

T.C.

PAMUKKALE ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
CENG 306 BİÇİMSEL DİLLER ve OTOMATA TEORİSİ DERSİ FİNAL SINAV SORULARI

Soru 1	Soru 2	Soru 3	Soru 4	Soru 5	Soru 6	Soru 7	Toplam
15	10	15	15	15	15	15	100

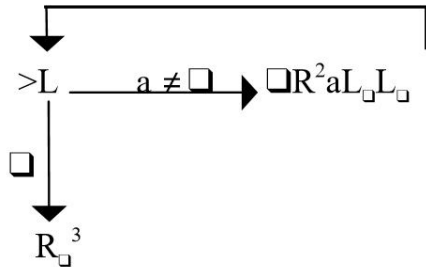
Süre: 120 dakika, Notlar: kapalı

Başarılar dilerim. Prof.Dr. Sezai TOKAT

SORU 1) $R_{\square}, L_{\square}, R_{\square}, L_{\square}, R, L$ basit TM makinelerini kullanarak girişi iki karakter sağa öteleyen Turing makinesini tasarlayınız.

Input: $\square w \square$ Output: $\square \square \square w \square$

ÇÖZÜM: Tasarımda w 'nin en sağdaki karakterinden başlayıp, bu yere boşluk yazmak, sonra iki kare sağa hareket ederek en son okunan bu karakteri bu yeni yere koymak bir çözüm olabilir. Sonra bir sonraki en soldaki karakteri bulmak için sola doğru tarama yapılarak, aynı işlem tekrarlanır, bu işlem sola doğru taramada sembol bulunamadığında sonlandırılır ve çıktıdaki gibi kafanın sembolün sağındaki ilk boşlukta durması için 3 kere boşluk bulana kadar sağa gidilir.

**SORU 2)**

$M = (K, \Sigma, \delta, s, \{h\})$, TM'de $K = \{q_0, q_1, q_2, h\}$, $\Sigma = \{a, \square, \Delta\}$, $s = q_0$, ve δ aşağıdaki tablo ile verilmiş olsun.

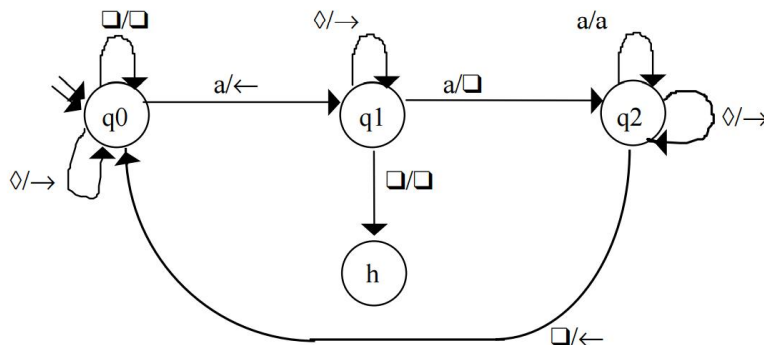
a) $n \geq 0$ olduğuna göre, $(q_0, \Delta \square a^n a)$ konfigürasyonu ile başladığında makinenin ne yaptığını anlatınız.

b) TM'nin ne zaman halt durumuna geçip ne zaman sonsuz çevrime gireceğini n değerine göre belirtiniz.

q	σ	$\delta(q, \sigma)$
q_0	a	(q_1, \leftarrow)
q_0	\square	(q_0, \square)
q_0	Δ	(q_0, \rightarrow)
q_1	a	(q_2, \square)
q_1	\square	(h, \square)
q_1	Δ	(q_1, \rightarrow)
q_2	a	(q_2, a)
q_2	\square	(q_0, \leftarrow)
q_2	Δ	(q_2, \rightarrow)

ÇÖZÜM:

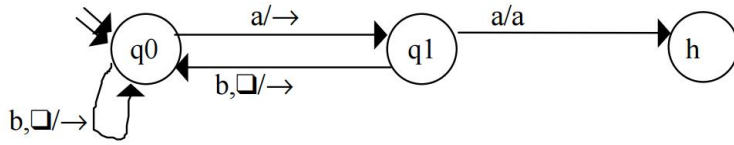
M en sağda üzerinde bulunduğu a ile başlar, birer a atlayarak a sembollerini başlangıçtan sola doğru tarayarak boşluk yapar. Eğer ilk karakter a ise ve n çift ise durur. Aksi halde sonsuz döngüye girer.



$q_0, \Delta \square a a a a$
 $q_1, \Delta \square a a a a$
 $q_2, \Delta \square a a a a$
 $q_0, \Delta \square a a a a$
 $q_1, \Delta \square a a a a$
 $q_2, \Delta \square a a a a$
 $q_0, \Delta \square a a a a$
 $q_1, \Delta \square a a a a$
 $h, \Delta \square a a a a$

SORU 3) $M = (K, \Sigma, \delta, s, \{h\})$, Turing Makinesinde $K = \{q_0, q_1, h\}$, $\Sigma = \{a, b, \square, \Delta\}$, $s = q_0$ olduğuna göre verilen bir girişi sağa doğru tarayan ve birbirini takip eden iki a bulduğunda halt durumuna geçen makineye ait geçiş fonksiyonu tablosunu veriniz.

ÇÖZÜM:



$M = (K, \Sigma, \delta, s, \{h\})$, where
 $K = \{q_0, q_1, h\}$,
 $\Sigma = \{a, b, \square, \Delta\}$,
 $s = q_0$

SORU 4) $R = \{S \rightarrow \epsilon, S \rightarrow SS, S \rightarrow [S]\}$ kurallarına sahip $G = (\{S, [,]\}, \{[,]\}, R, S)$ grameri ile tanımlanan dengeli köşeli parantez dili $L(G)$ olsun.

a) Bu $L(G)$ dilini tanıyan PDA'yı tasarlayınız.

b) Giriş katarı $[[[]]]$ için PDA'nın çalışmasını geçiş ilişkilerini ve yaprak düğümleri tablo şeklinde yazarak gösteriniz.

ÇÖZÜM:

Bu, gerçekleştirmekte zorluk çekmeyeceğiniz ve aşağıdaki PDA'yı elde edeceğiniz çok basit bir mekanik işlemdir:

a) $M = (\{p, q\}, \{[,]\}, \{S, [,]\}, \Delta, p, \{q\})$,

$\Delta = \{((p, \epsilon, \epsilon), (q, S)),$

$((q, \epsilon, S), (q, \epsilon)),$

$((q, \epsilon, S), (q, SS)),$

$((q, \epsilon, S), (q, [S])),$

$((q, [,], (q, \epsilon)),$

$((q,],], (q, \epsilon))\}$

SORU 5)

$L = \{uawb : u \text{ and } w \in \{a, b\}^* \text{ and } |u| = |w|\}$ dilini tanıyan $M = (\{q_1, q_2, q_3\}, \{a, b\}, \{x\}, q_1, \{q_3\})$ 'non-deterministik' PDA'sını tasarlayınız.

ÇÖZÜM: Buradaki kolaylık non-deterministik olarak makineyi tasarlayacak olmamızdır. Başlangıç durumu q_1 'de, a'ları ve b'leri okur ve görülen her karakter için yığına bir x iter. Böylece u'nun uzunluğunu sayar. Bir a görürse, bunun gerekli ayırıcı a olduğunu da non-deterministik olarak tahmin edebilir ve durum q_2 'ye geçebilir. Durum q_2 'de, a'ları ve b'leri okur ve görülen her karakter için yığından bir x çıkarır. Çıkarılacak bir şey yoksa, makine başarısız olur. Bir b görürse, bunun gerekli son b olduğunu da tahmin edebilir ve son durum olan durum q_3 'e geçebilir. Makine daha sonra hem girdinin hem de yığının boş olup olmadığını kabul eder.

$M = (\{q_1, q_2, q_3\}, \{a, b\}, \{x\}, q_1, \{q_3\})$,

$\Delta = \{((1, a, \epsilon), (1, x)) \text{ /* push an x on the stack for every input a}$

$((1, b, \epsilon), (1, x)) \text{ /* push an x on the stack for every input b}$

$((1, a, \epsilon), (2, \epsilon)) \text{ /* guess that this is the separator a. No stack action}$

$((2, a, x), (2, \epsilon)) \text{ /* for each input a, pop an x off the stack}$

$((2, b, x), (2, \epsilon)) \text{ /* for each input b, pop an x off the stack}$

$((2, b, \epsilon), (3, \epsilon)) \text{ /* guess that this is the final b and go to the final state } \}$

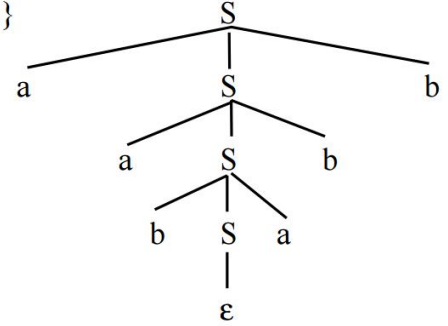
SORU 6) $L = \{w^R w'' : w \in \{a, b\}^* \text{ ve } w'' \text{ ise } w \text{ katarında (} w^R \text{ de değil!)} \text{ her bir görülen a yerine b, her bir görülen b yerine a konulan katar} \}$ olarak verilmiştir.

a) L dilini üreten CFG'yi elde ediniz.

b) $aababb \in L$ için türetme ağacını çizin. (türetme sırası sorulmuyor, sadece ağaç)

ÇÖZÜM:

$G = (\{S, a, b\}, \{a, b\}, R, S), R = \{ S \rightarrow aSb, S \rightarrow bSa, S \rightarrow \epsilon \}$



SORU 7) Bir L dilini tanımlayan düzenli gramer aşağıdaki gibidir:

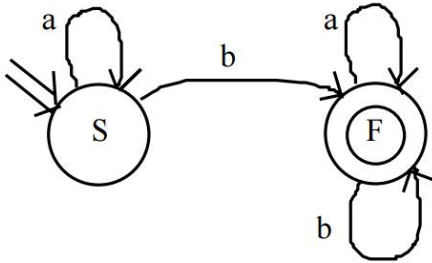
$S \rightarrow bF, S \rightarrow aS, F \rightarrow \epsilon, F \rightarrow bF, F \rightarrow aF$

a) L dilini tanıyan DSO'yu çiziniz.

b) L diline ait düzenli ifadeyi elde ediniz.

c) Bir cümle ile bu dili tanımlayınız.

(a)



(b) $(a \cup b)^*ba^*$ OR $a^*b(a \cup b)^*$

(c) $L = \{ w \in \{a, b\}^* : \text{there is at least one } b \}$