Univerzalni Turingov stroj

Nora Berdalović

4. kolovoza 2025.

Definicija 1. Turingov stroj $\mathcal{U}$ je *univerzalan* ako za svaki Turingov stroj $\mathcal{T}$ nad abecedom Σ te ulaznu riječ $w \in \Sigma$, vrijedi $\mathcal{U}(\langle \mathcal{T}, w\rangle) = \mathcal{T}(w)$. $\triangleleft$

Teorem 2. *Neka je $\mathcal{T}$ Turingov odlučitelj s k traka koji radi u vremenu t. Tad postoji ekvivalentni dvotračni Turingov odlučitelj $\mathcal{T}'$ koji radi u vremenu $\mathcal{O}(t \cdot \log t)$. [3]*

Dokaz. Konstruiramo stroj $\mathcal{T}'$ tako da prva traka, koja ima k tragova, reprezentira k traka stroja $\mathcal{T}$, dok drugu traku koristimo kao pomoćnu pri simulaciji rada $\mathcal{T}$. Budući da prva traka ima samo jednu glavu, za simulaciju pomaka glave određene trake $\mathcal{T}$ pomiče se cijeli trag koji joj odgovara u $\mathcal{T}'$ lijevo ili desno. Pritom stroj mora prepisati svaki simbol na tragu, što zahtjeva linearno vrijeme, za svaki korak u $\mathcal{T}$ — dakle, kvadratno povećanje složenosti.

Kako bismo smanjili složenost, koristimo sljedeći trik: svaki trag u $\mathcal{T}'$ dijelimo na dva traga, s grupama R_i na prvom i L_i na drugom, tako da je $|R_1| = |L_1| = 1$ i $|R_i| = |L_i| = 2|R_{i-1}| = 2|L_{i-1}|$.

	L_1	L_2	L_3	L_4	...
0	R_1	R_2	R_3	R_4	...

Grupa 0 dimenzije 1 na početku drugog podtraga sadržava simbol na kojem se nalazi glava. Uvodimo novi simbol $\emptyset$ za prazan znak $\mathcal{T}'$ (prazan znak stroja $\mathcal{T}$ obični je znak za $\mathcal{T}'$) te simbol \$, koji će služiti za označavanje lijevog ruba trake. Za poziciju 0 uvijek vrijedi da je različita od $\emptyset$ i \$, kako bi ondje uvijek postojao simbol $\mathcal{T}$ za čitanje. Za svaki L_i i R_i vrijedi da je *prazan*, *pun* ili *poluprazan*: ili su svi simboli u grupi jednaki $\emptyset$ (ili \$), ili nijedan, ili ih je točno pola jednako $\emptyset$ (ili \$). Osim toga, zahtjevamo da je $R_i + L_i$ poluprazan, za svaki i.

Ako je na k-toj traci $\mathcal{T}$ bilo zapisano $\alpha_0\alpha_1\alpha_2\alpha_3\ldots$, s glavom na lijevom rubu, početni izgled k-tog traga $\mathcal{T}'$ je sljedeći:

	\$	$\emptyset\,\emptyset$	$\emptyset\,\emptyset\,\emptyset\,\emptyset$	...
α_0	α_1	$\alpha_2\alpha_3$	$\alpha_4\alpha_5\alpha_6\alpha_7$	...

Cijeli prvi podtrag je ispunjen praznim znakovima, osim prvog mjesta, gdje \$ označava da smo na lijevom rubu, dok su na drugi podtrag zapisani simboli s trake, podijeljeni u grupe R_i odgovarajućih dimenzija. Budući da su sve R_i pune, a L_i prazne, ispunjen je uvjet polupraznosti $R_i + L_i$.

U slučaju da se glava neke trake $\mathcal{T}$ miče desno, odgovarajući trag $\mathcal{T}'$ pomiče se „ulijevo". Ako je L_1 prazan, stroj može odmah simbol sa pozicije 0 prepisati u L_1, te simbol s R_1 na poziciju 0. Pritom, ako se u L_1 nalazio \$ znamo da smo bili na lijevom rubu, te su sve ostale ćelije na prvom podtragu prazne, pa zapisujemo \$ u L_2.

	α_0	$\$\emptyset$	$\emptyset\emptyset\emptyset\emptyset$	...
α_1	$\emptyset$	$\alpha_2\alpha_3$	$\alpha_4\alpha_5\alpha_6\alpha_7$	...

Ako je L_1 pun, stroj provodi sljedeći postupak:

1. Na drugu, pomoćnu traku zapisuje simbol iz 0 te separator.

2. Glavu pomiče udesno dok ne nađe najmanji i takav da R_i nije prazan. Označimo ga s R_m.

3. Prepisuje simbole iz R_m na drugu, pomoćnu traku iza separatora.

4. Prepisuje simbole s druge trake iza separatora redom u 0 te prazne grupe ćelija koje ga slijede. U slučaju da je R_m bio poluprazan, sadržavao je 2^{m-2} elemenata, koji se sad prepisuju u $0, R_1, R_2, \dots R_{m-2}$. Ako je R_m bio pun, imao je 2^{m-1} elemenata, te se oni prepisuju u $0, R_1, R_2, \dots R_{m-2}, R_{m-1}$.

5. Kako bi se očuvalo pravilo da je $R_i + L_i$ poluprazan za sve i, simbol iz 0 (kojeg čitamo s pomoćne trake) te simboli iz $L_1, \dots L_{m-1}$, kojih ukupno ima $1 + 1 + 2 + \dots + 2^{m-2} = 2^{m-1}$ prepisuju se u dalje ćelije, obrnutim redoslijedom. Budući da su $R_1, \dots R_{m-1}$ bili prazni, znamo da su $L_1, \dots L_{m-1}$ puni, te isto tako (zbog nepraznosti R_m) da je L_m prazan ili poluprazan. Ako je R_m bio poluprazan, takav je i L_m, dok su sad R_{m-1} i R_m prazni, pa za očuvanje pravila moramo popuniti L_{m-1} i L_m, u koje točno stane 2^{m-1} simbola iz prethodnih ćelija. Ako je R_m bio pun, tad je L_m prazan, pa u njega zapisujemo svih 2^{m-1} simbola. Pri završetku zapisa u L_i blokove, na prvo iduće prazno mjesto dodajemo \$ za oznaku lijevog ruba.

Na primjer, uzmimo sljedeći trag:

	α_4	$\alpha_3\alpha_2$	$\alpha_1\alpha_0\$\emptyset$	$\emptyset\emptyset\emptyset\emptyset\emptyset\emptyset\emptyset\emptyset$	...
α_5	$\emptyset$	$\emptyset\emptyset$	$\alpha_6\alpha_7\emptyset\emptyset$	$\alpha_8\alpha_9\alpha_{10}\alpha_{11}\alpha_{12}\alpha_{13}\alpha_{14}\alpha_{15}$	...

Nakon pomaka udesno, on bi izgledao ovako:

	$\emptyset$	$\alpha_5\alpha_4$	$\alpha_3\alpha_2\alpha_1\alpha_0$	$\$\emptyset\emptyset\emptyset\emptyset\emptyset\emptyset\emptyset$	...
α_6	α_7	$\emptyset\emptyset$	$\emptyset\emptyset\emptyset\emptyset$	$\alpha_8\alpha_9\alpha_{10}\alpha_{11}\alpha_{12}\alpha_{13}\alpha_{14}\alpha_{15}$	...

Pomak glave ulijevo je posve analogan, osim toga što se još provjerava nalazimo li se na lijevom rubu — ako je u L_1 simbol \$, micanje ulijevo znači ostajanje na mjestu i nemijenjanje traga.

Ovaj postupak odgovara jednom koraku stroja $\mathcal{T}$. Za njegovu simulaciju stroju $\mathcal{T}'$ treba $\sum_i \mathcal{O}(2^i)$ vremena, gdje i ide od 1 do $\log t$, budući da su blokovi R_i i L_i dimenzija 2^i te da ih ima maksimalno $\log t$ (inače bi na nekoj traci stroja $\mathcal{T}$ bilo zapisano više od t simbola, što je nemoguće ako stroj za rad treba t koraka). Par blokova R_i, L_i razmatramo najviše jednom u 2^{i-2} koraka. Dobivamo da za t' = ukupan broj koraka $\mathcal{T}'$ vrijedi:

$$\begin{aligned} t' &\leq \sum_{i=1}^{\lceil \log t \rceil} (\text{broj razmatranja } R_i, L_i) \cdot (\text{vrijeme potrebno za korak}) \\ &= \sum_{i=1}^{\lceil \log t \rceil} \frac{t}{2^{i-2}} \cdot \mathcal{O}(2^i) \end{aligned}$$

(budući da je $\frac{\mathcal{O}(2^i)}{2^{i-2}} = c$ konstanta)

$$\begin{aligned} &\leq \sum_{i=1}^{\lceil \log t \rceil} t \cdot \mathcal{O}(1) \\ &= \log t \cdot t \cdot c. \end{aligned}$$

Dakle, za novodobiveni $\mathcal{T}'$ vrijedi $time_{\mathcal{T}'} \in \mathcal{O}(\log t \cdot t \cdot c) = \mathcal{O}(\log t \cdot t)$, što je i trebalo dokazati. □

Teorem 3. *Postoji univerzalni Turingov stroj $\mathcal{U}$. Ako je $\mathcal{T}$ polinomno vremenski složen Turingov stroj, $\mathcal{U}$ restringiran na riječi koje počinju s $\langle \mathcal{T} \rangle$ također je polinomno vremenski složen. [2]*

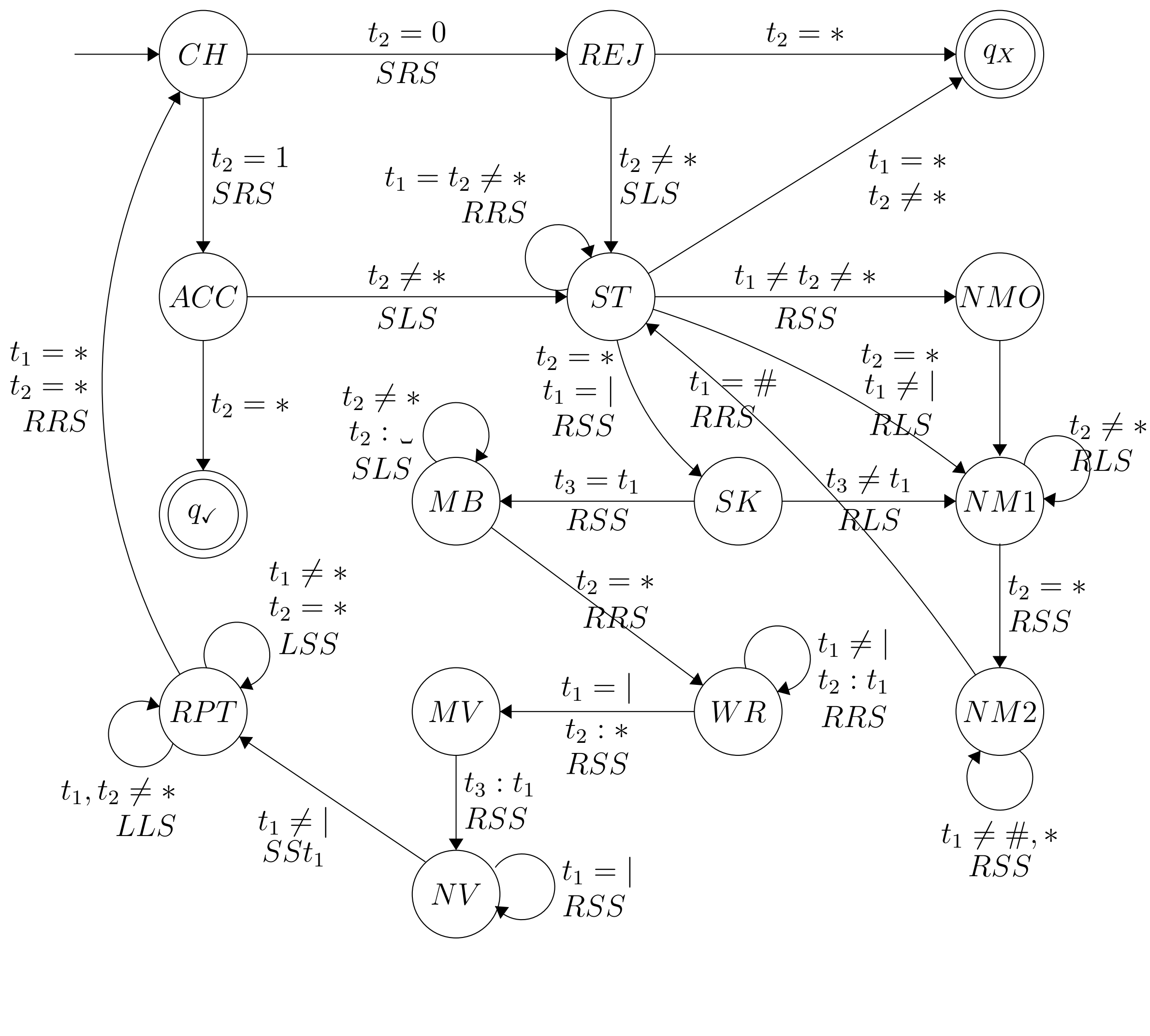
Dokaz. Neka je $\mathcal{T}$ proizvoljni Turingov stroj nad Σ i $w \in \Sigma$. Označimo duljinu koda od $\mathcal{T}$ s $p = |\langle \mathcal{T} \rangle|$, duljinu ulazne riječi w s n te broj koraka u $\mathcal{T}$-izračunavanju s w sa s. Ako pretpostavimo $\mathscr{L}(\mathcal{T}) \in P$, s je polinoman s obzirom na n. Konstruiramo Turingov stroj $\mathcal{U}$ koji za ulaz prima $\langle \mathcal{T}, w \rangle$ i simulira rad $\mathcal{T}$ nad w.

$\mathcal{U}$ će, osim ulazne, imati četiri dodatne trake. Na prvoj će se nalaziti popis prijelaza od $\mathcal{T}$, a na drugoj će biti trenutno stanje od $\mathcal{T}$ (koje će na početku odgovarati početnom stanju $\mathcal{T}$). Zapis jednog prijelaza je oblika $q_1\alpha_1q_2\alpha_2r$, pri čemu je q_1 trenutno stanje u kojem se čita znak α_1, a q_2 stanje u koje prelazi, zapisujući znak α_2 i pomičući glavu za r. Treća i četvrta traka će služiti za simulaciju rada $\mathcal{T}$. Na početku će na treću traku biti zapisana ulazna riječ w. U slučaju da je $\mathcal{T}$ višetračni stroj, prema teoremu 2 ga možemo transformirati u dvotračni uz povećanje složenosti. $\mathcal{T}$-izračunavanje će tada imati $s' \in \mathcal{O}(s \cdot \log s)$ koraka, što je i dalje polinomno.

Za simulaciju jednog koraka $\mathcal{T}$, $\mathcal{U}$ provodi sljedeći postupak. Prvo provjerava odgovara li trenutno stanje zapisano na drugoj traci jednom od završnih stanja $\mathcal{T}$. Ako odgovara $q_{\checkmark}$, riječ se prihvaća, a za q_X riječ se odbija. Ako trenutno stanje nije završno, ono se traži među prijelazima zapisanim na prvoj traci, kao stanje dano na početku nekog prijelaza. U slučaju da stanje nije pronađeno, riječ se odbija — ne postoji prijelaz kojim se iz trenutnog stanja može doći do $q_{\checkmark}$. Ako jest, $\mathcal{U}$ provjerava odgovara li znak koji se čita iz prijelaza znaku na trećoj traci na kojem se nalazi glava stroja. Ako nije, stroj nastavlja pretragu prve trake za valjanim prijelazom. Inače se obavlja dani prijelaz: stroj briše drugu traku te na nju zapisuje novo stanje, zapisuje novi znak na treću traku te pomiče glavu na trećoj traci lijevo ili desno, po pravilu prijelaza, uz postupak opisan u dokazu 2. Nakon toga ponovno ulazi u petlju i simulira idući korak.

Vidimo da se $\mathcal{U}$, prateći prijelaze od $\mathcal{T}$, može naći u stanju $q_{\checkmark}$ ako i samo ako se u njemu nađe $\mathcal{T}$. Dakle, $\mathcal{U}$ prihvaća riječ ako i samo ako ju prihvaća $\mathcal{T}$.

Simulator rada $\mathcal{U}$ dostupan je na [1]. Radi jednostavnosti, simulira samo jednotračne Turingove strojeve, pa su mu dovoljne tri pomoćne trake. Njegov je rad formalno zadan donjim dijagramom, izrađenim uz pomoć [4]. Dijagram opisuje rad nakon obavljene početne pripreme stroja, gdje su pomoćne trake t_1, t_2, t_3 odgovarajuće popunjene. Pritom je na početak i kraj zapisa na prvoj i drugoj traci dodano $*$, dok je između prijelaza na prvoj traci umjesto dvostrukog separatora $||$ zapisano $|\#$, radi lakše navigacije po trakama. Primjer izgleda traka te rada $\mathcal{U}$ može se vidjeti i u 4.



Znak koji se čita na određenoj traci na dijagramu je obilježen oznakom same trake, primjerice $t_2 = 0$ znači da je 0 znak na koji pokazuje glava na drugoj traci. Pomaci glava troje traka su označeni troslovima oblika SRL, gdje su S, R i L redom stajanje na mjestu te pomak desno odnosno lijevo. Sa SSt_1 označen je pomak na trećoj traci onako kako nalaže znak koji se čita s prve trake. Pisanje po traci naznačeno je s npr. $t_2 : t_1$ (zapiši na drugu traku znak koji se čita s prve). Prihvaćajuće stanje $q_\checkmark$ kodirano je s 1, a odbijajuće q_X s 0.

Pogledajmo vremensku složenost stroja $\mathcal{U}$. Za početnu pripremu stroja potrebno je iščitati prijelaze i početno stanje $\mathcal{T}$ te riječ w iz ulaza $\langle \mathcal{T}, w\rangle$ i zapisati ih na odgovarajuće pomoćne trake. Za to je dovoljan jedan prolaz po ulaznoj traci, što je (zbog 4 separatora između $\langle \mathcal{T}\rangle$ i $\langle w\rangle$) $p + 4 + n + 1$ koraka.

Prilikom simulacije jednog koraka $\mathcal{T}$, za provjeru je li trenutno stanje završno potrebno je 2 koraka (čitanje s druge trake). Nakon toga, pretražuje se prva traka, gdje zapisani prijelazi zauzimaju manje od $p = |\langle \mathcal{T}\rangle|$ ćelija. Svaki put kad se nađe trenutno stanje kao početno u nekom prijelazu, obavlja se usporedba znaka za čitanje iz prijelaza i znaka na trećoj traci, u jednom koraku. Za prolaz prvom trakom potrebno je najviše p koraka, te se obavlja maksimalno p usporedbi, pa je potreban broj koraka najviše $2p$.

Ako je nađen odgovarajući prijelaz, za brisanje trenutnog stanja s treće trake potrebno je manje od p koraka (budući da je to stanje dio koda $\langle \mathcal{T}\rangle$ duljine p), te opet manje od p koraka za zapisivanje novog stanja. Za pisanje znakova na treću traku te pomak glava potrebno je 2 koraka. Nakon toga, prva i druga traka se „premotavaju" na početak, za što je potrebno najviše p pomaka obje glave.

Sveukupno, za simulaciju jednog koraka $\mathcal{T}$ potrebno je $2 + 2p + p + p + 2 + p = 5p + 4$ koraka, što je u $\mathcal{O}(p)$. Kao što smo rekli, za početnu pripremu $\mathcal{U}$ treba $p + n + 5$ koraka. Budući da $\mathcal{T}$ radi u s' koraka, a n je duljina njegove ulazne riječi w, vrijedi $p + n + 5 \leq p+s'+5$. Dakle, složenost $\mathcal{U}$ je $(p+s'+5)+s'\cdot\mathcal{O}(p) \leq (s'+\mathcal{O}(p))+s'\cdot\mathcal{O}(p) \leq (2s'+1)\cdot\mathcal{O}(p)$. Ako restringiramo rad $\mathcal{U}$ na riječi koje počinju s $\langle \mathcal{T}\rangle$, p je kao duljina tog koda fiksan — dakle, broj koraka $\mathcal{U}$ je proporcionalan s', pa je $\mathcal{L}(\mathcal{U}|_\mathcal{T}) \in P$.

$\square$

Primjer 4. Pogledajmo kako $\mathcal{U}$ simulira jednostavni Turingov stroj $\mathcal{T}$ koji odlučuje jezik 0^* nad abecedom $\Sigma = \{0, 1\}$. Skup stanja $\mathcal{T}$ je $Q = \{q_X, q_\checkmark, q_0\}$, a stanjima su redom pridruženi kodovi $0, 1$ i 10. Početno stanje je q_0. Funkcija prijelaza je $\delta = \{(q_0, 0, q_0, 0, R), (q_0, \textvisiblespace, q_\checkmark, \textvisiblespace, R), (q_0, 1, q_X, 1, R)\}$, pri čemu je $\textvisiblespace$ prazan znak. U tablici je prikazan tijek rada $\mathcal{U}$ za $\mathcal{T}$ s ulaznom riječju $w = 00$. Položaj glave na svakoj traci naznačen je podcrtanim znakom.

trenutno stanje	prva traka	druga traka	treća traka
CH	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$\underline{0}0$
ACC	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$\underline{0}0$
ST	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$\underline{0}0$
ST	$\ast 1\underline{0}\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$\underline{0}0$
ST	$\ast 10\underline{\vert}0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$\underline{0}0$
SK	$\ast 10\vert\underline{0}\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$\underline{0}0$
MB	$\ast 10\vert 0\underline{\vert}10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$\underline{0}0$
MB	$\ast 10\vert 0\underline{\vert}10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\underline{\ast}$	$\underline{0}0$
WR	$\ast 10\vert 0\vert\underline{1}0\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{\textvisiblespace}$	$\underline{0}0$
WR	$\ast 10\vert 0\vert 1\underline{0}\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{\textvisiblespace}$	$\underline{0}0$
WR	$\ast 10\vert 0\vert 10\underline{\vert}0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\textvisiblespace}$	$\underline{0}0$
MV	$\ast 10\vert 0\vert 10\vert\underline{0}\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$\underline{0}0$
NV	$\ast 10\vert 0\vert 10\vert 0\underline{\vert}1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$\underline{0}0$
NV	$\ast 10\vert 0\vert 10\vert 0\vert\underline{1}\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$\underline{0}0$
RPT	$\ast 10\vert 0\vert 10\vert 0\vert\underline{1}\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$0\underline{0}$
. . . vraćanje glava na početak t_1 i t_2 . . .			
CH	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$0\underline{0}$
. . . analogni postupak . . .			
CH	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$00\underline{\textvisiblespace}$
ACC	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$00\underline{\textvisiblespace}$
ST	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$00\underline{\textvisiblespace}$
ST	$\ast 1\underline{0}\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$00\underline{\textvisiblespace}$
ST	$\ast 10\underline{\vert}0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$00\underline{\textvisiblespace}$
SK	$\ast 10\vert\underline{0}\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 10\underline{\ast}$	$00\underline{\textvisiblespace}$
$NM1$	$\ast 10\vert 0\underline{\vert}10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{0}\ast$	$00\underline{\textvisiblespace}$
$NM1$	$\ast 10\vert 0\vert\underline{1}0\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$00\underline{\textvisiblespace}$
$NM1$	$\ast 10\vert 0\vert 1\underline{0}\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\underline{\ast}10\ast$	$00\underline{\textvisiblespace}$
$NM2$	$\ast 10\vert 0\vert 1\underline{0}\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\underline{\ast}10\ast$	$00\underline{\textvisiblespace}$
. . . pomak glave do idućeg prijelaza na t_1 . . .			
ST	$\ast 10\vert 0\vert 10\vert 0\vert 1\vert\#\underline{1}0\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}0\ast$	$00\underline{\textvisiblespace}$
. . . analogni postupak . . .			
CH	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast\underline{1}\ast$	$00\textvisiblespace\underline{\textvisiblespace}$
ACC	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{\ast}$	$00\textvisiblespace\underline{\textvisiblespace}$
$q_{\checkmark}$	$\ast\underline{1}0\vert 0\vert 10\vert 0\vert 1\vert\#10\vert\textvisiblespace\vert 1\vert\textvisiblespace\vert 1\vert\#10\vert 1\vert 0\vert 1\vert 1\ast$	$\ast 1\underline{\ast}$	$00\textvisiblespace\underline{\textvisiblespace}$

Literatura

[1] Nora Berdalović. *Simulator univerzalnog Turingovog stroja*. 2025. URL: https://github.com/ring-bearer/univ.

[2] Peter Gács i László Lovász. *Complexity of Algorithms*. 1999.

[3] Peeter Laud. *Complexity Theory: Lecture 3*. 2011. URL: https://research.cyber.ee/~peeter/teaching/keerukus11s/Lecture3.pdf.

[4] Evan Wallace. *Finite State Machine Designer*. 2010. URL: https://madebyevan.com/fsm/.