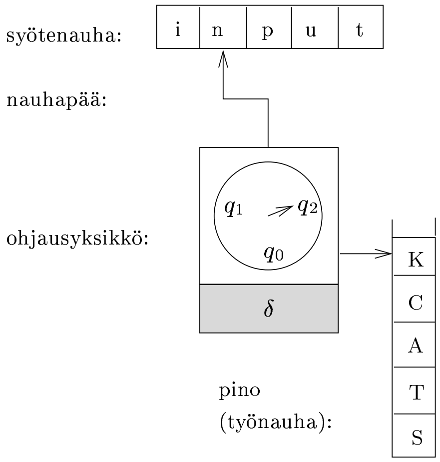
4.3 Pinoautomaatit

Kuten jo edellä huomasimme, eivät äärelliset automaatit kykene pitämään kirjaa lukemistaan syötemerkeistä. Esimerkiksi sulkulausekkeen tasapainoisuuden tutkimiseksi meidän täytyisi käyttää jonkinlaista laskuria, jota kasvatetaan aina kun vastaan tulee alkusulku '(' ja vähennetään, kun kohdataan loppusulku ')'. Pinoautomaatit ratkaisevat tämän ongelman tarjoamalla käyttöömme pinon, johon voi tallettaa symboleja myöhempää tutkimusta varten.

Pinoautomaatit eivät ole ohjelmoijalle yhtä kullanarvoisia apulaisia kuin äärelliset automaatit. Kontekstittonmien kielioppien pohjalta voi näet suoraan laatia tehokkaita ohjelmia. Tästä syystä käsittelemme pinoautomaatteja vain lyhyesti, ikään kuin johdatuksena Turingin koneisiin. Pinoautomaatit ovat kuitenkin hyödyllinen abstrakti laskennan malli kontekstittomien kielioppien tutkimuksessa ja toisinaan voi olla helpompi laatia ongelman kuvaava (epädeterministinen) pinoautomaatti, kuin laatia vastaava kielioppi.



Kuva 4.2: Pinoautomaatti.

- Pinoautomaatti on äärellinen automaatti, johon on lisätty yksi mielivaltaisen kokoinen pino työnauhaksi
- Ts. automaatti voi lukea ja kirjoittaa vain työnauhan toista päätä (pinon huippua), ja päästäkseen lukemaan aiemmin kirjoittamiaan merkkejä sen täytyy pyyhkiä viimeisin merkki pois
- Työnauha antaa automaatille "muistin", jonka avulla voidaan välttää äärellisen automaatin (joitakin) rajoituksia. (Huom! Tällainen pinomuisti ei riitä

esimerkiksi sen tarkistamiseen, onko merkkijonossa yhtä monta a:ta, b:tä ja c:tä.)

Määritelmä: Pinoautomaatti (engl. pushdown automaton) on kuusikko

$$M = (Q, \Sigma, \Gamma, \delta, q_0, F),$$

missä

- Q on *tilojen* äärellinen joukko;
- Σ on *syöteaakkosto*;
- Γ on *pinoaakkosto*;
- $\delta : Q \times (\Sigma \cup \{\epsilon\}) \times (\Gamma \cup \{\epsilon\}) \rightarrow \mathcal{P}(Q \times (\Gamma \cup \{\epsilon\}))$ on (joukkoarvoinen) *siirtymäfunktio*;
- $q_0 \in Q$ on *alkutila*;
- $F \subseteq Q$ on *(hyväksyvien) lopputilojen* joukko

- Siirtymäfunktion

$$\delta(q, \sigma, \gamma) = \{(q_1, \gamma_1), \ldots, (q_k, \gamma_k)\}$$

  tulkinta: Ollessaan tilassa q ja lukiessaan syötemerkin σ ja pinomerkin γ automaatti voi siirtyä johonkin tiloista $q_1, \ldots, q_k$ ja korvata pinon päällimmäisen merkin jollakin merkeistä $\gamma_1, \ldots, \gamma_k$.

  1. Jos $\sigma = \epsilon$, niin automaatti tekee siirtymän syötemerkkiä lukematta;
  2. Jos $\gamma = \epsilon$, niin automaatti ei lue pinomerkkiä, mutta pistää uuden merkin pinon päälle ($push(\gamma_i)$);
  3. Jos $\gamma \neq \epsilon$ ja $\gamma_i = \epsilon$, niin luetaan pinosta γ, mutta ei panna uutta merkkiä pinoon ($pop(\gamma)$)
  4. Jos sekä $\gamma \neq \epsilon$ että $\gamma_i \neq \epsilon$, niin sekä luetaan pinon päällimäinen merkki γ että kirjoitetaan pinoon γ_i (ts. $pop(\gamma), push(\gamma_i)$)

- Pinoautomaatit ovat siis yleisessä tapauksessa *epädeterministisiä*
- Automaatin tilanne on kolmikko $(q, w, \alpha) \in Q \times \Sigma^* \times \Gamma^*$
- *Alkutilanne syötteellä* x on kolmikko (q_0, x, ϵ)

- Tilanne $(q, w, \alpha) \sim$ automaatti on tilassa q, syötemerkkijonon käsittelemätön osa on w ja pinossa on ylhäältä alas lukien merkkijono α

- Tilanne (q, w, α) *johtaa suoraan* tilanteeseen (q', w', α'), merk.

$$(q, w, \alpha) \underset{M}{\vdash} (q', w', \alpha'),$$

  jos $w = \sigma w'$, $\alpha = \gamma\beta$, $\alpha' = \gamma'\beta$ $(|\sigma|, |\gamma|, |\gamma'| \leq 1)$, siten että

$$(q', \gamma') \in \delta(q, \sigma, \gamma)$$

- Tilanne (q, w, α) *johtaa tilanteeseen* (q', w', α'), merk.

$$(q, w, \alpha) \underset{M}{\vdash}^{*} (q', w', \alpha'),$$

  jos on olemassa tilannejono $(q_0, w_0, \alpha_0), \ldots, (q_n, w_n, \alpha_n)$,
  $n \geq 0$, siten että
  $(q, w, \alpha) = (q_0, w_0, \alpha_0) \underset{M}{\vdash} \cdots \underset{M}{\vdash} (q_n, w_n, \alpha_n) = (q', w', \alpha')$

- Pinoautomaatti M *hyväksyy* merkkijonon $x \in \Sigma^*$, jos

$$(q_0, x, \epsilon) \underset{M}{\vdash}^{*} (q_f, \epsilon, \alpha) \qquad \text{joillakin } q_f \in F \text{ ja } \alpha \in \Gamma^*$$

  (jos se syötteen loppuessa on jossakin hyväksyvässä lopputilassa); muuten M *hylkää* x:n

- Automaatin M *tunnistama kieli* on

$$L(M) = \{x \in \Sigma^* \mid (q_0, x, \epsilon) \underset{M}{\vdash}^{*} (q_f, \epsilon, \alpha) \text{ joillakin } q_f \in F \text{ ja } \alpha \in \Gamma^*\}$$

Esim. Ei-säännöllinen kontekstiton kieli $\{a^k b^k \mid k \geq 0\}$ voidaan tunnistaa seuraavanlaisella pinoautomaatilla:

$$M = (\{q_0, q_1, q_2, q_3\}, \{a, b\}, \{A, \underline{A}\}, \delta, q_0, \{q_0, q_3\}),$$

missä

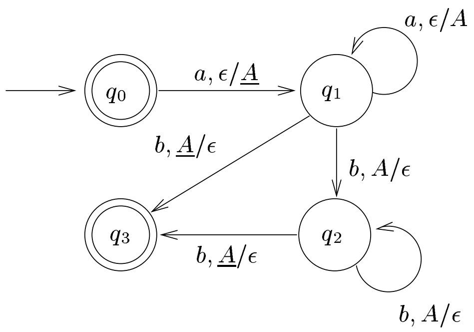
$$\begin{array}{rcll}
\delta(q_0, a, \epsilon) & = & \{(q_1, \underline{A})\}, & \\
\delta(q_1, a, \epsilon) & = & \{(q_1, A)\}, & \\
\delta(q_1, b, A) & = & \{(q_2, \epsilon)\}, & \\
\delta(q_1, b, \underline{A}) & = & \{(q_3, \epsilon)\}, & \\
\delta(q_2, b, A) & = & \{(q_2, \epsilon)\}, & \\
\delta(q_2, b, \underline{A}) & = & \{(q_3, \epsilon)\}, & \\
\delta(q, \sigma, \gamma) & = & \emptyset & \text{muilla } (q, \sigma, \gamma).
\end{array}$$

Esimerkiksi syötteellä *aabb* automaatti M toimii seuraavasti:

$$\begin{aligned}(q_0, aabb, \epsilon) &\vdash (q_1, abb, \underline{A}) \vdash (q_1, bb, A\underline{A}) \\ &\vdash (q_2, b, \underline{A}) \quad\;\; \vdash (q_3, \epsilon, \epsilon).\end{aligned}$$

Koska $q_3 \in F = \{q_0, q_3\}$, on siis $aabb \in L(M)$

Kielen $\{a^k b^k \mid k \geq 0\}$ tunnistava pinoautomaatti:



Kuva 4.3: Kielen $\{a^k b^k \mid k \geq 0\}$ tunnistava pinoautomaatti, jossa pinoaakkosto on $\{A, \underline{A}\}$.

Tässä on käytetty tilasiirtymäkaaviotyyppinen esitys:

- Siirtymäehto muotoa $a_i, \gamma/\gamma'$, missä
  - automaatti lukee merkkijonosta merkin a_i ja
  - pinosta merkin γ sekä
  - kirjoittaa pinoon merkin γ'
- "Push"-operaatio: ei lue pinosta merkkiä, mutta kirjoittaa uuden merkin γ' pinon päälle: $a_i, \epsilon/\gamma'$
- "Pop"-operaatio: lukee pinon päällimmäisen merkin γ, mutta ei kirjoita mitään pinoon: $a_i, \gamma/\epsilon$
- jos syötemerkki $a_i = \epsilon$, niin siirtymä syötemerkkiä lukematta – voi silti lukea tai kirjoittaa pinomerkkejä!

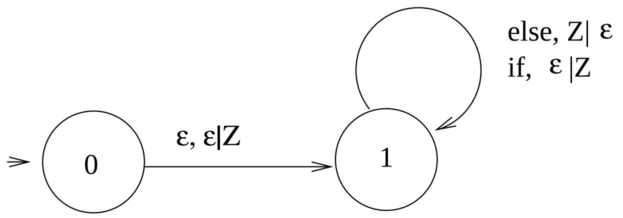
4.3.1 *Pinoautomaatin muunnos: "tyhjän pinon automaatti"

Määritellään seuraavan todistuksen avuksi pinoautomaatin muunnos, jolla ei ole erillistä hyväksyvää lopputilaa, vaan merkkijono hyväksytään missä tahansa tilassa, kunhan vain **pino on tyhjä syötteen loppuessa**. (Tavallinen pinoautomaattihan hyväksyy syötteen, jos se on syötteen loppuessa hyväksyvässä lopputilassa, olipa pinossa mitä tahansa.) Tällaista pinoautomaattia kutsutaan usein ”tyhjän pinon automaatiksi” (*empty stach automaton*, *null stack automaton*).

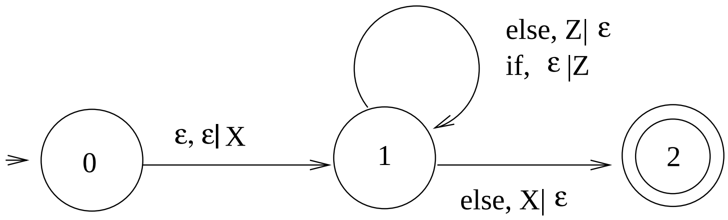
Tyhjän pinon automaatit ovat hyödyllinen apulainen, sillä ne tunnistavat *täsmälleen* samat kielet kuin tavalliset pinoautomaatit. (Ks. todistus esim. Hopcroft, Motwani, Ullman: luku 6.2)

Tavallinen pinoautomaatti hyväksyy x:n, jos	Tyhjän pinon automaatti hyväksyy x:n jos
$(q_0, x, \epsilon) \underset{M}{\vdash}^* (q_f, \epsilon, \alpha)$ $q_f \in F$ $\alpha \in \Gamma^*$	$(q_0, x, \epsilon) \underset{M}{\vdash}^* (q, \epsilon, \epsilon)$ $q \in Q$

Esim. Muodostetaan pinoautomaatti, joka tunnistaa virheellisen ohjelmakoodin, jossa *if*-*else*-parien lukumäärät eivät täsmää. Automaatti hyväksyy syötteen, jos siinä on enemmän *else*:jä kuin *if*:ejä.



Kuva 4.4: Tyhjän pinon automaatti $if - else$ -virheen haivaitsemiseksi.



Kuva 4.5: Saman kielen tunnistava lopputilallinen automaatti.

Lause: Kieli on kontekstiton, jos ja vain jos se voidaan tunnistaa pinoautomaatilla.
**Todistuksen idea:*
Käytetään hyväksi tyhjän pinon automaatteja. Jos väite pätee tyhjän pinon automaatille M, se pätee myös ekvivalentille lopputilalliselle pinoautomaatille M' ($L(M) = L(M')$).

1)”$\Rightarrow$” Jos kieliopin G kuvaama kieli $L(G)$ on kontekstiton, niin on olemassa tyhjän pinon pinoautomaatti M, joka hyväksyy kielen $L(M) = L(G)$.

- Muodostetaan pinoautomaatti, jossa on kaksi tilaa: alkutila S ja hyväksyvä lopputila F.
- Aluksi pistetään pinoon erikoissymboli X (pinon pohja) ja siirrytään tilaan F.
- Tämän jälkeen on kaksi vaihtoehtoista askelta:
  1. Jos pinon päällä on välikesymboli C ja kieliopissa sääntö $C \to w$, niin korvaa C w:llä pinossa (ts. $pop(C), push(w)$).
  2. Jos pinon päällä on päätesymboli (merk. a), niin lue seuraava syötemerkki (merk. a'). Jos $a = a'$, niin lue pinon huippu pois ($pop(a)$).
- Automaatin siirtymät ovat siis:
  1. $\delta(S, \epsilon, \epsilon) = \{(F, X)\}$
  2. $\delta(F, \epsilon, C) = \{(F, w)\}$
  3. $\delta(F, a, a) = \{(F, \epsilon)\}$
- Intuitiivisesti: Automaatin laskenta simuloi syötejonon *vasenta johtoa*. Jokainen johtoaskel toteutetaan pistämällä pinoon säännön $S \to w$ oikea puoli w. Johtoaskelten välissä automaatti lukee päätesymbolit pinosta ja vertaa niitä syötteeseen. Kun säännön vasemman puoleinen välike C havaitaan pinossa, automaatti suorittaa seuraavan askeleen.
- Formaalissa todistuksessa pitäisi vielä osoittaa (induktiolla), että $w \in L(G) \Leftrightarrow w \in L(M)$.

2)”$\Leftarrow$ ” Jos olemassa tyhjän pinon pinoautomaatti M, joka hyväksyy kielen $L(M)$, niin on olemassa kontekstiton kielioppi G, joka kuvaa kielen $L(G) = L(M)$.

- Muodostetaan kielioppi, jonka jokainen välikesymboli vastaa ”tapahtumaa”, jossa
  1. luetaan jokin symboli X pinosta (voi koostua useasta siirtymästä)
  2. muutetaan tapahtumaa edeltävä tila p tapahtuman lopussa vallitsevaksi tilaksi q (symbolin X lukeminen pinosta tapahtuu siis siirtymäpolun $p - -- > q$ aikana)

- Merkitään tällaista tapahtumaa $[pXq]$:lla.
- Kieliopin muuttujat: alkusymboli S ja kaikki tapahtumasymbolit $[pXq]$, missä $p, q \in Q$ ja $X \in \Gamma$ (pinosymboli).
- Kieliopin säännöt:
  1. Jokaista tilaa $p \in Q$ kohden siirtymä $S \rightarrow [q_0 Z_0 p]$. (ts. symboli $[q_0 Z_0 p]$ tuottaa kaikki ne merkkijonot w, jotka aiheuttavat symbolin Z_0 nostamisen pinosta kuljettaessa tilasta q_0 tilaan p eli $(q_0, w, Z_0) \underset{M}{\vdash}^* (p, \epsilon, \epsilon)$.)
  2. Olkoon siirtymäfunktion $\delta(q, a, X)$ arvojoukossa pari $(r, Y_1 Y_2 ... Y_k)$, missä a on päätesymboli tai ϵ, ja $k \geq 0$. Silloin jokaista tilajoukkoa $r_1, r_2, ..., r_k$ kohti kieliopissa on sääntö

     $$[qXr_k] \rightarrow a[rY_1r_1][r_1Y_2r_2]...[r_{k-1}Y_kr_k].$$

     (t.s. yksi tapa lukea pinosta symboli X ja siirtyä tilasta q tilaan r_k on lukea ensin a, sitten käyttää syötettä Y_1:n poppaamiseksi siirtyen samalla tilaan r_1, käyttää syötettä Y_2:n poppaamiseksi siirtyen samalla tilaan r_2, jne.)
- Formaalissa todistuksessa pitäisi vielä osoittaa (induktiolla), että $[qXp] \underset{G}{\Rightarrow}^* w$ jos ja vain jos $(q, w, X) \underset{M}{\vdash}^* (p, \epsilon, \epsilon)$. Tällöin $S \underset{G}{\Rightarrow}^* w$ joss $[q_0 Z_0 p] \underset{G}{\Rightarrow}^* w$ jollain p joss $(q, w, Z_0) \underset{M}{\vdash}^* (p, \epsilon, \epsilon)$ eli $L(G) = L(M)$.

□

4.3.2 Deterministiset ja epädeterministiset pinoautomaatit

- Pinoautomaatti M on *deterministinen*, jos jokaisella tilanteella (q, w, α) on enintään yksi mahdollinen seuraaja (q', w', α'), jolla

$$(q, w, \alpha) \underset{M}{\vdash} (q', w', \alpha')$$

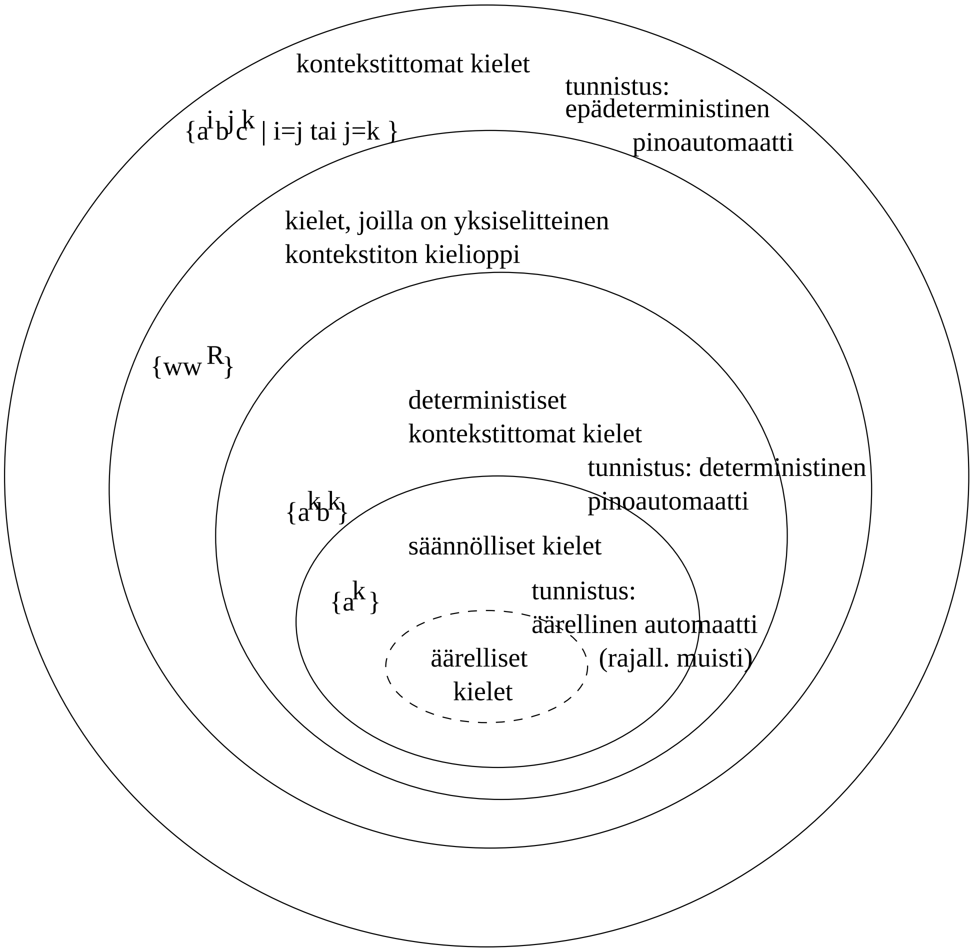
- Huom: *Epädeterministiset pinoautomaatit ovat aidosti vahvempia kuin deterministiset*! (vrt. äärelliset automaatit)

- Esim. kieli $\{ww^R \mid w \in \{a, b\}^*\}$ voidaan tunnistaa epädeterministisellä, mutta ei deterministisellä pinoautomaatilla (todistus sivuutetaan)

- Kontekstiton kieli on *deterministinen*, jos se voidaan tunnistaa jollakin deterministisellä pinoautomaatilla

- Deterministiset kielet ovat tärkeä kieliluokka, sillä siihen kuuluvat kielet voidaan jäsentää oleellisesti tehokkaammin kuin yleiset, mahdollisesti epädeterministisen automaatin vaativat kontekstittomat kielet

- Kielten keksinäisiä suhteita kuvaa seuraava kuva:


kontekstittomat kielet
tunnistus:
epädeterministinen
pinoautomaatti
{a^i b^j c^k | i=j tai j=k }
kielet, joilla on yksiselitteinen
kontekstiton kielioppi
{ww^R}
deterministiset
kontekstittomat kielet
tunnistus: deterministinen
pinoautomaatti
{a^k b^k}
säännölliset kielet
{a^k}
tunnistus:
äärellinen automaatti
(rajall. muisti)
äärelliset
kielet