \boldsymbol{\Gamma}^*$, începând cu înlocuirea simbolului din prima celulă, astfel încât dacă $|w| = k \neq 0$ se produce automat deplasarea în jos cu *k* celule a conținutului stivei. Dacă $w = \lambda$ se şterge simbolul din prima celulă a stivei.

Un cuvânt aflat pe banda de intrare este *recunoscut* de ***APD***, după ce a fost citit ultimul simbol din alcătuirea cuvântului, numai în două situații: când ***APD*** a ajuns într-o stare finală sau când stiva s-a golit.

Formal, activitatea lui ***APD*** este descrisă mai jos.

Fie perechea (t, γ), unde $t \in \boldsymbol{K}$ este o nouă stare a automatului, iar $\gamma \in \boldsymbol{\Gamma}^*$ este un şir de simboluri înscrise în celulele stivei. Pentru $(t, \gamma) \in \boldsymbol{\delta}(s, a, X)$, atunci sunt posibile următoarele situaţii:

- dacă $\gamma = \lambda$, atunci se efectuează o operație *pop* asupra stivei, adică se șterge X iar conținutul stivei este împins în sus;
- dacă $\gamma = X$, stiva rămâne neschimbată;
- dacă $\gamma = Y \in \boldsymbol{\Gamma}$, atunci se înlocuiește simbolul X din capul stivei cu simbolul Y, fără a se efectua nicio operație pop sau push;
- dacă $\gamma = YZ \in \boldsymbol{\Gamma}^*$, atunci simbolul X se înlocuiește cu simbolul Z, iar asupra simbolului Y se efectuează o operație *push*, adică Y este împins în jos pe stivă.

Definiția 2. Fie $s, t \in \boldsymbol{K}$, $a \in \boldsymbol{\Sigma}$, $w \in \boldsymbol{\Sigma}^*$, $X \in \boldsymbol{\Gamma}$, $\alpha, \gamma \in \boldsymbol{\Gamma}^*$. ***APD***-ul ***P*** trece din configurația curentă $(s, aw, X\alpha)$ în configuraţia curentă $(t, w, \gamma\alpha)$ şi se scrie $(s, aw, X\alpha) \vdash (t, w, \gamma\alpha)$, dacă şi numai dacă $(t, \gamma) \in \boldsymbol{\delta}(s, a, X)$.

Se notează prin $\dot{\vdash}$ închiderea tranzitivă şi reflexivă a relaţiei $\vdash$.

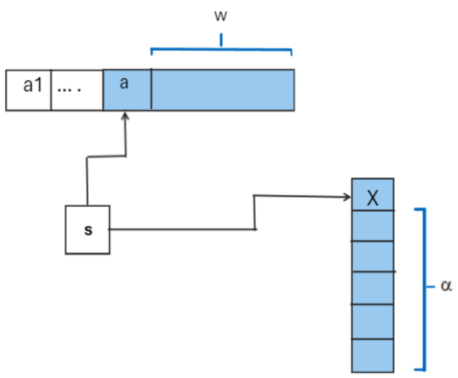
Definiția 3. Fie x și y două configurații curente. Se scrie $\boldsymbol{x} \dot{\vdash} \boldsymbol{y}$, dacă $x = y$ sau dacă există un șir finit de configurații curente $x = x_1, x_2, \ldots, x_n = y$, unde $x_i \vdash x_{i+1}$, $i = \overline{1, n-1}$.

Observația 1.

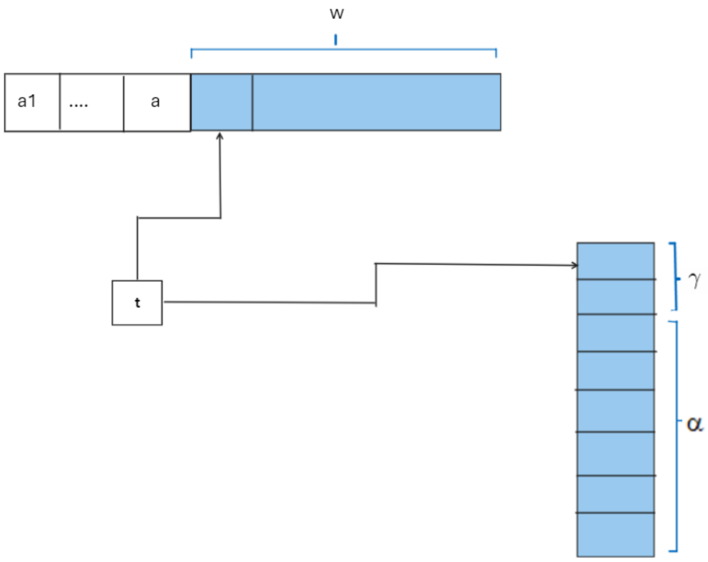
a) Configurația curentă (s, w, λ) nu poate precede nicio altă configurație curentă, deoarece $\lambda \notin \boldsymbol{\Gamma}$ și deci nu are sens $\boldsymbol{\delta}(s, w, \lambda), \forall\, w \in \boldsymbol{\Sigma}^*$. Acesta este motivul pentru care, la începutul activităţii, în stivă în prima celulă se află întotdeauna simbolul iniţial Z_0.

b) Configurația curentă (s, w, γ) reprezintă faptul că automatul se află în starea s, va citi cuvântul w de pe banda de intrare și că în stivă se află cuvântul γ scris de jos în sus, simbolul cel mai din dreapta lui γ, fiind scris în prima celulă.

c) Relația $(s, aw, X\alpha) \vdash (t, w, \gamma\alpha)$, indică faptul că automatul se află în starea s, că simbolul X se află în prima celulă a stivei și că a este simbolul de pe intrare ce urmează a fi citit:



Dacă $a = \lambda$ se efectuează o operație pop, dacă $a \neq \lambda$ singura evoluție posibilă după citirea simbolului a este trecerea automatului în starea t și înlocuirea simbolului X prin cuvântul γ, automatul urmând să citească cuvântul w:



d) Relația $(s, a_1a_2 \dots a_nw, \alpha) \dot{\vdash} (t, w, \beta)$ indică faptul că automatul se află în starea *s*, că α este conținutul stivei, că $a_1a_2 \dots a_n$ sunt simbolurile de pe banda de intrare ce urmează să fie citite și după cel puțin *n* pași, deoarece la unii pași banda de intrare poate să nu se deplaseze, automatul trece în starea *t*, stiva va conține cuvântul β și rămâne de citit cuvântul *w* de pe banda de intrare.

Definiția 4. ***APD***-ul ***P*** poate să **accepte** un cuvânt $w \in \boldsymbol{\Sigma}^*$ în două moduri:

i) **acceptă cuvântul w cu stare finală** dacă există $s \in \boldsymbol{F}$ și $\alpha \in \boldsymbol{\Gamma}^*$, deci configurația finală (s, λ, α) astfel încât:

$$(s_0, w, Z_0) \dot{\vdash} (s, \lambda, \alpha);$$

ii) **acceptă cuvântul w cu memorie pushdown vidă** (prin golirea stivei) dacă există $s \in \boldsymbol{K}$ astfel încât:

$$(s_0, w, Z_0) \dot{\vdash} (s, \lambda, \lambda).$$

Observația 2. În cazul unui ***APD*** care acceptă cu memorie vidă se consideră implicit că $\boldsymbol{F} = \emptyset$.

Definiția 5. **Limbajul acceptat de *APD*-ul *P***, notat prin $\boldsymbol{L(P)}$, este format din mulțimea tuturor cuvintelor acceptate de el, indiferent de modul de acceptare.

Exemplul 1. Se consideră ***APD***-ul $\boldsymbol{P = (\Sigma, K, \Gamma, \delta, s_0, Z_0, F)}$, care să accepte cu stare finală limbajul $\boldsymbol{L} = \{a^nb^n | \ n \geq 0\}$, unde:

- $\boldsymbol{\Sigma} = \{a, b\}$;
- $\boldsymbol{K} = \{s_0, s_1, s_2\}$;
- $\boldsymbol{\Gamma} = \{a, Z_0\}$;
- $\boldsymbol{F} = \{s_0\}$;
- $\boldsymbol{\delta} : \boldsymbol{K} \times (\boldsymbol{\Sigma} \cup \{\lambda\}) \times \boldsymbol{\Gamma} \to 2^{\boldsymbol{K} \times \boldsymbol{\Gamma}^*}$, unde:

$$\boldsymbol{\delta}(s_0, a, Z_0) = \{(s_1, aZ_0)\}; \quad (1)$$
$$\boldsymbol{\delta}(s_1, a, a) = \{(s_1, aa)\}; \quad (2)$$
$$\boldsymbol{\delta}(s_1, b, a) = \{(s_2, \lambda)\}; \quad (3)$$

$$\boldsymbol{\delta}(s_2, b, a) = \{(s_2, \lambda)\}; \qquad (4)$$
$$\boldsymbol{\delta}(s_2, \lambda, Z_0) = \{(s_0, \lambda)\}. \qquad (5)$$

a) Să se verifice dacă cuvântul $w_1 = aaabbb \in \boldsymbol{L}(\boldsymbol{P})$.

Într-adevăr are loc:

$$(s_0, w_1, Z_0) = (s_0, aaabbb, Z_0) \underset{(1)}{\vdash} (s_1, aabbb, aZ_0) \underset{(2)}{\vdash} (s_1, abbb, aaZ_0) \underset{(2)}{\vdash}$$
$$\underset{(2)}{\vdash} (s_1, bbb, aaaZ_0) \underset{(3)}{\vdash} (s_2, bb, aaZ_0) \underset{(4)}{\vdash} (s_2, b, aZ_0) \underset{(4)}{\vdash} (s_2, \lambda, Z_0) \underset{(5)}{\vdash} (s_0, \lambda, \lambda),$$

de unde rezultă că $w_1 \in \boldsymbol{L}(\boldsymbol{P})$, deoarece $s_0 \in \boldsymbol{F}$.

Se observă că automatul ar putea accepta cuvântul w_1 şi cu memorie pushdown vidă, caz în care ar trebui ca ***F*** = $\emptyset$.

b) Să se verifice dacă cuvântul $w_2 = aabbb \in \boldsymbol{L}(\boldsymbol{P})$.

Într-adevăr are loc:

$$(s_0, w_2, Z_0) = (s_0, aabbb, Z_0) \underset{(1)}{\vdash} (s_1, abbb, aZ_0) \underset{(2)}{\vdash} (s_1, bbb, aaZ_0) \underset{(2)}{\vdash}$$
$$\underset{(2)}{\vdash} (s_1, bb, aZ_0) \underset{(3)}{\vdash} (s_2, b, Z_0) \underset{(5)}{\vdash} (s_2, b, \lambda),$$

configurația curentă (s_2, b, Z_0) este totuna cu $(s_2, \lambda b, Z_0)$ și permite aplicarea lui (5), după care automatul nu mai poate citi simbolul *b*, căci rămâne în starea $s_2 \notin$ ***F***, adică $w_2 \notin \boldsymbol{L}(\boldsymbol{P})$.

Se poate observa că w_2 nu poate fi acceptat nici cu memorie pushdown vidă, căci deși memoria pushdown este vidă nu a terminat de citit cuvântul și este blocat într-o stare nefinală.

Exemplul 2. Să se construiască un ***APD*** notat $\boldsymbol{P} = (\boldsymbol{\Sigma}, \boldsymbol{K}, \boldsymbol{\Gamma}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{Z_0}, \boldsymbol{F})$**,** care să accepte limbajul $\boldsymbol{L} = \{ww^R |\, w \in \{a, b\}^*\}$, unde $w^R = a_n a_{n-1} \dots a_1$ dacă *w* = $a_1 a_2 \dots a_n$.

Se poate constata că acest limbaj este independent de context, fiind generat de următoarea gramatică independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$, unde:

$\boldsymbol{N} = \{S\}, \boldsymbol{T} = \{a, b\}, \boldsymbol{P} = \{S \to \lambda, S \to aSa, S \to bSb\}$.

Construcția lui ***P*** se face astfel:

Σ = $\{a, b\}$, $\boldsymbol{K} = \{s_0, s_1, s_2\}$, $\boldsymbol{\Gamma} = \{a, b, Z_0\}$, $\boldsymbol{F} = \{s_2\}$, $\boldsymbol{\delta} : \boldsymbol{K} \times (\boldsymbol{\Sigma} \cup \{\lambda\}) \times \boldsymbol{\Gamma} \to 2^{\boldsymbol{K} \times \boldsymbol{\Gamma}^*}$, unde:

$$\boldsymbol{\delta}(s_0, a, Z_0) = \{(s_0, aZ_0)\}; \quad (1)$$
$$\boldsymbol{\delta}(s_0, b, Z_0) = \{(s_0, bZ_0)\}; \quad (2)$$
$$\boldsymbol{\delta}\,(s_0, a, a) = \{(s_0, aa)\}; \quad (3)$$
$$\boldsymbol{\delta}\,(s_0, b, a) = \{(s_0, ba)\}; \quad (4)$$
$$\boldsymbol{\delta}\,(s_0, a, b) = \{(s_0, ab)\}; \quad (5)$$
$$\boldsymbol{\delta}\,(s_0, b, b) = \{(s_0, bb)\}; \quad (6)$$
$$\boldsymbol{\delta}\,(s_0, \lambda, a) = \{(s_1, a)\}; \quad (7)$$
$$\boldsymbol{\delta}\,(s_0, \lambda, b) = \{(s_1, b)\}; \quad (8)$$
$$\boldsymbol{\delta}\,(s_0, \lambda, \lambda) = \{(s_1, \lambda)\}; \quad (9)$$
$$\boldsymbol{\delta}\,(s_1, a, a) = \{(s_1, \lambda)\}; \quad (10)$$
$$\boldsymbol{\delta}\,(s_1, b, b) = \{(s_1, \lambda)\}; \quad (11)$$

$$\boldsymbol{\delta}\ (s_1, \lambda, Z_0) = \{(s_2, \lambda)\}. \qquad (12)$$

Se poate verifică dacă cuvântul $w_1 = abba \in \boldsymbol{L}(\boldsymbol{P})$. Într-adevăr:

$$(s_0, w_1, Z_0) = (s_0, abba, Z_0) \underset{(1)}{\vdash} (s_0, bba, aZ_0) \underset{(2)}{\vdash} (s_0, ba, baZ_0) \underset{(9)}{\vdash}$$
$$\underset{(9)}{\vdash} (s_1, ba, baZ_0) \underset{(11)}{\vdash} (s_1, a, aZ_0) \underset{(10)}{\vdash} (s_1, \lambda, Z_0) \underset{(12)}{\vdash} (s_2, \lambda, \lambda),$$

și deoarece $s_2 \in \boldsymbol{F}$, rezultă $w_1 \in \boldsymbol{L}(\boldsymbol{P})$.

Construcția unui automat pushdown echivalent cu o gramatică independentă de context

Definiția 6. O gramatică independentă de context ***G*** **este echivalentă** cu un automat pushdown ***P*** dacă $\boldsymbol{L}(\boldsymbol{G}) = \boldsymbol{L}(\boldsymbol{P})$.

Fiind dată gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ se poate **construi un *APD*** notat $\boldsymbol{P} = (\boldsymbol{\Sigma}, \boldsymbol{K}, \boldsymbol{\Gamma}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{Z_0}, \boldsymbol{F})$ echivalent, în felul următor:

- $\boldsymbol{\Sigma} = \boldsymbol{T}$;
- $\boldsymbol{K} = \{s_0\}$;
- $\boldsymbol{\Gamma} = \boldsymbol{N} \cup \boldsymbol{T} \cup \{\boldsymbol{Z_0}\}$;
- $\boldsymbol{Z_0} = \boldsymbol{S}$;
- $\boldsymbol{F} \neq \emptyset$;
- Funcția de tranziție $\boldsymbol{\delta}$ se construieşte astfel:
  - $\boldsymbol{\delta}\ (s_0, \lambda, A) = \{(s_0, \propto) \mid (A \to \alpha) \in \boldsymbol{P}\}$ pentru $\forall A \in \boldsymbol{N}$;
  - $\boldsymbol{\delta}\ (s_0, a, a) = \{\{s_0, \lambda\}\}$ pentru $\forall a \in \boldsymbol{T}$.

Exemplul 3. Se consideră gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ care generează expresii aritmetice simple în variabila a, unde:

$\boldsymbol{N} = (S, B, D, E)$, $\boldsymbol{T} = \{a, +, *, (,)\}$, iar ***P*** este alcătuită din producțiile:

$S \to E$	(1)
$E \to E+B$	(2)
$E \to B$	(3)
$B \to B*D$	(4)
$B \to D$	(5)
$D \to (E)$	(6)
$D \to a$	(7)

Fie expresia aritmetică $w = a * (a + a * a)$. Se poate verifica dacă $w \in \boldsymbol{L}(\boldsymbol{G})$. Într-adevăr:

$$S \underset{(1)}{\longrightarrow} E \underset{(3)}{\longrightarrow} B \underset{(4)}{\longrightarrow} B * D \underset{(5)}{\longrightarrow} D * D \underset{(7)}{\longrightarrow} a * D \underset{(6)}{\longrightarrow} a * (E) \underset{(2)}{\longrightarrow} a * (E + B) \underset{(3)}{\longrightarrow}$$
$$\underset{(3)}{\longrightarrow} a * (B + B) \underset{(5)}{\longrightarrow} a * (D + B) \underset{(7)}{\longrightarrow} a * (a + B) \underset{(4)}{\longrightarrow} a * (a + B * D) \underset{(5)}{\longrightarrow}$$

$$\underset{(5)}{\rightarrow} a * (a + D * D) \underset{(7)}{\rightarrow} a * (a + a * D) \underset{(7)}{\rightarrow} a * (a + a * a) = w,$$

adică $w \in \boldsymbol{L}(\boldsymbol{G})$.

Folosind procedeul de mai sus, se construiește un ***APD*** notat $\boldsymbol{P} = (\boldsymbol{\Sigma}, \boldsymbol{K}, \boldsymbol{\Gamma}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{Z_0}, \boldsymbol{F})$ echivalent cu gramatica $\boldsymbol{G}$, astfel:

$\boldsymbol{\Sigma} = \{a, +, *, (,)\}$, $\boldsymbol{K} = \{s_0\}$, $\boldsymbol{\Gamma} = \{S, B, D, E, a, +, *, (,)\}$, $\boldsymbol{Z_0} = S$, $\boldsymbol{F} = \emptyset$, iar funcția de tranziție $\boldsymbol{\delta} : \boldsymbol{K} \times (\boldsymbol{\Sigma} \cup \{\lambda\}) \times \boldsymbol{\Gamma} \rightarrow 2^{\boldsymbol{K} \times \boldsymbol{\Gamma}^*}$ se definește astfel:

$$\begin{array}{ll}
\boldsymbol{\delta}(s_0, \lambda, S) = \{(s_0, E)\}, & (1) \\
\boldsymbol{\delta}(s_0, \lambda, E) = \{(s_0, E + B), (s_0, B)\}, & (2.1;2.2) \\
\boldsymbol{\delta}(s_0, \lambda, B) = \{(s_0, B * D), (s_0, D)\}, & (3.1;3.2) \\
\boldsymbol{\delta}(s_0, \lambda, D) = \{(s_0, (E)), (s_0, a)\}, & (4.1;4.2) \\
\boldsymbol{\delta}(s_0, b, b) = \{(s_0, \lambda) \mid \forall b \in \Sigma\}. & (5)
\end{array}$$

Să se verifice dacă $w \in \boldsymbol{L}(\boldsymbol{G})$. Într-adevăr:

$$(s_0, w, Z_0) = (s_0, a * (a + a * a), S) \underset{(1)}{\vdash} (s_0, a * (a + a * a), E) \underset{(2.2)}{\vdash}$$

$$\underset{(2.2)}{\vdash} (s_0, a * (a + a * a), B) \underset{(3.1)}{\vdash} (s_0, a * (a + a * a), B * D) \underset{(3.2)}{\vdash}$$

$$\underset{(3.2)}{\vdash} (s_0, a * (a + a * a), D * D) \underset{(4.2)}{\vdash} (s_0, a * (a + a * a), a * D)$$

$$\underset{(5)}{\vdash} (s_0, * (a + a * a), * D) \underset{(5)}{\vdash} (s_0, (a + a * a), D) \underset{(4.1)}{\vdash} (s_0, (a + a * a), (E)) \underset{(5)}{\vdash}$$

$$\underset{(5)}{\vdash} (s_0, a + a * a), E)) \underset{(2.1)}{\vdash} (s_0, a + a * a), E + B)) \underset{(2.2)}{\vdash} (s_0, a + a * a), B + B)) \underset{(4.2)}{\vdash}$$

$$\underset{(4.2)}{\vdash} (s_0, a + a * a), a + B)) \underset{(5)}{\vdash}$$

$$\underset{(5)}{\vdash} (s_0, + a * a), + B)) \underset{(5)}{\vdash} (s_0, a * a), B) \underset{(3.1)}{\vdash} (s_0, a * a), B * D)) \underset{(4.2)}{\vdash}$$

$$\underset{(4.2)}{\vdash} (s_0, a * a), a * D)) \underset{(5)}{\vdash} (s_0, * a), * D) \underset{(5)}{\vdash} (s_0, a), D)) \underset{(4.2)}{\vdash} (s_0, a), a)) \underset{(5)}{\vdash}$$

$$\underset{(5)}{\vdash} (s_0,),)) \underset{(5)}{\vdash} (s_0, \lambda, \lambda).$$

Deci automatul acceptă cuvântul w cu memorie pushdown vidă.

Construcția unei gramatici independente de context echivalentă cu un automat pushdown

Fie $\boldsymbol{P} = (\boldsymbol{\Sigma}, \boldsymbol{K}, \boldsymbol{\Gamma}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{Z_0}, \boldsymbol{F})$ un ***APD*** care acceptă cu memorie pushdown vidă. Se poate construi o gramatică independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ echivalentă cu ***P***, în felul următor:

- ***N*** = $\{S\} \cup \{[sXt] | \, s, t \in \boldsymbol{K}, X \in \boldsymbol{\Gamma}\}$, unde:
  $S \notin \boldsymbol{K} \times \boldsymbol{\Gamma} \times \boldsymbol{K}$, iar tripletul $(s, X, t) \in \boldsymbol{K} \times \boldsymbol{\Gamma} \times \boldsymbol{K}$ şi este notat, concentrat, prin $[sXt]$;
- ***Σ*** = ***T***;
- producțiile din mulțimea ***P*** se obțin astfel:
  - $\forall s \in \boldsymbol{K}$ se obține producția $S \rightarrow [s_0 Z_0 s]$;
  - pentru orice $a \in \boldsymbol{\Sigma} \cup \{\lambda\}$ încât $(t, \lambda) \in \boldsymbol{\delta}(s, a, X)$ se obține producția:
    $[sXt] \rightarrow a$;
  - dacă $\boldsymbol{\delta}\,(s, a, X) = (t, Y_1 Y_2 \dots Y_k), k \in \mathbb{N}^*$, atunci se obțin producțiile:
    $[sXs_k] \rightarrow a[tY_1 s_1][s_1 Y_2 s_2] \dots [s_{k-1} Y_k s_k], \forall s_1, s_2, \dots, s_k \in \boldsymbol{K}$.

Exemplul 4. Se consideră limbajul $\boldsymbol{L_{01}} = \{01^n 0 | \, n \in \mathbb{N}^*\}$ care este acceptat cu memorie pushdown vidă de către ***APD-ul P***$= (\boldsymbol{\Sigma}, \boldsymbol{K}, \boldsymbol{\Gamma}, \boldsymbol{\delta}, \boldsymbol{s_0}, \boldsymbol{Z_0}, \boldsymbol{F} = \emptyset)$, definit astfel:

- ***Σ*** = $\{0,1\}$;
- ***K*** = $\{s_0, s_1, s_2\}$;
- ***Γ*** = $\{X, Z_0\}$;
- funcția de tranziție $\boldsymbol{\delta} : \boldsymbol{K} \times (\boldsymbol{\Sigma} \cup \{\lambda\}) \times \boldsymbol{\Gamma} \rightarrow 2^{\boldsymbol{K} \times \boldsymbol{\Gamma}^*}$ se definește astfel:

$$\boldsymbol{\delta}\,(s_0, 0, Z_0) = \{(s_1, XZ_0)\} \quad (1)$$
$$\boldsymbol{\delta}\,(s_1, 1, X) = \{(s_1, X)\} \quad (2)$$
$$\boldsymbol{\delta}\,(s_1, 0, X) = \{(s_2, \lambda)\} \quad (3)$$
$$\boldsymbol{\delta}\,(s_2, \lambda, Z_0) = \{(s_2, \lambda)\}. \quad (4)$$

Se poate verifica dacă $\boldsymbol{L_{01}}$ coincide cu $\boldsymbol{L(P)}$. Într-adevăr se poate scrie:

$$(s_0, 01^n 0, Z_0) \underset{(1)}{\vdash} (s_1, 1^n 0, XZ_0) \underset{(2)}{\vdash} (s_1, 1^{n-1} 0, XZ_0) \underset{(2)}{\vdash} (s_1, 1^{n-2} 0, XZ_0) \underset{(2)}{\vdash} \dots \underset{(2)}{\vdash}$$

$$\underset{(2)}{\vdash} (s_1, 10, XZ_0) \underset{(2)}{\vdash} (s_1, 0, XZ_0) \underset{(3)}{\vdash} (s_2, \lambda, Z_0) \underset{(4)}{\vdash} (s_2, \lambda, \lambda),$$

adică cuvântul $w_n = 01^n 0$ este acceptat cu memorie pushdown vidă de către ***APD***-ul ***P*** și $w_n \in \boldsymbol{L(P)}, n \in \mathbb{N}^*$.

Conform procedurii de mai sus, gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ echivalentă cu ***P***, se construiește în felul următor:

- $\boldsymbol{N} = \{S\} \cup \{[sXt] \mid s, t \in \boldsymbol{K}, X \in \boldsymbol{\Gamma}\}$;
- $\boldsymbol{T} = \boldsymbol{\Sigma} = \{0,1\}$;
- producțiile din mulțimea ***P*** se obțin astfel:

$$
\begin{aligned}
&S \rightarrow [s_0Z_0s_0];\\
&S \rightarrow [s_0Z_0s_1];\\
&S \rightarrow [s_0Z_0s_2];\\
&[s_1Xs_2] \rightarrow 0;\\
&[s_1Xs_1] \rightarrow 1;\\
&[s_2Z_0s_2] \rightarrow \lambda;\\
&[s_1Xs_0] \rightarrow 1[s_1Xs_0];-\\
&[s_1Xs_1] \rightarrow 1[s_1Xs_1];\\
&[s_1Xs_2] \rightarrow 1[s_1Xs_2];\\
&[s_0Z_0s_0] \rightarrow 0[s_1Xs_0][s_0Z_0s_0];\\
&[s_0Z_0s_1] \rightarrow 0[s_1Xs_1][s_1Z_0s_1];\\
&[s_0Z_0s_2] \rightarrow 0[s_1Xs_2][s_2Z_0s_2];\\
&[s_0Z_0s_1] \rightarrow 0[s_1Xs_0][s_0Z_0s_1];\\
&[s_0Z_0s_2] \rightarrow 0[s_1Xs_0][s_0Z_0s_2];\\
&[s_0Z_0s_0] \rightarrow 0[s_1Xs_1][s_1Z_0s_0];\\
&[s_0Z_0s_2] \rightarrow 0[s_1Xs_1][s_1Z_0s_2];\\
&[s_0Z_0s_0] \rightarrow 0[s_1Xs_0][s_0Z_0s_0];\\
&[s_0Z_0s_1] \rightarrow 0[s_1Xs_2][s_2Z_0s_1].
\end{aligned}
$$

Gramatica independentă de context $\boldsymbol{G} = (\boldsymbol{N}, \boldsymbol{T}, \boldsymbol{S}, \boldsymbol{P})$ astfel construită se poate simplifica obţinându-se:

$\boldsymbol{N} = \{S, [s_0Z_0s_2], [s_1Xs_2], [s_2Z_0s_2], [s_1Xs_2]\}$, $\boldsymbol{T} = \{0,1\}$ și producțiile:

$$
\begin{aligned}
&S \rightarrow [s_0Z_0s_2], && (1)\\
&[s_1Xs_2] \rightarrow 0, && (2)\\
&[s_2Z_0s_2] \rightarrow \lambda, && (3)\\
&[s_1Xs_2] \rightarrow 1[s_1Xs_2], && (4)\\
&[s_0Z_0s_2] \rightarrow 0[s_1Xs_2][s_2Z_0s_2]. && (5)
\end{aligned}
$$

Se poate verifica dacă $\boldsymbol{L_{01}}$ coincide cu $\boldsymbol{L(G)}$. Într-adevăr se poate scrie:

$$S \underset{(1)}{\rightarrow} [s_0Z_0s_2] \underset{(5)}{\rightarrow} 0[s_1Xs_2][s_2Z_0s_2] \underset{(4)}{\rightarrow} 01[s_1Xs_2][s_2Z_0s_2] \underset{(4)}{\rightarrow} 01^2[s_1Xs_2][s_2Z_0s_2] \underset{(4)}{\rightarrow}$$

$$\underset{(4)}{\rightarrow} \dots \underset{(4)}{\rightarrow} 01^n[s_1Xs_2][s_2Z_0s_2] \underset{(2)}{\rightarrow} 01^n0[s_2Z_0s_2] \underset{(3)}{\rightarrow} 01^n0,\ n \in \mathbb{N}^*.$$

Deci cuvântul $w_n = 01^n0 \in \boldsymbol{L(G)}, n \in \mathbb{N}^*$.