9. PCP. Algoritmikus kérdések

1. Igazolja, hogy rekurzív a PCP-nek az a változata, amikor csak egyetlen szópár adott ($k = 1$)!

   Megoldás: Minden jó indexsorozat valahány 1-ből áll, azaz $s_1^k = t_1^k$ valamilyen $k \geq 1$ számra. Mivel a két oldal hossza azonos, ezért $|s_1| = |t_1|$ és akkor $k = 1$ is megoldás kell legyen, azaz $s_1 = t_1$ is teljesül. Ha pedig $s_1 \neq t_1$, akkor nincs megoldás. Az pedig, hogy s_1 is t_1 megegyezik-e véges lépésben eédönthető Turing-géppel, tehát ez a változat valóban rekurzív.

2. Rekurzív-e a PCP-nek az a változata, amikor minden szópárra $|s_i| = |t_i|$ teljesül?

   Megoldás: Igen. Mert ilyenkor a megoldhatóság szükséges és elégséges feltétele az, hogy legyen olyan szópár, amire $s_i = t_i$.

3. Rekurzív-e a PCP-nek az a változata, amikor minden szópár legfeljebb egyszer használható?

   Megoldás: Igen, hiszen ilyenkor csak véges sok lehetőség van (n szópár esetén kevesebb, mint $(n + 1)^n$), ezeket sorban ki tudjuk próbálni egy Turing-géppel, hogy jó megoldást adnak-e. Ez egy véges eljárás.

4. Legyen $((s_1,t_1), \ldots, (s_n,t_n); m) \in L$ az olyan esetekben, amikor az (s_i,t_i) párok által meghatározott Post megfeleltetési problémának van legfeljebb m hosszú indexsorozatból álló megoldása. Mutassa meg, hogy az L nyelv rekurzív!

   Megoldás: Az előzőhöz hasonlóan, a legfeljebb m hosszú indexsorozatok száma véges, (kevesebb, mint $(n + 1)^m$), ezek mindegyikét végig lehet próbálni.

5. Algoritmikusan eldönthető-e a következő feladat:

   (a) Adott G környezetfüggetlen nyelvtan és adott R reguláris nyelvtan esetén kérdés, hogy $L(G) = L(R)$ igaz-e.

   (b) Adott G környezetfüggetlen nyelvtan és adott R reguláris nyelvtan esetén kérdés, hogy $L(G)$ tartalmazza-e az $L(R)$ nyelvet.

   (c) Az adott G környezetfüggetlen nyelvtan generál-e 10-nél kevesebb karakterből álló szót.

   (d) Adott két környezetfüggetlen nyelvtan G_1 és G_2 a Σ felett. Igaz-e, hogy $L(G_1) \cup L(G_2) = \Sigma^*$?

   Megoldás:

   (a) Nem. Indirekt bizonyítás: tegyük fel, hogy van rá algoritmus, azaz mindig megálló M TG. Megmutatjuk, hogy akkor olyan M' is van, ami egy adott G CF nyelvtanról eldönti, hogy $L(G) = \Sigma^*$ teljesül-e, ami a tanultak szerint ellentmondás.

   M' egy adott G bemenet esetén futtassa az M gépet a (G,R) páron, ahol R egy reguláris nyelvtan, ami a Σ^* nyelvet generálja, és akkor fogadjon el, ha M elfogad. Így M' pontosan akkor fogadja el a G nyelvtant, ha $L(G) = L(R) = \Sigma^*$, ami csak akkor lehet, ha $L(G) = \Sigma^*$.

   (b) Nem. Az)aÖ-ban megadott konstrukció most is működik, hiszen $L(G) \supseteq L(R) = \Sigma^*$ csak akkor lehet, ha $L(G) = \Sigma^*$.

   (c) Igen. Ehhez csak ellenőrizni kell az összes 10-nél rövidebb szót. Egy szóra annak eldöntése, hogy benne van-e egy CF nyelvtan által generált nyelvben történhet pl. úgy, hogy Chomsky-normálformájúvá alakítjuk és utána alkalmazzuk a CYK algoritmust. (Vagy az elágazás és korlátozás módszerével a levezetésre kipróbáljuk az összes lehetőséget, azzal a megszorítással, hogy amikor a generálásnál egy legalább 10 hosszú szimbólumsorozatot kapunk, akkor azon az ágon nem megyünk tovább.)

   (d) Nem. Indirekt bizonyítás: tegyük fel, hogy van rá algoritmus, azaz mindig megálló M TG. Megmutatjuk, hogy akkor olyan M' is van, ami egy adott G CF nyelvtanról eldönti, hogy $L(G) = \Sigma^*$ teljesül-e, ami a tanultak szerint ellentmondás.

   M' egy adott G bemenet esetén futtassa az M gépet a $G_1 = G_2 = G$ választással és akkor fogadjon el, ha M elfogad. Így M' pontosan akkor fogadja el a G nyelvtant, ha $L(G) \cup L(G) = L(G) = \Sigma^*$.