Zadanie:

Podrobne dokážte, že problém zastavenia pre LBA je rozhodnuteľný.

Princíp riešenia:

Vieme, že pre LBA môžu nastať prípady:
- akceptuje a zastaví,
- neakceptuje a zastaví,
- nezastaví (zacyklí sa).

LBA sa zacyklí $\Leftrightarrow$ ak $\exists$ konfigurácia, taká, že sa vo výpočte bude vyskytovať viac ako 1 krát.
Teda, rozhodnuteľnosť zastavenia dokážem pomocou univerzálneho turigovho stroja, ktorý zistí, či sa daný LBA zacyklí (t.j., či sa opakuje konfigurácia).

Riešenie:

Počet rôznych konfigurácií vo výpočte môžme zhora presne ohraničiť vzťahom medzi počtom stavov, počtom znakov pracovnej abecedy a dĺžkou vstupného slova.

Nech je daný LBA $A = \{K, \Sigma, \Gamma, \delta, q_0, F\}$, kde
$| K | = k$,
$| \Gamma | = l$ (obsahuje špeciálne znaky ¢ a $),
a slovo w; $| w | = n$.

Označím b-tu konfiguráciu vo výpočte

$$\{q_a, ¢\ w\ \$, x\}_b,$$

pričom $0 \leq x \leq | w | + 1$.

Keďže konfigurácia je usporiadaná trojica troch množín, môžem povedať, že počet rôznych trojíc je rovný súčinu mohutností týchto množín.

Pre A je teda počet rôznych možných konfigurácií rovný

$$k * (l-2)^n * (n+1);$$

kde l-2 je počet znakov pracovnej abecedy bez špeciálnych symbolov ¢ a $.
Ak počet konfigurácií vo výpočte prekročí konštantu $k * (l-2)^n * (n+1)$ (t.j. $\exists$ b také, že $0 \leq b \geq (k * (l-2)^n * (n+1))$, vo výpočte nastane situácia, keď $\{q_a, ¢\ w\ \$, x\}_i = \{q_a, ¢\ w\ \$, x\}_j$, pričom $i \neq j$.
Kedže sú konfigurácie totožné, sú totožné aj nasledujúce kroky výpočtu tohto automatu.
Automat sa zacyklí.

To, či automat zastaví viem zistiť pomocou univerzálneho turingovho stroja (TSA), ktorý bude simulovať LBA.

Náčrt konšturkcie TSA:

$A=(K,\Sigma,\Gamma,\delta,q_0,F)$

$\Sigma \subseteq \Gamma$
Γ- obsahuje špeciálny symbol B
δ: $K x \Gamma \rightarrow 2^{K x (\Gamma - \{B\}) x \{-1,0,1\}}$

Konfigurácia TSA:

$$K x (\Gamma - \{B\})^* \upharpoonright (\Gamma - \{B\})^*$$

Krok výpočtu TSA:

-relácia $\vdash$ na konfigurácii definovaná:

$$(q, ¢ a_1 \ldots a_n \$, k) \vdash (p, ¢, a_1 \ldots b_n \$, l) \Leftrightarrow$$
$$(p, b_k, l-k \in \delta(q, a_k),\ a_i = b_i\ \forall i \neq k$$

Práca TSA na vstupe <LBA,w>:

1. TSA zapíše na druhú stopu kód stavu q_0
2. nájde v kóde LBA 5-ticu s príslušným stavom a symbolom
3. zrealizuje zmenu dát na páske LBA
4. ak TSA dosiahne akceptačný stav LBA, tak akceptuje

Popis δ-funkcie TSA:

1. TSA zo vstupu zistí potrebné konštanty (počet stavov, počet znakov pracovnej abecedy a dĺžky vstupného slova LBA) na

výpočet ohraničenia počtu konfigurácií. Ohraničenie zapíše na koniec pásky vo forme 1 (na koniec pripíše 0).
2. TSA bude simulovať δ-funkciu LBA.
3. Po každom kroku TSA prepíše poslednú 1 na 0.
4. Môžu nastať 2 prípady:
   - na konci pásky už nie je žiadna 1, nasledujú už len 0. Vtedy TSA zastaví bez akceptovania (LBA sa zacyklil).
   - LBA zastavil na vstupnom slove skôr, ako TS prepísal všetky 1 na konci pásky. TSA zastaví a akceptuje.

Záver:

Keďže mohli nastať 2 prípady, a to keď TSA zastavil a neakceptoval alebo zastavil a akceptoval, vieme rozhodnúť, či sa LBA zastaví.