Zadanie:

Definujte viacpáskový Turingov stroj (TS) a formálne dokážte, že jednopáskový Turingov stroj vie odsimulovat viacpáskové TS. Teda, že pre každý k-páskový TS T, existuje 1-páskový TS T' taký, že L(T) = L(T').

Definície:

Definícia 1:
Nedeterministický k-páskový TS je 6-tica $T = (K, \Sigma, \Gamma, \delta, q_0, F)$, kde:
K - konečná množina stavov
Σ - vstupná abeceda
Γ - abeceda pracovných symbolov, ktorá obsahuje špeciálny symbol B (Blank), tj. $B \in \Gamma$, pričom $B \notin \Sigma$ a $\Sigma \subset \Gamma$
q_0 - počiatočný stav, pričom $q_0 \in K$
F - množina akceptačných stavov, pričom $F \subseteq K$
δ - prechodová funkcia: δ: $K \times \Gamma \times \Gamma \times \dots \times \Gamma \to 2^{K \times (\Gamma - \{B\}) \times \{-1, 0, 1\} \times \dots \times (\Gamma - \{B\}) \times \{-1, 0, 1\}}$
(k-krát Γ) (k-krát $^{(\Gamma - \{B\}) \times \{-1, 0, 1\}}$)

Definícia 2:
Konfigurácia nedeterministického k-páskového TS T je 2k+1-tica $(q, \alpha_1, i_1, \dots, \alpha_k, i_k)$, kde $q \in K \wedge \forall x; x \in 1..k : \alpha_x \in (\Gamma - \{B\})^* \wedge i_x \in 0..|\alpha_x| + 1$

Poznámka:
Zrejme prázdne pásky môžeme zapísať viacerými spôsobmi, teda prázdnou páskou, pričom hlava je kdekoľvek, lebo prechodová δ funkcia nie je závislá na pozícii hlavy, iba na symbole, na ktorý ukazuje, čo však je vždy blank B.

Definícia 3:
Krok výpočtu nedeterministického k-páskového TS T je relácia $\vdash_T$ na konfiguráciach definovaná nasledovne:
$(q, \alpha_1, i_1, \dots, \alpha_k, i_k) \vdash_T (p, \beta_1, j_1, \dots, \beta_k, j_k) \Leftrightarrow$
$\Big(\exists u_1, u_2, \dots u_k \in \Gamma \wedge \exists v_1, v_2, \dots v_k \in (\Gamma - \{B\}) \wedge \exists s_1, s_2, \dots s_k \in \{-1, 0, 1\}$ také, že $\Big($
$(p, v_1, s_1, v_2, s_2, \dots v_k, s_k) \in \delta(q, u_1, u_2, \dots u_k) \wedge \forall d; d \in 1..k$ platí: $\Big[$
$(i_d = 0 \wedge \beta_d = v_d\alpha_d \wedge u_d = B \wedge s_d = j_d - 1)$
$\vee (0 < i_d < |\alpha_d| + 1 \wedge (\forall m \in 1..|\alpha_d| : m \neq i_d \Rightarrow \alpha_d[m] = \beta_d[m]) \wedge \alpha_d[i_d] = u_d \wedge \beta_d[i_d] = v_d \wedge s_d = j_d - i_d)$
$\vee (i_d = |\alpha_d| + 1 \wedge u_d = B \wedge \beta_d = \alpha_d v_d \wedge s_d = j_d - |\beta_d|)$
$\Big] \Big) \Big)$

Definícia 4:
Jazyk akceptovaný nedeterministickým k-páskovým TS T je
$L(T) = \{ w \in \Sigma^* \mid (q_0, w, 1, \varepsilon, 0, ..., \varepsilon, 0) \vdash^*_T (q, \alpha_1, i_1, \dots, \alpha_k, i_k) \wedge q \in F\}$
(k-1 krát $\varepsilon, 0$)

Neformálne Riešenie:

T' zostrojíme ako 2k stopový, dve stopy pre každú pásku stroja T. Jedna stopa zaznamenáva obsah zodpovedajúcej pásky stroja T, a v druhej má označené to políčko, na ktorom je zodpovedajúca hlava v stroji T. Riadiaca jednotka T' si okrem stavu povodného automatu pamätá i pozície značiek, t.j. pre každú hlavu v stroji T či je naľavo alebo napravo od hlavy stroja T'.
Na odsimulovanie jedného kroku stroja T musí T' prezrieť (a zapamätať si) všetky označené políčka (t.j. zistiť čo snímajú jednotlivé hlavy v stroji T). Pri prechádzaní cez značku si musí vhodne upraviť informáciu o smere, kde je značka pre tú konkrétnu pásku stroja T vzhladom na hlavu stroja T'. Po zistení všetkých symbolov, ktoré by čítal povodný automat, zistí, ktorý krok by urobil stroj T a realizuje ho tak, že znova prejde všetky označené políčka, zmení čítané symboly a posunie značky podľa toho, ako sa posúvali hlavy v stroji T. Samozrejme, ak stroj T v novom stave akceptuje, akceptuje aj stroj T'.

Ukážka pásky stroja T' („-“ = B = Blank):

Hlava 1	-	+	-	-	-	
Páska 1	A1	A2	A3	A4	A5	...
Hlava 2	-	-	-	-	+	
Páska 2	-	-	B1	B2	B3	...
Hlava 3	+	-	-	-	-	
Páska 3	C1	C2	C3	C4	C5	...
Hlava 4	-	-	-	+	-	
Páska 4	-	-	-	D1	D2	...

Formálne riešenie:

Konštrukcia:

Bez újmy ma všeobecnosti predpokladajme, že
$K \cap (\Sigma \cup K' \cup \Gamma \cup \Gamma') = \{\}$
$\Gamma \cap (K \cup K') = \{\}$
$\Gamma \cap \Gamma' = \Sigma$

Požadovaný automat T' zostrojíme nasledovne:
$T' = (K', \Sigma, \Gamma', \delta', q_0', F')$, kde:
$K' = \{q_{accept}, q_0', q_A, q_B, q_C\}$
$\cup \{q[p, i_1, \dots, i_k, a_1, \dots, a_k, n]\ \forall p \in K, \forall i_1, \dots, i_k \in \{-1, 1\}, \forall a_1, \dots, a_k \in \Gamma \cup \{B\}, \forall n \in 1..k+1 \}$
$\cup \{ q[p, i_1, \dots, i_k, c_1, \dots, c_k, s_1, \dots, s_k, n], q_{posun}[p, i_1, \dots, i_k, c_1, \dots, c_k, s_1, \dots, s_k, n]$
$\forall p \in K, \forall i_1, \dots, i_k \in \{-1, 1\}, \forall a_1, \dots, a_k \in \Gamma \cup \{B\}, \forall n \in 1..k+1, \forall s_1, \dots, s_k \in \{-1, 0, 1\} \}$
$\Gamma' = \Sigma \cup \{B\} \cup \{(h_1, x_1, ..., h_k, x_k)\ \forall x_i \in \Gamma \cup \{B\}\ \forall h_i \in \{+,-\} \}$
$F' = \{q[p, i_1, \dots, i_k, a_1, \dots, a_k, 1]\ \forall p \in F, \forall i_1, \dots, i_k \in \{-1, 1\}, \forall a_1, \dots, a_k \in \Gamma \cup \{B\} \}$

Prechodová funkcia:

1. Vytvorenie 2k stopej pásky s označenými pozíciami hláv v pôvodnom TS T a súčastne vygenerovanie dostatku poschodových blankov B na obe strany od vstupného slova.

P1:
Máme dve možnosti:
1) $\varepsilon \in L(T)$
2) $\varepsilon \notin L(T)$

1: prípad, že neakceptuje prázdne slovo
$\delta'(q_0', x) = \{(q_A, (+, x, +, B, ..., +, B), 1)\}\ \forall x \in \Sigma$ (k-1 krát ,+, B)
- vytvorenie 1. poschodového znaku, nad ktorým sú všetky značky (hlavy pôvodného TS T)
- v prípade, že nájde B, tak sa zasekne – aj tak ho nechceme akceptovat!

2: prípad, že akceptuje prázdne slovo
$\delta'(q_0', B) = \{(q_{accept}, (+, B, +, B, ..., +, B), 1)\}$ (k krát ,+, B)
- našiel B ako prvý znak, t.j. vstup je prázdny, t.j. akceptujeme
$\delta'(q_0', x) = \{(q_A, (+, x, +, B, ..., +, B), 1)\}\ \forall x \in \Sigma$ (k-1 krát ,+, B)
- vytvorenie 1. poschodového znaku, nad ktorým sú všetky značky (hlavy pôvodného TS T)

Pozn.: Ďalej už nie sú rozdiely medzi týmito dvoma možnosťami.

P2:
$\delta'(q_A, x) = \{(q_A, (-, x, -, B, ..., -, B), 1)\}\ \forall x \in \Sigma$ (k-1 krát ,-, B)
- vytvorenie k-poschodovej pásky zo vstupných symbolov

P3:
$\delta'(q_A, B) = \{(q_A, (-, B, \dots, -, B), 1), (q_B, (-, B, \dots, -, B), 0)\}$ (k krát ,-, B) , (k krát ,-, B)
- vytvorenie poschodového blank-u za poschodovým vstupom
- nedeterministické rozhodnutie, že toľko blank-ov napravo stačí a návrat cez vytvorené blanky späť

P4:
$\delta'(q_B, (-, x, \dots, -, B)) = \{(q_B, (-, x, \dots, -, B), -1)\}\ \forall x \in \Sigma \cup \{B\}$ (k-1 krát ,-,B) ,(k-1 krát ,-,B)
- návrat späť cez vytvorený poschodový vstup a poschodové blanky

P5:
$\delta'(q_B, (+, x, +, B \dots, +, B)) = \{(q_B, (+, x, \dots, +, B), -1)\}\ \forall x \in \Sigma$ (k-1 krát ,+,B),(k-1 krát ,+,B)
- návrat späť cez vytvorený poschodový vstup a poschodové blanky

P6:
$\delta'(q_B, B) = \{(q_B, (-, B, \dots, -, B), -1), (q_C, (-, B, \dots, -, B), 0)\}$ (k krát ,-, B) , (k krát ,-, B)
- vytvorenie poschodového blank-u pred poschodovým vstupom
- nedeterministické rozhodnutie, že toľko blank-ov naľavo stačí, a návrat cez vytvorené blanky späť

P7:
$\delta'(q_C, (-, B, \dots, -, B)) = \{(q_C, (-, B, \dots, -, B), 1)\}$ (k krát ,-,B)
- návrat na poschodový vstup

P8:

$\delta'(q_C, (+, x, +, B\ldots, +, B)) = \{(q[q_0, 1, ..., 1, B, ..., B, 1], (+, x, +, B\ldots, +, B), 0)\}\ \forall x \in \Sigma \cup \{B\}$
(k-1 krát ,+,B), (k krát ,1), (k krát ,B), (k-1 krát ,+,B)
- nájdenie začiatku vstupu ➔ začiatok simulácie

---

vysvetlenie označenia stavu:
$q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, n]$
p - stav v TS T
$i_1, \ldots, i_k$ - pozície hláv stroja T ozhľadom na pozíciu hlavy stroja T'
$a_1, \ldots, a_k$ - načítané označené symboly
n - číslo „pásky“, na ktorej sa hľadá označený symbol

---

2. Načítanie označených symbolov, aby sme sa mohli podľa nich rozhodnúť ako stroj T.

P9:
$\delta'(q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, j], (h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k)) =$
$\{(q[p, (i_1\ (h_1 = -)) + (-i_j\ (h_1 = +)), \ldots, (i_k\ (h_k = -)) + (-i_j\ (h_k = +)), a_1, \ldots, a_k, j],$
$(h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k), i_j)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall j \in \{1, 2, \ldots, k\},$
$\forall h_1, \ldots, h_{j-1}, h_{j+1}, \ldots, h_k \in \{+,-\}, \forall d_1, \ldots, d_k \in \Gamma$
- nie sme na pozícii, kde je hlava j-tej pásky, teda posúvame sa ďalej smerom kde je hľadaná hlava, pričom ak prejdeme nejakou hľavou, tak si upravíme jej polohu

P10:
$\delta'(q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, j], (h_1, d_1, \ldots, h_{j-1}, d_{j-1}, +, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k)) =$
$\{(q[p, (i_1\ (h_1 = -)) + (-i_j\ (h_1 = +)), \ldots, (i_k\ (h_k = -)) + (-i_j\ (h_k = +)), a_1, \ldots, a_{j-1}, d_j, a_{j+1}, \ldots, a_k, j+1],$
$(h_1, d_1, \ldots, h_{j-1}, d_{j-1}, +, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k), 0)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall j \in \{1, 2, \ldots, k\},$
$\forall h_1, \ldots, h_{j-1}, h_{j+1}, \ldots, h_k \in \{+,-\}, \forall d_1, \ldots, d_k \in \Gamma$
- našli sme hľadanú hlavu, teda si zapamätáme symbol a ideme hľadať ďalší symbol pre ďalšiu hlavu

3. Máme načítané všetky symboly pod hlavami, teda T' sa môže rozhodnúť ako stroj T.

P11:
$\delta'(q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, k+1], \alpha) =$
$\{(q[p_0, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, 1], \alpha, 0) \mid \delta(p, a_1, \ldots, a_k) \ni (p_0, c_1, s_1, \ldots, c_k, s_k)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall \alpha \in \Gamma'-\{B\}$

---

vysvetlenie označenia stavu:
$q[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, n]$
p_0 – nový stav v TS T
$i_1, \ldots, i_k$ - pozície hláv stroja T ozhľadom na pozíciu hlavy stroja T'
$c_1, \ldots, c_k$ - aké symboly sa zapíšu
$s_1, \ldots, s_k$ - ktorým smerom sa posunú hlavy
n - číslo „pásky“, na ktorej sa pracuje

---

4. Teraz už vieme, ktoré znaky pod „hlavami“ treba za aké nahradiť, a tiež vieme, kam posunúť hlavy, aby sme odsimulovali TS T

P12:

$\delta'(q[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, j], (h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k)) =$
$\{(q[p, (i_1 (h_1 = -)) + (-i_j (h_1 = +)), \ldots, (i_k (h_k = -)) + (-i_j (h_k = +)), c_1, \ldots, c_k, s_1, \ldots, s_k, j],$
$(h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k), i_j)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall j \in \{1, 2, \ldots, k\},$
$\forall h_1, \ldots, h_{j-1}, \ldots, h_{j+1}, \ldots, h_k \in \{+,-\}, \forall d_1, \ldots, d_k \in \Gamma, \forall c_1, \ldots, c_k \in \Gamma, \forall s_1, \ldots, s_k \in \{-1, 0, 1\}$
- nie sme na pozícii, kde je hlava j-tej pásky, teda posúvame sa ďalej smerom, kde je hľadaná hlava, pričom, ak prejdeme nejakou hľavou, tak si upravíme jej polohu

P13:

$\delta'(q[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, j], (h_1, d_1, \ldots, h_{j-1}, d_{j-1}, +, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k)) =$
$\{(q_{posun}[p, (i_1 (h_1 = -)) + (-s_j (h_1 = +)), \ldots, (i_k (h_k = -)) + (-s_j (h_k = +)), c_1, \ldots, c_k, s_1, \ldots, s_k, j],$
$(h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, c_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k), s_j)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall j \in \{1, 2, \ldots, k\},$
$\forall h_1, \ldots, h_{j-1}, \ldots, h_{j+1}, \ldots, h_k \in \{+,-\}, \forall d_1, \ldots, d_k \in \Gamma, \forall c_1, \ldots, c_k \in \Gamma, \forall s_1, \ldots, s_k \in \{-1, 0, 1\}$
- našli sme hľadanú hlavu, teda zapíšeme symbol a ideme posunúť hlavu v stave q_{posun}

---

vysvetlenie označenia stavu:
$q_{posun}[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, n]$
p_0 – nový stav v TS T
$i_1, \ldots, i_k$ - pozície hláv stroja T ozhľadom na pozíciu hlavy stroja T'
$c_1, \ldots, c_k$ - aké symboly sa zapíšu
$s_1, \ldots, s_k$ - ktorým smerom sa posunú hlavy
n - číslo „pásky“, na ktorej sa pracuje

P14:

$\delta'(q_{posun}[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, j], (h_1, d_1, \ldots, h_{j-1}, d_{j-1}, -, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k)) =$
$\{(q[p, (i_1 (h_1 = -)) + (-s_j (h_1 = +)), \ldots, (i_k (h_k = -)) + (-s_j (h_k = +)), c_1, \ldots, c_k, s_1, \ldots, s_k, j+1],$
$(h_1, d_1, \ldots, h_{j-1}, d_{j-1}, +, d_j, h_{j+1}, d_{j+1}, \ldots, h_k, d_k), 0)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall j \in \{1, 2, \ldots, k\},$
$\forall h_1, \ldots, h_{j-1}, \ldots, h_{j+1}, \ldots, h_k \in \{+,-\}, \forall d_1, \ldots, d_k \in \Gamma, \forall c_1, \ldots, c_k \in \Gamma, \forall s_1, \ldots, s_k \in \{-1, 0, 1\}$
- vytvorenie novej značky pre hlavu, a nadstavenie na update ďalšej dvojice stôp

P15:

$\delta'(q[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, k+1], \alpha) = \{(q[p, i_1, \ldots, i_k, B, \ldots, B, 1], \alpha, 0)\}$
$\forall p \in K,\ \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall c_1, \ldots, c_k \in \Gamma, \forall s_1, \ldots, s_k \in \{-1, 0, 1\}; \forall \alpha \in \Gamma'-\{B\}$
- po posunutí poslednej hlavy sme odsimulovali 1 krok stroja T, a teda môžeme ísť simulovať ďalší

<u>Dôkaz:</u>

Musíme ukázať, že T' akceptuje všetky slová, ktoré akceptuje T a ďalej, že neakceptuje nič viac, resp. že všetko čo akceptuje, akceptuje aj T.

Prípad pre **w** = ε je vyriešený v P1 definície δ', lebo ak ε ∈ L(T), tak na 1 krok v T' prejdeme do akceptačného stavu. Ak ε ∉ L(T), tak sa stroj T' pri ε vstupe zasekne, čo sme chceli.

Teraz rozoberme prípad **w** ≠ ε.

V ďalšom budeme o všetkých konfiguráciách stroja T’ predpokladať, že sú zdravé, t.j. pre konfigurácie tvarov:

Typ 1: $(q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, n], \alpha, i)$
Typ 2: $(q[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, n], \alpha, i)$
Typ 3: $(q_{posun}[p, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, n], \alpha, i)$

kde $\alpha = (h_{1,1}, d_{1,1}, \ldots, h_{k,1}, d_{k,1}) \ldots (h_{1,n}, d_{1,n}, \ldots, h_{k,n}, d_{k,n})$

platí:
$\forall x \in 1..k$:
1) $1 = \Sigma_{y=1..n}\ (h_{x,y} = +)$
2) $h_{x,i} = - \Rightarrow \exists\ z\ ;\ (i_x = 1 \Rightarrow z > i \wedge h_{x,z} = +) \wedge (i_x = -1 \Rightarrow z < i \wedge h_{x,z} = +)$

Ešte si dokážme, že toto naozaj platí:

<u>Základ indukcie:</u>
Z počiatočnej konfigurácie
$(q_{0'}, (w_1, \ldots, w_{|w|}), 1)$
sa na 1 krok použitím P1 dostávame do konfigurácie
$(q_A, ((+, w_1, +, B, \ldots, +, B)\ w_2, \ldots, w_{|w|}), 2)$
ďalej |w|-1 násobným použitím P2 sa dostávame do konfigurácie
$(q_A, ((+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)), |w|+1)$
ďalej „right“ násobným použitím P3.1 sa dostávame do konfigurácie
$(q_A, ((+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), |w|+1+„right”)$ „right” krát $(-, B, \ldots, -, B)$
ďalej použitím P3.2 sa dostávame do konfigurácie
$(q_B, ((+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), |w|+1+„right”)$ „right”+1 krát $(-, B, \ldots, -, B)$
ďalej „right“+|w| násobným použitím P4 sa dostávame do konfigurácie
$(q_B, ((+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), 1)$ „right”+1 krát $(-, B, \ldots, -, B)$
ďalej použitím P5 sa dostávame do konfigurácie
$(q_B, ((+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), 0)$ „right”+1 krát $(-, B, \ldots, -, B)$
ďalej „left“ násobným použitím P6.1 sa dostávame do konfigurácie
$(q_B, ((-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)$ „left” krát $(-, B, \ldots, -, B)$
$(+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), 0)$ „right”+1 krát $(-, B, \ldots, -, B)$
ďalej použitím P6.2 sa dostávame do konfigurácie
$(q_C, ((-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)$ „left”+1 krát $(-, B, \ldots, -, B)$
$(+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), 1)$ „right”+1 krát $(-, B, \ldots, -, B)$
ďalej „left“+1 násobným použitím P7 sa dostávame do konfigurácie
$(q_C, ((-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)$ „left”+1 krát $(-, B, \ldots, -, B)$
$(+, w_1, +, B, \ldots, +, B)\ (-, w_2, -, B, \ldots, -, B) \ldots (-, w_{|w|}, -, B, \ldots, -, B)$
$(-, B, \ldots, -, B) \ldots (-, B, \ldots, -, B)), „left“+2)$ „right”+1 krát $(-, B, \ldots, -, B)$

ďalej použitím P8 sa dostávame do konfigurácie

$(q[q_0, 1, ..., 1, B, ..., B, 1],$

$((-, B, ..., -, B) ... (-, B, ..., -, B)$ „left"+1 krát $(-, B, ..., -, B)$

$(+, w_1, +, B, ..., +, B) (-, w_2, -, B, ..., -, B)... (-, w_{|w|}, -, B, ..., -, B)$

$(-, B, ..., -, B) ... (-, B, ..., -, B)),$ „left"+2) „right"+1 krát $(-, B, ..., -, B)$

čo je zdravá konfigurácia, a pritom iné kroky neboli v jednotlivých konfiguráciách možné, teda každý akceptačný výpočet takto postupoval.

Indukčný krok:

predpokladajme, že to platí pre konfiguráciu A, ukážeme, že to platí i pre všetky možné konfigurácie B také, že $A \vdash_T B$.

máme 3 možnosti konfigurácie A:

1) A je typu $(q[p, i_1, \dots, i_k, a_1, \dots, a_k, j], \alpha, i)$. Potom môžeme použiť pravidlá:

P9: podmienku 1) triviálne neporuší

podmienku 2) neporuší, pretože nový stav je

$q[p, (i_1 (h_1 = -)) + (-i_j (h_1 = +)), \dots, (i_k (h_k = -)) + (-i_j (h_k = +)), a_1, \dots, a_k, j]$

a hlava sa pohla smerom i_j

teda $\forall x \in 1..k$:

ak $h_x = +$, tak $i_x = -i_j$ – tj. podmienka 2) platí, lebo sme zaznačili, že značka je v opačnom smere ako sme pohli hlavou, čo skutočne je, kedže bola pod hlavou – čo je podmienka 2)

ak $h_x = -$, tak $i_x = i_x$ – tj. podmienka 2) platí, lebo sme zaznačili, že značka je tým istým smerom, ako bola, teda platí podmienka 2), pretože značka nebola pod hlavou, a teda pri posunutí sme ju buď dosiahli, alebo je stále tým istým smerom

P10: podmienky triviálne platia, pretože neposúvame značky a hlavu tiež neposúvame

P11: ako P10

Pre možnosti P12-P15 platí obdobne ako pre P9

Koniec dôkazu o zdravých konfiguráciách

<u>Definícia:</u> Konfigurácia $(q, \alpha_1, i_1, ..., \alpha_k, i_k)$ stroja T je <u>pridružená</u> konfigurácii (p, β, j) stroja T' práve vtedy, keď

$\exists j_1, ..., j_k, a_1, ..., a_k : p = q[q, j_1, ..., j_k, a_1, ..., a_k, 1] \wedge$

$\beta = h_{1,1}\, b_{1,1} ... h_{1,n}\, b_{1,n}\, h_{2,1}\, b_{2,1} ... h_{k,n}\, b_{k,n} \wedge$

$\forall x \in 1..k : \exists u,v : ($ - pre každú simulovanú pásku

$1 = \Sigma_{y=1..n} (h_{x,y} = +) \wedge$ - na každej simulovanej páske je práve 1 hlava

$\alpha_x = b_{x,u}\, b_{x,u+1} ... b_{x,v} \wedge$ - x-tá páska je zhodná so stopou, kt. ju predstavuje

$b_{x,u} \neq B \wedge [\ \forall m \in 1..n : m<u \Rightarrow b_{x,m} = B\ ] \wedge$ - u je prvý platný index

$b_{x,v} \neq B \wedge [\ \forall m \in 1..n : v<m<n+1 \Rightarrow b_{x,m} = B\ ] \wedge$ - v je posledný platný index

$\forall y \in 1..n : \quad h_{x,y} = + \Rightarrow y-u+1 = i_x$ - pozícia hláv

$)$

<u>**L(T) ⊂ L(T')**</u>

Nech $w \in L(T)$ a $w \neq \varepsilon$, teda existuje akceptačný výpočet w v TS T, t.j.

$(q_0, w, 1, \varepsilon, 0, ..., \varepsilon, 0) \vdash^*_T (q, \alpha_1, i_1, \dots, \alpha_k, i_k)$, kde $q \in F$.

Predpokladajme nasledovné tvrdenia, ktoré neskôr dokážeme:

<u>Tvrdenia:</u>

1. Zo začiatočnej konfigurácie stroja T' sa vieme dostať do takej konfigurácie, ktorá bude pridružená začiatočnej konfigurácii stroja T.

2. Nech A, B sú konfigurácie stroja T a C konfigurácia stroja T' také, že A $\vdash_T$ B a A je pridružená s C. Potom existuje konfigurácia C stroja T' taká, že C $\vdash_{T'}^{*}$ D $\wedge$ B je pridružená s C.

<u>Potom dôkaz existencie akceptačného výpočtu je nasledovný:</u>

<u>Matematickou indukciou</u> na počet konfigurácií vo výpočte stroja T ukážeme, že existuje výpočet z počiatočnej konfigurácie stroja T' do konfigurácie, ktorá je pridružená n-tej konfigurácii vo výpočte stroja w.
(a zrejme (z def. pridružených konfigurácií) stav konfigurácie, ktorá je pridružená akceptačnej konfigurácii vo výpočte v stroji T, je akceptačný; čo sme chceli ukázať – slovo akceptuje aj stroj T')

Báza indukcie:
Pre prvú konfiguráciu tvrdenie platí na základe tvdrenia 1.

Indukčný krok:
IP: Predpokladajme, že sa pre n-tú konfiguráciu tvrdenie platí.
Na základe IP a tvrdenia 2 však platí i pre n+1 konfiguráciu.

<u>Dôkaz tvrdenia 1</u>

Stačí použiť v uvedenom poradí nasledovné časti δ' funkcie:
1 krát P1
|w| -1 krát P2
dostatočne veľa krát 1. časť P3 – neskôr si povieme koľko krát minimálne
1 krát 2. časť P3
P4 toľko krát koľko sa dá
1 krát P5
dostatočne veľa krát 1. časť P6 – neskôr si povieme koľko krát minimálne
1 krát druhú časť P6
P7 toľko krát koľko sa dá
1 krát P8

Teraz sme v konfigurácii, ktorá je podľa definície prislúcha začiatočnej konfigurácii stroja T. (ako pri zdravých konfiguráciách)

Dostatočne veľa krát znamená nasledovné: pôvodný stroj T využíval na páskach niekoľko pozícií oboma smermi od začiatočnej pozície hláv. Aby sme mohli i našim strojom chodiť po pozíciám, ktoré im zodpovedajú, musíme si ich vytvoriť ako poschodové. A to dostatočne veľa, čo je určite napr. maximálna používaná dĺžka niektorej z pások pôvodného stroja a ešte zvýšená napr. o 10. (k vôli istote krajných prípadov ☺)

Dôkaz tvrdenia 2

Majme teda A, B, C, ktoré sú podľa definície nasledovné:

$A = (q, \alpha_1, i_1, \ldots, \alpha_k, i_k) \vdash_T (p, \beta_1, j_1, \ldots, \beta_k, j_k) = B$
$C = (q[q, j_1, ..., j_k, a_1, ..., a_k, 1], h_{1,1}\ b_{1,1} \ldots h_{1,n}\ b_{1,n}\ h_{2,1}\ b_{2,1} \ldots h_{k,n}\ b_{k,n}, j)$

kde krok z A do B bol realizovaný na základe nejakej časti δ funkcie:
$\delta(p, a_1, \ldots, a_k) \ni (p_0, c_1, s_1, \ldots, c_k, s_k)$
a teda δ' odsahuje nasledovnú časť:

$\delta'(q[p, i_1, \ldots, i_k, a_1, \ldots, a_k, k+1], \alpha) =$ *** =
$\{(q[p_0, i_1, \ldots, i_k, c_1, \ldots, c_k, s_1, \ldots, s_k, 1], \alpha, 0) \mid \delta(p, a_1, \ldots, a_k) \ni (p_0, c_1, s_1, \ldots, c_k, s_k)\}$
$\forall p \in K,\ \forall a_1, \ldots, a_k \in \Gamma, \forall i_1, \ldots, i_k \in \{-1, 1\}, \forall \alpha \in \Gamma' - \{B\}$

Teda stroj T' je v konfigurácii C (typ 1) a ideme nájsť výpočet z konfigurácie C ku konfigurácii D.

Konfigurácia C nám v stave hovorí, že ideme hľadať 1. symbol, 1. značku. Kedže vieme, ktorým smerom je značka pre každú pásku, používaním pravidla P9 sa k nej stále približujeme. Keď už hlava ukazuje na značku, tak použitím P10 si v stave zapamätáme na čo značka ukazuje (v konfigurácii C ide o a_j pre j-tu pásku) a ideme hľadať ďalšiu značku a symbol, na ktorý ukazuje. Takto si zapamätáme všetky symboly, na ktoré ukazujú značky. Dostávame sa do konfigurácie, kedy by sme mali ísť hľadať k+1 značku. Takto však δ' nie je definovaná. Namiesto toho prechádza do konfigurácie typu 2 použitím ***.

V týchto konfiguráciách si v stave pamätáme symboly, na ktoré sa majú prepísať symboly, na ktoré ukazujú značky a ďalej smery, ktorými treba posunúť značky a samozrejme číslo „pásky“, ktoré práve nahrádzame.

Nahradenie prebieha znova hľadaním značky používaním P12, nahradením symbolu a zrušením značky (P13), pričom sa pohybujeme smerom, kde treba umietniť značku a súčastne prechádzame do stavu, v ktorom zapíšeme značku (P14) a znova prejdeme do stavu, kedy sa ide hľadať ďalšia značka.

Keď sme už nahradili a posunuly všetko, t.j. chceme hľadať k+1 značku, tak môžeme a musíme použiť P15, ktorá stroj dostáva do stavu, kedy sú všetky symboly pod pôvodnými značkami nahradené novými a značku posunuté, pričom tento proces prebehol na základe vlastností pôvodnej δ funkcie, z ktorých sme získali potrebnú *** časť prechodovej funkcie stroja T'.

Táto konfigurácia je zrejme pridružená konfigurácii B, čo sme chceli.

$L(T') \subset L(T)$

teda máme ukázať, že T' neakceptuje nič viac ako T.

Majme akceptačný výpočet stroja T' pre neprázdny vstup (prázdny sme už vyriešili).

Kedže akceptačné konfigurácie sú typu 1, tak aj náš výpočet musel prejsť takouto konfiguráciou. Do tejto konfigurácie sa mohol dostať použitím niektorej z P8, P9, P15.

Prípad P8: Teda konfigurácia q_0 v pôvodnom je akceptačná. T.j. pôvodný automat akceptuje všetky slová, teda i toto.

Prípad P9: Tento prípad nastáva, iba ak predošlá konfigurácia bola akceptačná, lebo P9 mení iba tú časť stavu, od ktorej nezáleží, čo ho automat akceptuje. Teda stačí sa baviť o predchádzajúcej konfigurácii, a teda o prechode do akceptačnej konfigurácie pomocou P8 alebo P15.

Prípad P15: Kedze sa použilo P15, musel výpočet byť v konfigurácii typu 3, do ktorej sa dostal použitím P14. Avšak do konfigurácie typu 3 sa dá dostať iba použítím P13. Do konfigurácie, v ktorej sa dá použiť P13 sa dostávame použitím P12, resp. P11. A do konfigurácie, v ktorej sa dá použiť P12 sa dostávame iba použitím P11 resp. P12. Teda v konečnom dôsledku sme museli použiť P11, čím dostávame, že v T existuje nejaká časť prechodovej funkcie. Do konfigurácie, v ktorej sa dá použiť P11 sa dá dostať použitím P10 a do konfigurácie, v ktorej sa dá použiť P10 sa dá dostať iba z P9 a P15, P8. Tu sa cyklíme...

Teda výpočet má tvar A $\vdash_{T'}$ B_1 $\vdash_{T'}^{*}$ B_2 $\vdash_{T'}^{*}$ B_3 ... $\vdash_{T'}^{*}$ B_n, kde A je počiatočná konfigurácia, B_i sú konfigurácie typu 1 také, že predchádzajúca konfigurácia nebola takého typu.

Ukážeme, že ku každej takejto konfigurácii B_i existuje pridružená v stroji T, pričom pre B_1 je pridružená počiatočnej konfigutácii, a pre B_i, B_{i+1} platí, že k nim pridružené sú v relácii krok výpočtu. Teda indukciou potom platí, že ku výpočtu existuje výpočet v T, že pre i-ta konfigurácia výpočtu v T je pridružená B_i. A teda existuje akceptačný výpočet ku každému slovu, ktoré akceptuje stroj T'.

Prvá konfigurácia takéhoto typu je: (ukázané pri zdravých konfiguráciách)
(q[q_0, 1, ..., 1, B, ..., B, 1],
((-, B, ..., -, B) ... (-, B, ..., -, B) „left"+1 krát (-, B, ..., -, B)
(+, w_1, +, B, ..., +, B) (-, w_2, -, B, ..., -, B)... (-, $w_{|w|}$, -, B, ..., -, B)
(-, B, ..., -, B) ... (-, B, ..., -, B)), „left"+2) „right"+1 krát (-, B, ..., -, B)
, čo je zrejme pridružená k počiatočnej konfigurácii stroja T.

Ešte ukážme druhú časť tvrdenia:
z neakceptačnej konfigurácie B_i sa do konfigurácie B_{i+1} dá dostať iba načítaním znakov pod značkami, prepísaním týchto znakov a posunutím značiek. Teda pridružená konfigurácia konfigurácii B_{i+1} je taká, čo má vymenené symboly pod pôvodnými pozíciami hláv a posunuté hlavy. Do takejto konfigurácie sa však v pôvodnom automate vieme dostať z konfigurácie pridruženej konfigurácii B_i, ak existuje také pravidlo, avšak také existuje, lebo sme urobili krok v T' použitím P11. ☺