

OTOMATA TEORİSİ

Bölüm 6

TURING MAKİNELERİ

6.1. TURING MAKİNELERİ

- Turing makineleri, genel amaçlı bilgisayarlar için matematiksel bir modeldir.
- Herhangi bir makineyle yapılabilecek işlemleri Turing makinesi ile modellemek mümkündür.
- Başka bir deyişle, bir işlemin herhangi bir makineyle yapılabilmesi için, bu işlemin Turing makineleriyle yapılabilir (modellenebilir) bir işlem olması gerekir.
- Bu özellikleri dolayısıyla Turing makinesi modeli, genel amaçlı sayısal bilgisayarlar için önemli bir modeldir.

6.1. TURING MAKİNELERİ

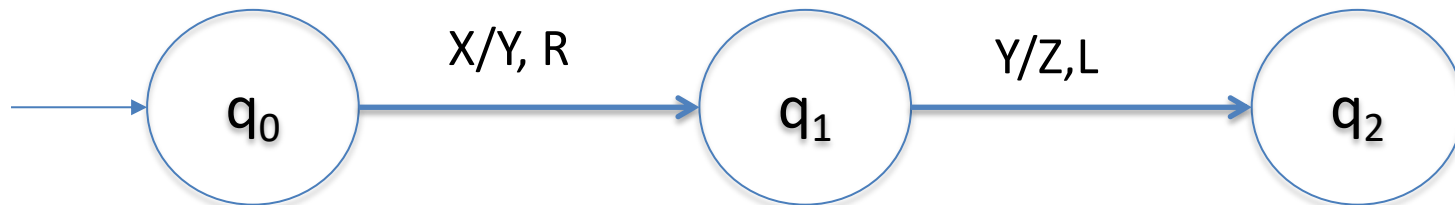
- Turing makinelerinin kullanım alanları aşağıdaki gibi sınıflandırılabilir:
- **1.Dil Tanıyıcı:** Turing makineleri kısıtlamasız (unrestricted) ya da özyineli sayılabilir (r.e. : recursively enumerable) dilleri tanımak için kullanılabilir. Buna göre Turing makinesi, verilen bir tümcenin dilin tümcesi olup olmadığını bulabilir.
- **2. Hesaplayıcı:** Turing makineleri, kısmi özyineli tamsayı fonksiyonların (partially recursive integer functions) hesaplanmasında kullanılabilir.
- **3. Dil üreticisi :** Turing makineleri ile r.e. dillerin tümcelerini ardarda üretmek mümkündür. Eğer biçimsel bir dil Turing makinesi tarafından üretilebiliyorsa bu dil r.e. bir dildir. Diğer taraftan her r.e. dil için, dilin tüm tümcelerini ardarda üreten bir Turing makinesi vardır.
- Bu derste Turing makinelerinin dil tanıyıcı ve hesaplayıcı olarak kullanımı üzerinde durulacak ve **r.e.** dilleri tanıyan ve basit hesaplamaları gerçekleştiren Turing makinesi örnekleri verilecektir.

6.1.1. Turing Makinelerinin Temel Modeli

- Biçimsel olarak, Turing makinesinin temel modeli bir yedili olarak tanımlanır:
 - $M = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$
 - Q : Sonlu sayıda durum içeren Durumlar Kümesi
 - Σ : Sonlu sayıda giriş simgesinden oluşan Giriş Alfabeti
 - Γ : Sonlu sayıda simge içeren Şerit Alfabeti. Şerit alfabeti, giriş alfabetinin tüm simgelerini içeren bir kümedir: $\Gamma \supseteq \Sigma$
 - q_0 : Başlangıç durumu ($q_0 \in Q$). Başlangıç durumu durumlar kümesinin bir elemanı olduğuna göre Q boş olmayan bir kümedir.
 - B : Şerit alfabetindeki simgelerden blank olarak adlandırılan özel bir simge. B şerit alfabetinde yer alan ancak giriş alfabetinde yer almayan bir simgedir: $B \in \Gamma, B \notin \Sigma$
 - F : Uç durumlar kümesi. Durumlar kümesinin bir altkümesidir: $F \subseteq Q$
 - δ : Geçiş ya da hareket işlevi (transition or move function)
- Turing makinelerinin temel modeli deterministik bir modeldir. Bu modelde, hareket işlevi
- $[Q \times \Gamma]^*$ 'dan $[Q \times \Gamma \times \{L, R\}]^*$ 'ye bir eşleşme oluşturur.

6.1.1. Turing Makinelerinin Temel Modeli

- Tanımda yer alan L ve R simgeleri, hareketin sonunda okuma kafasının bir sağdaki (**R**) ya da bir soldaki (**L**) hücreye geçeceğini gösterir. Okuma kafasının hareketiyle ilgili seçenekler arasına “sağa/sola hareket etmeyip aynı hücre üzerinde kalma (**S**)” seçeneğini eklemek ve iki elemanlı $\{ L, R \}$ kümesi yerine üç seçenekli $\{ L, R, S \}$ kümesini kullanmak da mümkündür. Ancak bu kitapta çoğunlukla $\{ L, R \}$ kümesi kullanılacak ve Turing makineleri oluşturulurken, okuma kafasının her harekette bir sağ ya da bir sol hücreye geçmesi sağlanacaktır. Buna göre Turing makinesinin hareketleri aşağıdaki örneklerde görüldüğü gibi tanımlanacak, istenirse hareketler bir çizenekle de gösterilecektir.
- $\delta(q_0, X) = (q_1, Y, R)$ $q_0, q_1, q_2 \in Q$
- $\delta(q_1, Y) = (q_2, Z, L)$ $X, Y, Z \in \Gamma$



6.1.2. Soyut Makine Görünümü

Tanımlandığı biçimiyle Turing makinesi matematiksel bir modeldir. Ancak bundan önceki modellerde olduğu gibi, Turing makinesini de soyut bir makine olarak düşünmek mümkündür.

Turing makinesinin nasıl çalıştığını daha iyi anlayabilmek için, aşağıdaki bileşenlerden oluşan soyut bir makine modeli kullanılmaktadır (Çizim 6.1):

- Hücrelerden oluşan ve her hücresinde bir giriş simgesi bulunan bir mıknatıslı şerit. Hem okuma hem de yazma işlemleri yapılablen şeridin sağ ucu sonsuzdur.
- Bir sonlu denetim birimi (SDB)
- Bir okuma kafası

Turing makinesinin Çizim 6.1 'de görülen soyut makine modeli aşağıda açıklanan biçimde çalışır.

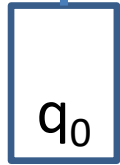
- başlangıçta şerit üzerinde giriş simgelerinden oluşan bir dizgi (w) kayıtlıdır ve okuma kafası bu dizginin ilk (en soldaki) simgesi üzerindedir.
- Makinenin her hareketinde aşağıdaki işlemler yapılır:
 - Şeritten bir simge (okuma kafasının üzerinde bulunduğu simge) okunur.
 - Okunan simgenin yerine yeni bir simge yazılır.
 - Okuma kafası sağdaki (R) ya da soldaki (L) hücreye geçer
 - Sonlu denetim birimi yeni duruma geçer.

6.1.2. Soyut Makine Görünümü

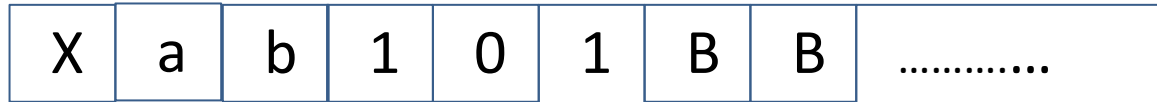


Okuma Kafası

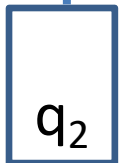
a) Başlangıç görünümü



Sonlu Denetim
Birimi



b) Birkaç hareket Sonraki Görünümü

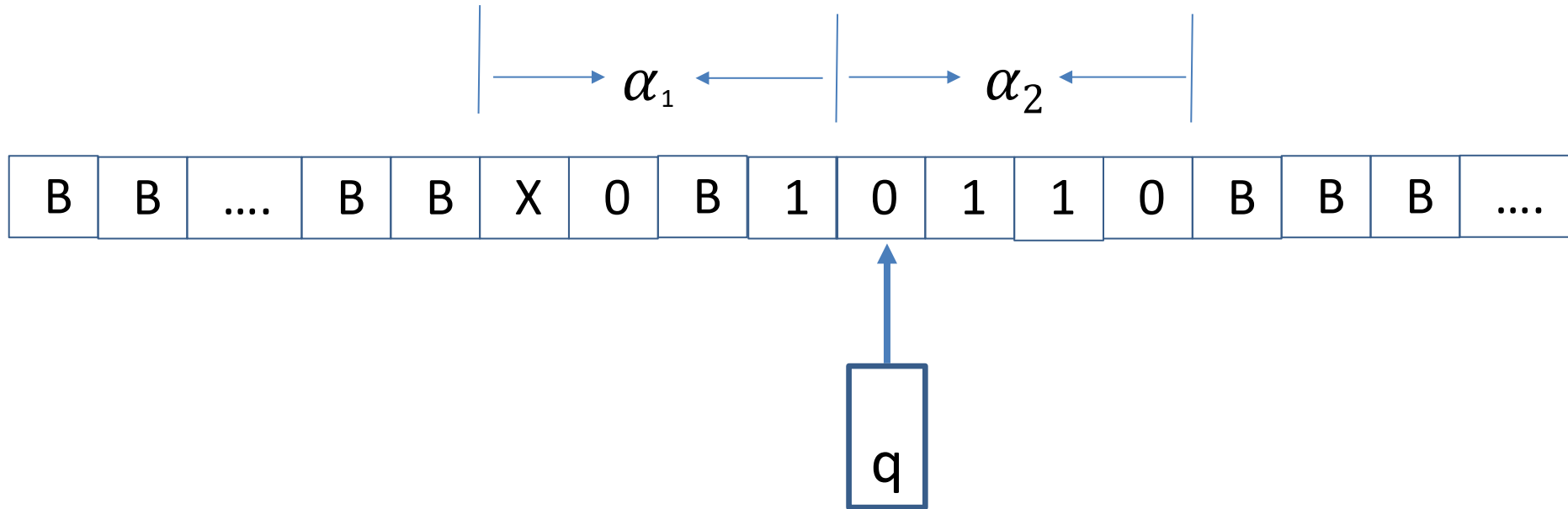


Çizim 6.1. Turing Makinesinin Soyut Makine Görünümü

6.1.3. Anlık Tanımlar (Instantaneous Descriptions)

- Turing makinesi, şerit üzerinde her iki yönde hareket ederek hem okuma hem de yazma yapabilen bir model olduğundan, makinenin sonraki davranışlarını kestirebilmek için belirli bir andaki üç bilginin bilinmesi gerekir:
- Şerit üzerinde, okuma kafasının solunda bulunan dizgi (α_1)
- Sonlu denetim biriminin durumu (q)
- Şerit üzerinde okuma kafasının sağındaki dizgi (α_2). Okuma kafasının üzerinde bulunduğu simge α_2 'ye dahildir.
- Bu üç bilginin belirli bir andaki değerlerinden oluşan üçlüye, Turing makinesinin anlık tanımı denir:
- **Anlık tanım (ID) = (α_1, q, α_2)**
 - q : makinenin durumu
 - α_1 : okuma kafasının solundaki dizgi
 - α_2 : okuma kafasının sağındaki dizgi (okuma kafası α_2 'nin en solundaki simge üzerinde bulunur)

6.1.3. Anlık Tanımlar (Instantaneous Descriptions)



- **B (blank)** simgesi boşlukları gösterdiği için, şeridin sağ ve sol tarafındaki, salt **B** simgelerinden oluşan kesimler α_1 ve α_2 'ye dahil edilmez. Buna göre α_1 'in en solundaki, α_2 'nin de en sağındaki simge B olamaz; ancak α_1 ve α_2 'nin içinde **B** simgesi bulunabilir.

6.1.4. Turing Makinesinin Tanıdığı Dil

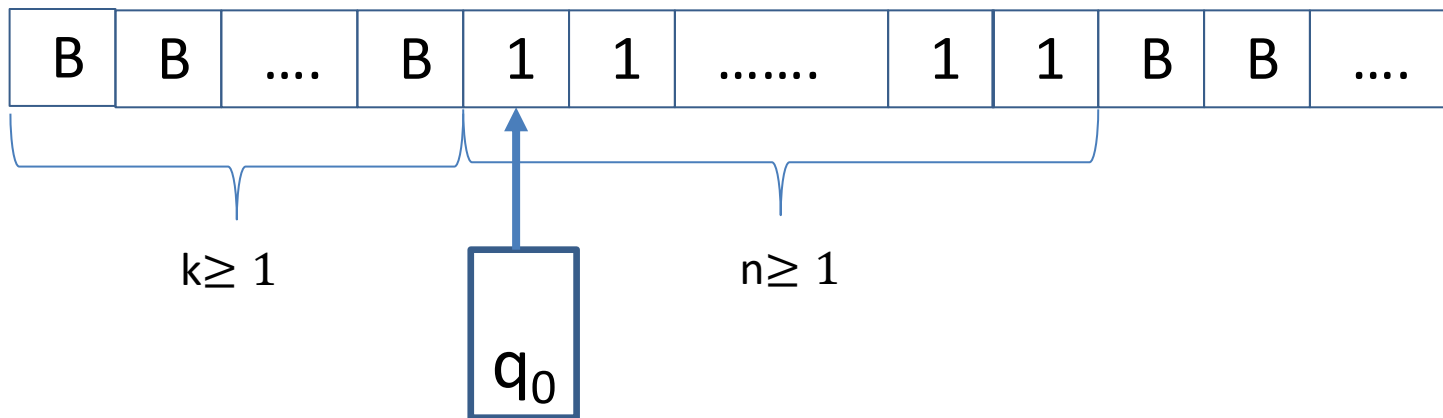
- Dil tanıyıcı olarak kullanılan bir Turing makinesinin tanıdığı dil biçimsel olarak aşağıdaki gibi tanımlanabilir:
- $T(M) = \{w \mid w \in V_T^*, (q_0, w) \vdash^* (\alpha_1, p, \alpha_2), \alpha_1, \alpha_2 \in \Gamma^*, p \in F\}$

Yukarıdaki tanıma göre, giriş simgelerinden oluşan w dizgisinin Turing makinesi tarafından tanınabilmesi için, makinenin bir uç durumda durması gerekir. Başka bir deyişle, bitiş konfigürasyonunu gösteren anlık tanımda **(α_1, p, α_2)** , önemli olan **p** 'nin bir uç durum olmasıdır. Şeridin görünümü **$(\alpha_1$ ve $\alpha_2)$** tanımada etkili değildir.

6.2. TURING MAKİNESİ ÖRNEKLERİ

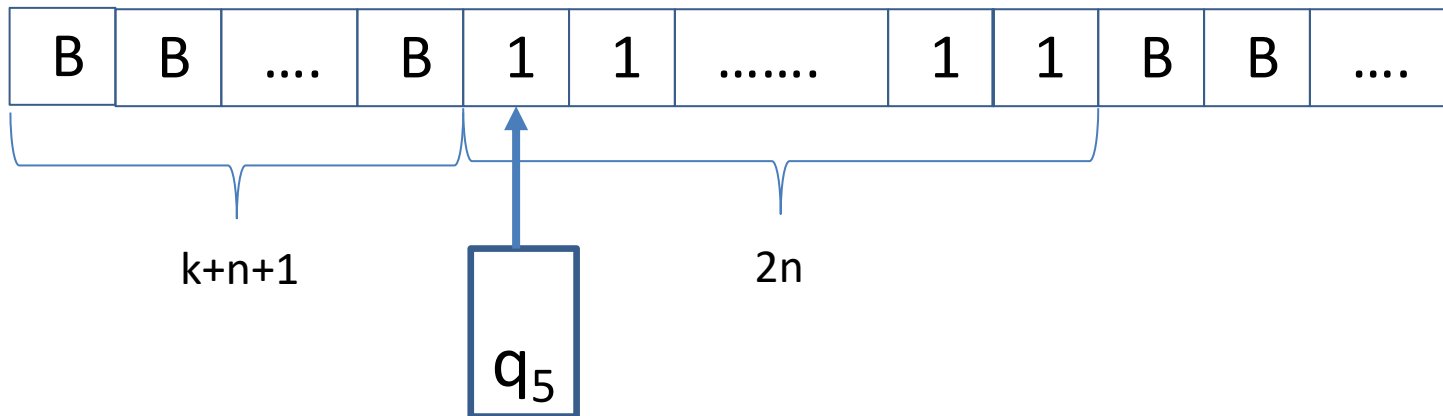
6.2.1. Örnek 6.1

- n sıfırdan büyük pozitif bir tamsayı veya sıfır olmak üzere $f(n) = 2n$ değerini hesaplayan Turing makinesini tasarlayalım. Makinede n sayısı, yanyana n tane 1 simgesiyle gösterilir.
- Başlangıç konfigürasyonu:



6.2.1. Örnek 6.1

- Hesaplama yöntemi:
- **1**'ler öbeğinin sağındaki ilk hücreye bir işaretleyici (örneğin **ç**) yazılır.
- **1**'ler öbeğinin en solundaki **1**, yerine **B** yazılarak silinir. Silinen **1**'e karşılık, sağdaki ilk iki **B**'nin yerine **1** yazılır.
- 2'deki işlemler, **ç**'nin solundaki **1**'ler bitinceye kadar tekrarlanır. **ç**'nin solundaki **1**'ler bittiğinde, **ç**'nin yerine **B** yazılır ve hesaplama biter.
- Bitiş konfigürasyonu:



6.2.1. Örnek 6.1

Makinenin biçimsel tanımı:

$$M_{6.1} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q: \{q_0, q_1, q_2, q_3, q_4, q_5\}$$

$$\Sigma : \{1\}$$

$$\Gamma : \{B, 1, \epsilon\}$$

q_0 : Başlangıç Durumu

$F : \Phi$ (makine bir tanıyıcı değil, bir hesaplayıcı olduğundan, uç durum anlamsızdır).

Durumların anlamları:

q_0 : Başlangıç durumu. 1'ler öbeğinin sağına işaretleyicinin $\{\epsilon\}$ konulduğu durum bu durumdur.

q_1 : Sola ilerleyip ilk B'nin bulunduğu durum.

q_2 : 1'ler öbeğinin en solundaki 1'in silindiği durum.

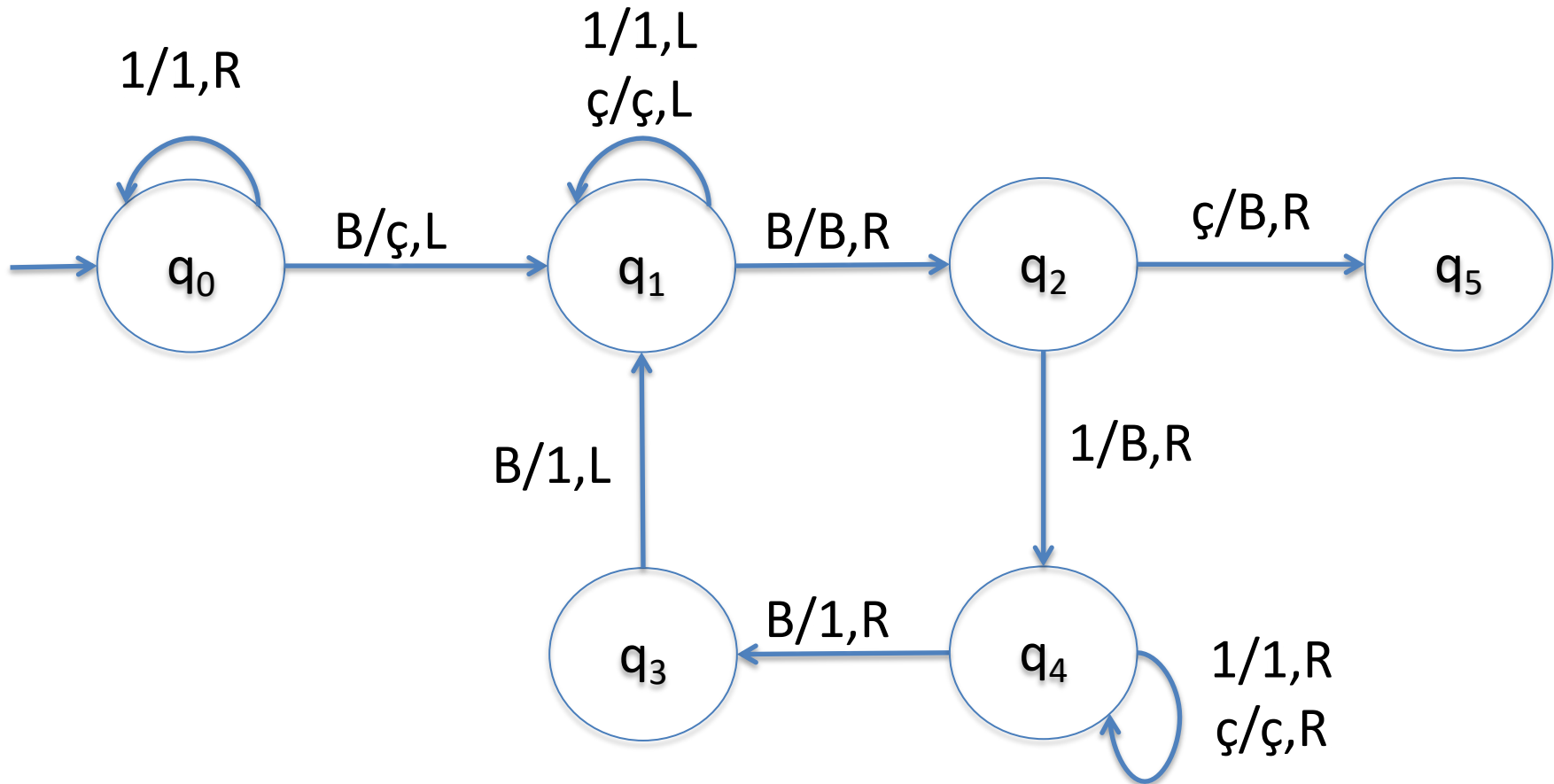
q_3 : Sağa ilerleyip ilk B'nin bulunduğu ve yerine 1 yazıldığı durum.

q_4 : İkinci B'nin yerine 1 yazıldığı durum.

q_5 : Son durum. Hesaplama tamamlandığında makine durumu gelir ve durur.

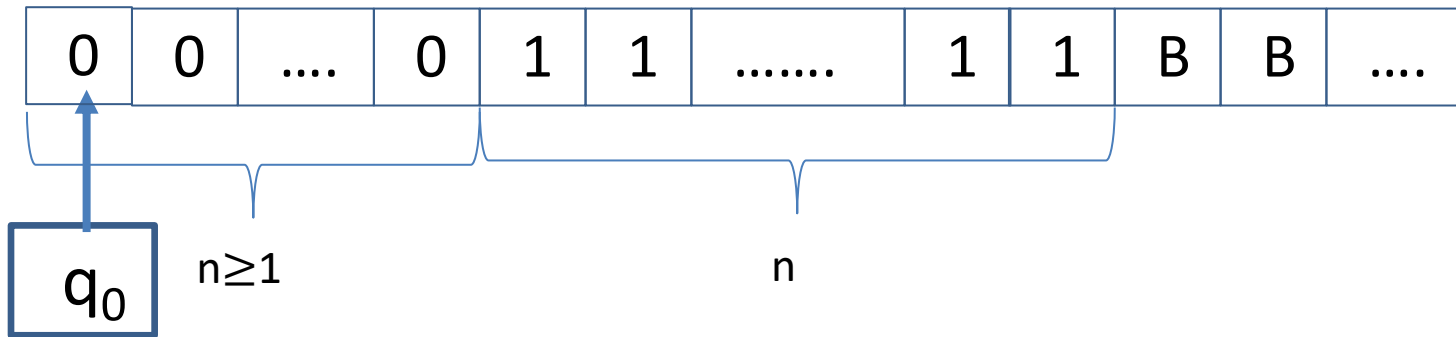
6.2.1. Örnek 6.1

Hareket Çizelgesi	1	ç	B
q_0	$(q_0, 1, R)$	-	$(q_1, ç, L)$
q_1	$(q_1, 1, L)$	$(q_1, ç, L)$	(q_2, B, R)
q_2	(q_3, B, R)	(q_5, B, R)	-
q_3	$(q_3, 1, R)$	$(q_3, ç, R)$	$(q_0, 1, R)$
q_4	-	-	$(q_1, 1, L)$
q_5	-	-	-



6.2.2. Örnek 6.2

- $L_{6.2}$ dili aşağıdaki gibi tanımlanıyor.
- $L_{6.2} = \{0^n 1^n \mid n \geq 1\}$
- Bu dil bağlamdan-bağımsız (CF) bir dildir. Ancak Turing makineleri tarafından tanınan diller sınıfı, bağlamdan-bağımsız dilleri de içerdiği için, $L_{6.2}$ dilini tanıyan bir Turing makinesi tanımlamak mümkündür. Başlangıç konfigürasyonu:



Çalışma yöntemi:

- En soldaki **0**'ın yerine **X** yazılır.
- En soldaki **1**'in yerine **Y** yazılır. Eğer yerine **X** yazılan bir **0**'a karşılık **1** bulunamazsa, makine uç olmayan bir durumda durur ve dizgi tanınmaz.
- **1** ve **2**'deki işlemler **0**'lar bitinceye kadar tekrarlanır.
- **0**'lar bittiğinde, **1** kalmadığı denetlenir. Eğer **1** kalmamışsa makine uç durumda durur ve dizgi tanınır; **1** kalmışsa makine uç olmayan bir durumda durur ve dizgi tanınmaz.

6.2.2. Örnek 6.2

$$M_{6.2} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q: \{q_0, q_1, q_2, q_3, q_4\}$$

$$\Sigma : \{0, 1\}$$

$$\Gamma : \{B, 0, 1, X, Y\}$$

q_0 : Başlangıç Durumu

$$F : \{q_4\}$$

Durumların Anlamları:

q_0 : en soldaki 0 ın X yapıldığı durum; aynı zamanda başlangıç durumu.

q_1 : sağa ilerleyip en soldaki 1 in Y yapıldığı durum.

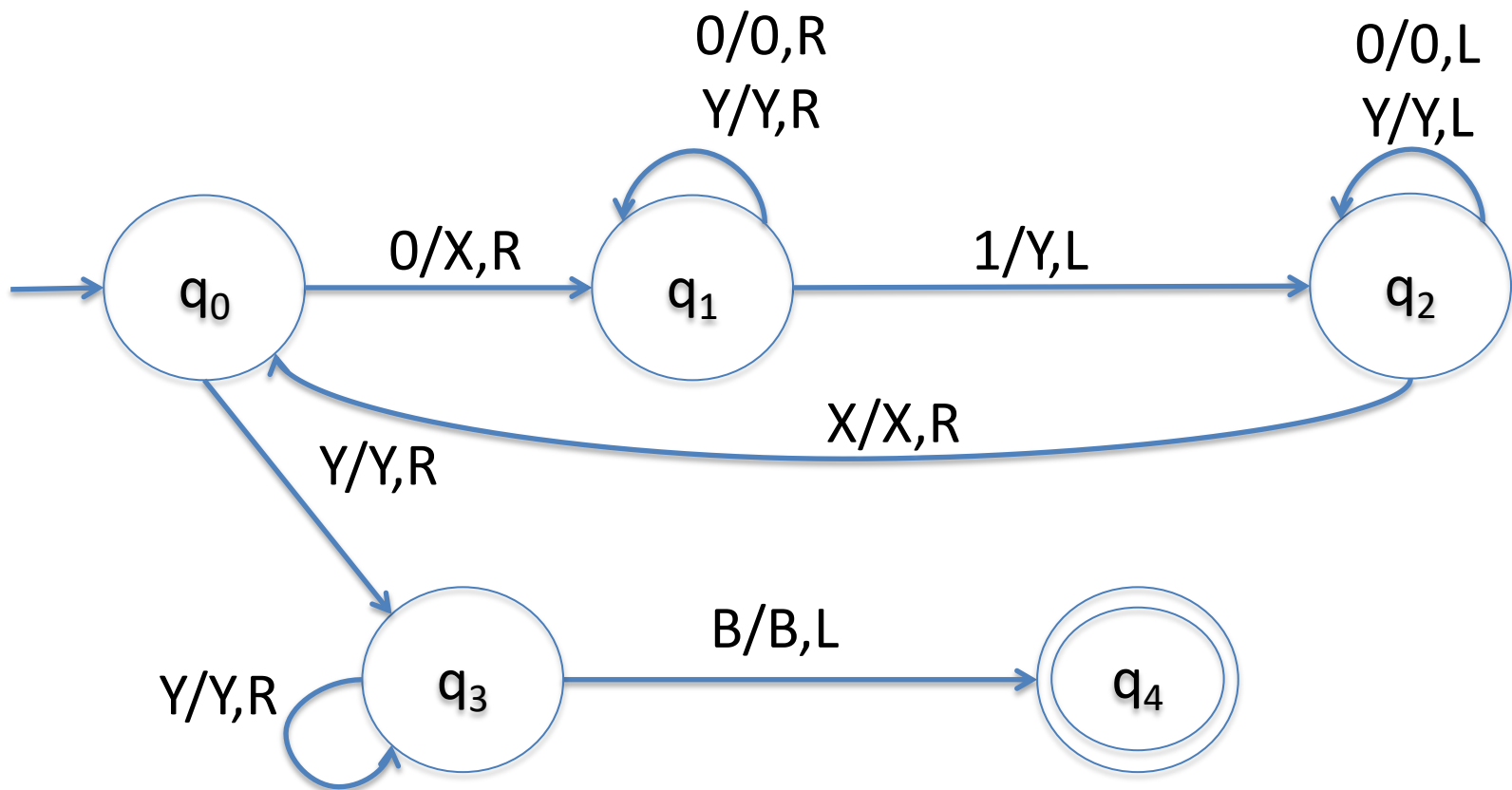
q_2 : sola ilerleyip ilk X de sağa dönülen durum.

q_3 : sağa ilerleyip başka 1 kalmadığının denetlendiği durum.

q_4 : Uç durum. Makine q_4 de durduğunda dizgi tanınır.

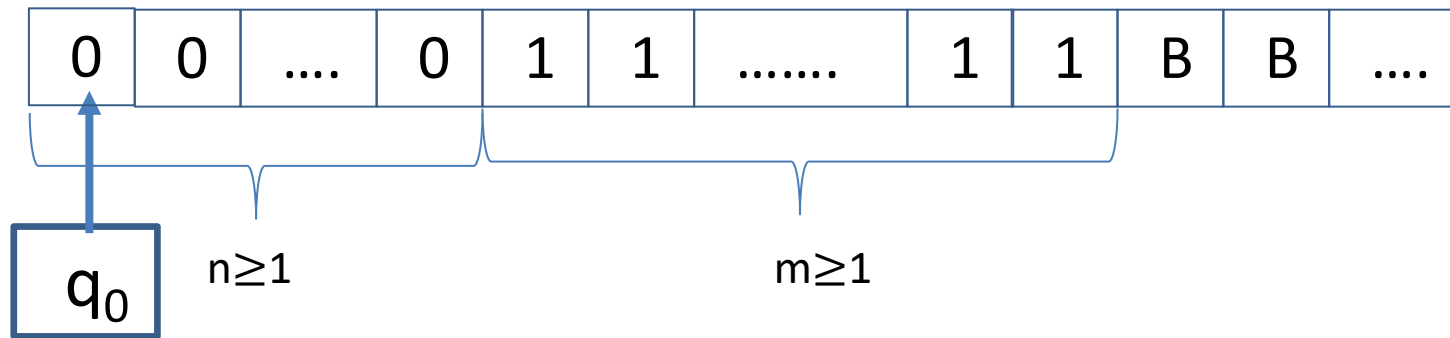
6.2.2. Örnek 6.2

Hareket Çizelgesi	0	1	X	Y	B
q_0	(q_1, X, R)	-	-	(q_3, Y, R)	-
q_1	$(q_1, 0, R)$	(q_2, Y, L)	-	(q_1, Y, R)	-
q_2	$(q_2, 0, L)$	-	(q_0, X, R)	(q_2, Y, L)	-
q_3	-	-	-	(q_3, Y, R)	(q_4, B, L)
q_4	-	-	-	-	-



6.2.3. Örnek 6.3

- n ve m sıfırdan büyük pozitif birer tamsayı olmak üzere
- $f(n, m) = n \times m$ değerini hesaplayan Turing makinesini tasarlayalım.
- **Başlangıç konfigürasyonu:**



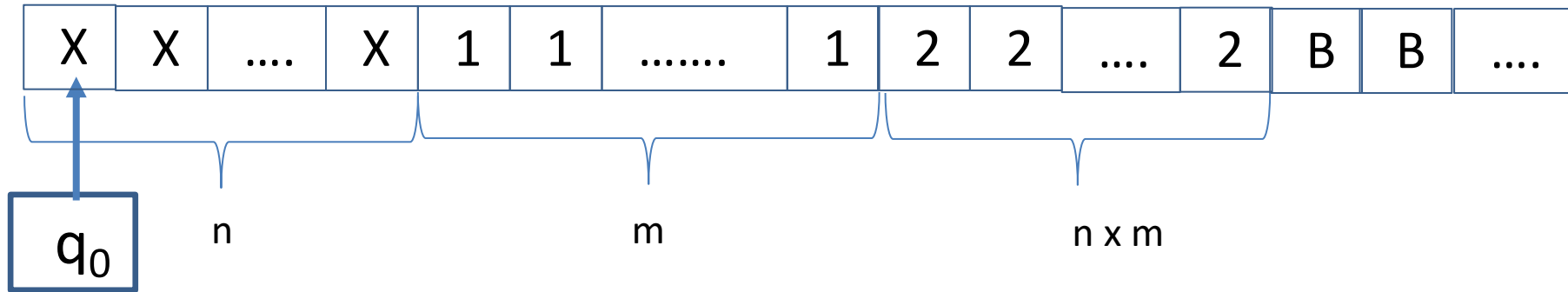
Hesaplama yöntemi:

Çarpma işlemi, m 'nin n kere toplanması ile gerçekleştirilecektir.

Bunun için, n 'yi oluşturan her 0 için, 1 'ler öbeği kopyalanacaktır.

1 'ler öbeği n kere kopyalandığında $n \times m$ değeri elde edilmiş olacaktır.

6.2.3. Örnek 6.3



6.2.3 Örnek 6.3

$$M_{6.3} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$Q: \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8\}$

$\Sigma : \{0, 1\}$

$\Gamma : \{B, 0, 1, 2, X, Y\}$

q_0 : Başlangıç Durumu

$F : \Phi$ (makine bir tanıyıcı değil, bir hesaplayıcı olduğundan, uç durum anlamsızdır).

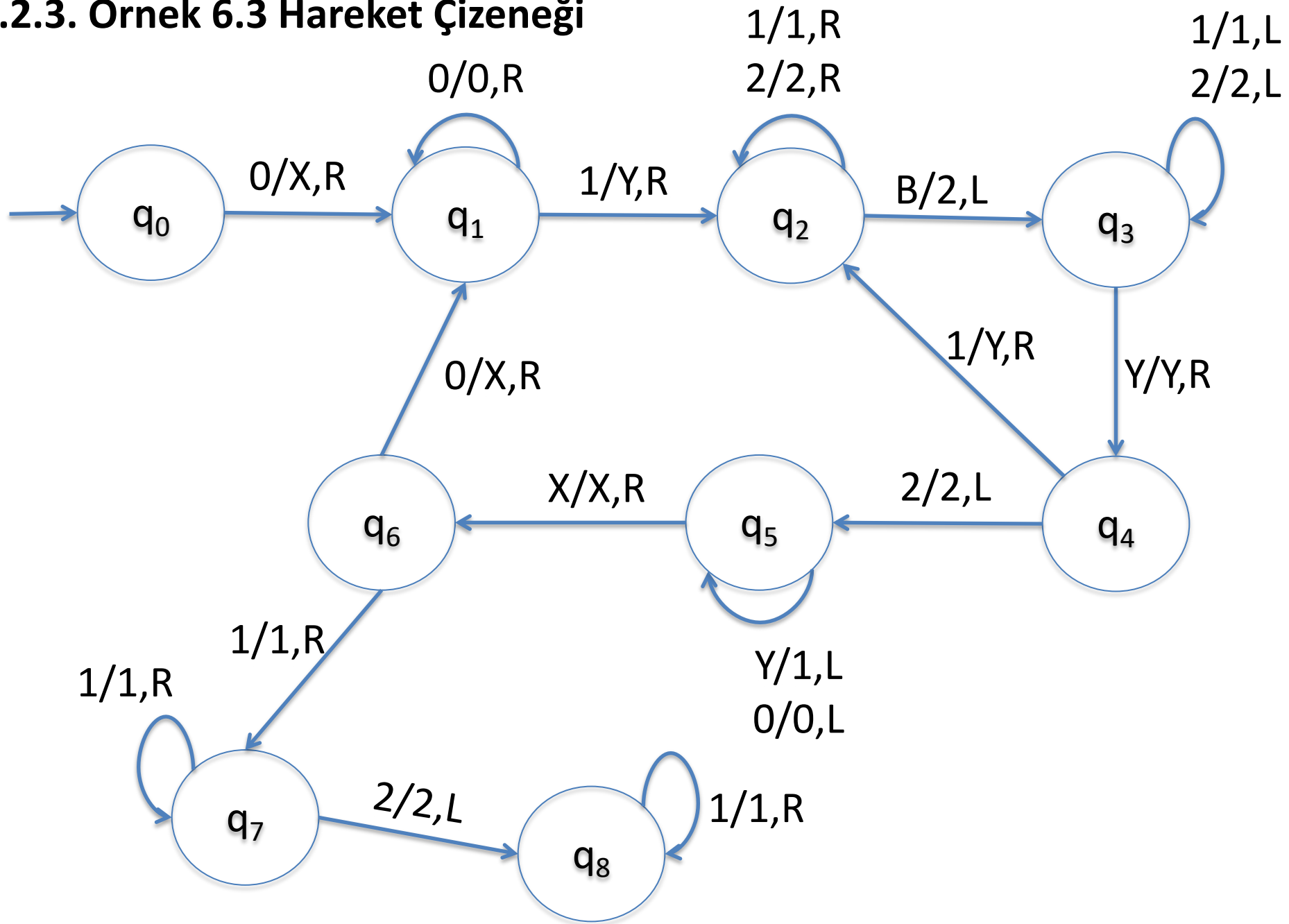
Durumların Anlamları:

- q_0 : Başlangıç durumu. Dış döngünün birinci adımını başlatır.
- q_1 : İç döngünün birinci adımını başlatır.
- q_2 : Sağa ilerleyerek ilk B'nin 2 yapıldığı durum.
- q_3 : İç döngüde geriye dönüşü sağlayan durum.
- q_4 : İç döngünün sonraki adımlarını başlatan; iç döngü bitti ise dış döngünün geriye dönüşünü başlatan durum.
- q_5 : Dış döngünün geriye dönüşünü sağlayan, bu arada 1'ler öbeğine eski görünümünü kazandıran durum.
- q_6 : Dış döngünün sonraki adımlarını başlatan durum
- q_7, q_8 : Bitiş konfigürasyonunda, okuma kafasının 2'ler öbeğinin başında olmasına sağlayan durumlar.

6.2.3. Örnek 6.3 Hareket Çizelgesi

Hareket Çizelgesi	0	1	X	Y	B
q ₀	(q ₁ ,X,R)	-	-	-	-
q ₁	(q ₁ ,0,R)	(q ₂ ,Y,R)	-	-	-
q ₂	-	(q ₂ ,1,R)	(q ₂ ,2,R)	-	(q ₃ ,2,L)
q ₃	-	(q ₃ ,1,L)	(q ₃ ,2,L)	(q ₄ ,Y,R)	-
q ₄	-	(q ₂ ,Y,R)	(q ₅ ,2,L)	-	-
q ₅	(q ₅ ,0,L)	-	-	(q ₅ ,1,L)	-
q ₆	(q ₁ ,X,R)	(q ₇ ,1,R)	-	-	-
q ₇	-	(q ₇ ,1,R)	(q ₈ ,2,L)	-	-
q ₈	-	(q ₈ ,1,R)	-	-	-

6.2.3. Örnek 6.3 Hareket Çizeneği

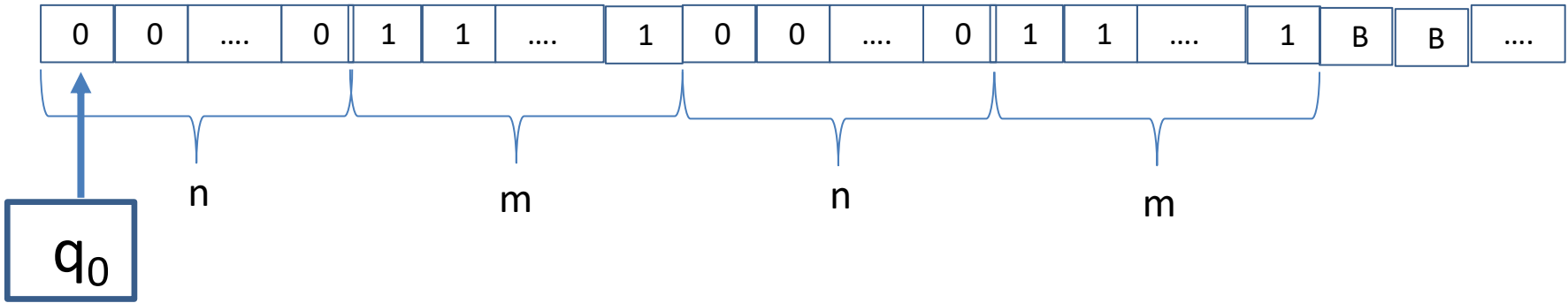


6.2.4. Örnek 6.4

- $L_{6.4}$ dili aşağıdaki gibi tanımlanıyor.

$$L_{6.4} = \{0^n 1^m 0^n 1^m \mid n \geq 1, m \geq 1\}$$

- Bu dil bağlamdan-bağımsız (CF) bir dil değildir. Bu dili ancak bir Turing makinesiyle tanımak mümkündür. $L_{6.4}$ dilini tanıyan bir Turing makinesi aşağıdaki gibi tanımlanabilir.
- Başlangıç konfigürasyonu:**



Çalışma yöntemi:

- Önce **0** öbeklerindeki simgeler karşılıklı olarak yerlerine **X** ve **Z** yazılarak silinir.
- Sonra da **1** öbeklerindeki simgeler karşılıklı olarak yerlerine **Y** ve **W** yazılarak silinir.
- Sonunda hiç **0** ve **1** kalmadıysa dizgi tanınır.

6.2.4. Örnek 6.4

$$M_{6.4} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$Q: \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10}\}$

$\Sigma: \{0, 1\}$

$\Gamma: \{B, 0, 1, X, Y, Z, W\}$

q_0 : Başlangıç Durumu

$F: \{q_{10}\}$

Durumların Anlamları:

q_0 : en soldaki 0 ın X yapıldığı durum; aynı zamanda başlangıç durumu.

q_1 : sağa ilerleyerek soldaki 1 ler öbeğinin bulunduğu durum.

q_2 : sola ilerleyerek ikinci 0 lar öbeğinin ilk 0'ın bulunup Z yapıldığı durum.

q_3 : X i buluncaya kadar sola ilerlenen durum.

q_4 : Soldaki ilk 0 ın X yapıldığı; 0 lar bittiğinde ise ilk 1 in Y yapıldığı durum

q_5 : sağa ilerleyerek soldaki Z ler öbeğinin bulunduğu durum

q_6 : sağa ilerleyerek ikinci 1'ler öbeğindeki ilk 1 in bulunup W yapıldığı durum

q_7 : Y i buluncaya kadar sola ilerlenen durum

q_8 : Soldaki ilk 1'in Y yapıldığı durum.

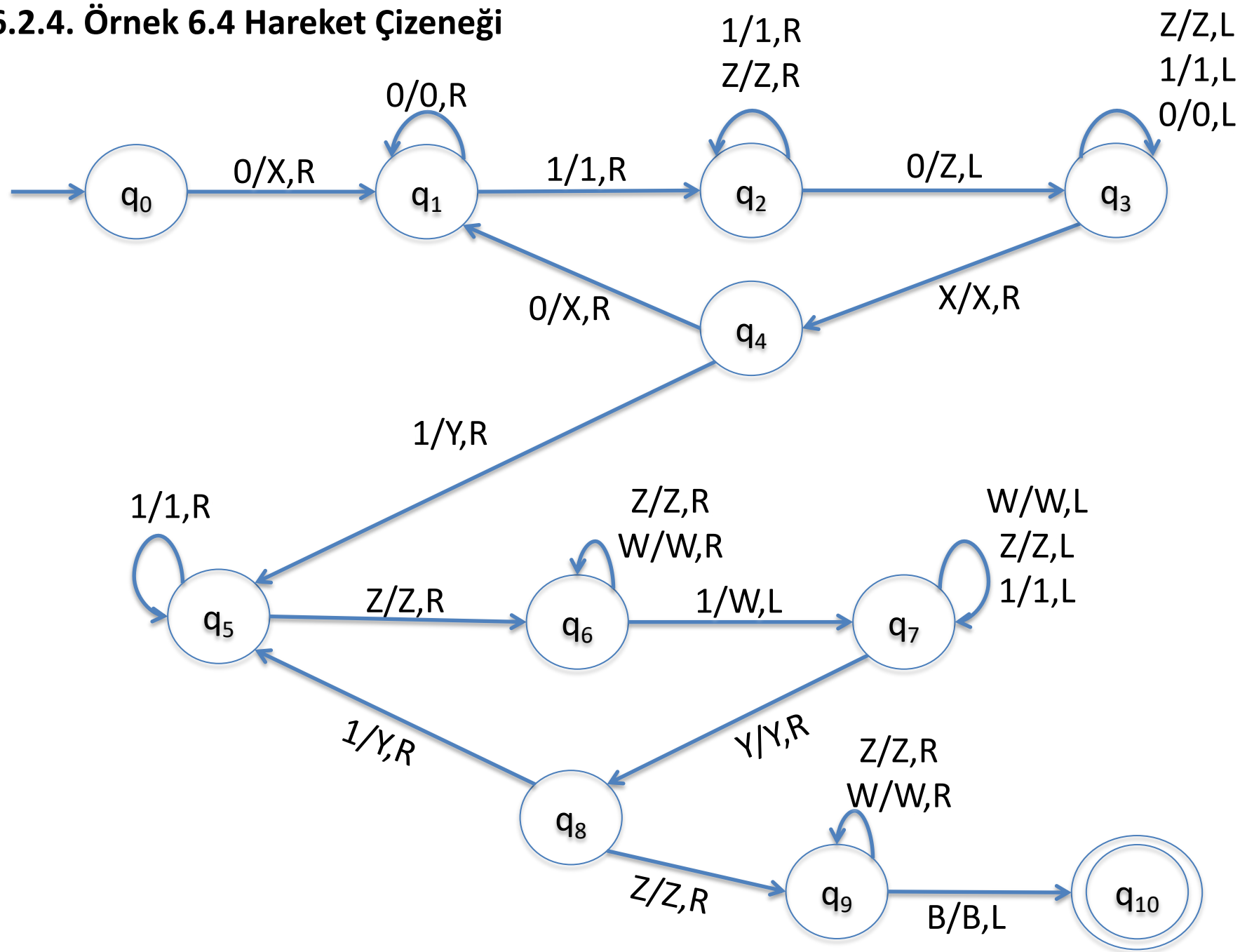
q_9 : Sağa ilerlenerek başka 0 ve 1 kalmadığının denetlendiği durum

q_{10} : Uç durum.

6.2.4. Örnek 6.4 Hareket Çizelgesi

Hareket Çizelgesi	0	1	X	Y	Z	W	B
q ₀	(q ₁ ,X,R)	-	-		-	-	
q ₁	(q ₁ ,0,R)	(q ₂ ,1,R)	-		-	-	
q ₂	(q ₃ ,Z,L)	(q ₂ ,1,R)	-		(q ₂ ,Z,R)	-	
q ₃	(q ₃ ,0,L)	(q ₃ ,1,L)	(q ₄ ,X,R)		(q ₃ ,Z,L)	-	
q ₄	(q ₁ ,X,R)	(q ₅ ,Y,R)	-		-	-	
q ₅	-	(q ₅ ,1,R)	-		(q ₆ ,Z,R)	-	
q ₆	-	(q ₇ ,W,L)	-		(q ₆ ,Z,R)	(q ₆ ,W,R)	
q ₇	-	(q ₇ ,1,L)	-	(q ₈ ,Y,R)	(q ₇ ,Z,L)	(q ₇ ,W,L)	
q ₈	-	(q ₅ ,Y,R)	-		(q ₉ ,Z,R)	-	
q ₉	-	-	-		(q ₉ ,Z,R)	(q ₉ ,W,R)	(q ₁₀ ,B,L)
q ₁₀	-	-	-		-	-	-

6.2.4. Örnek 6.4 Hareket Çizeneği



6.3. TURING MAKİNESİ MODELİNDE DEĞİŞİKLİKLER

- Turing makinelerinin temel modelinde yalnız bir ucu sonsuz, tek izli bir şerit ile tek okuma kafası kullanılmakta ve her harekette yalnız bir simge okunup-yazılabilmektedir.
- Şeridin iki yönde sonsuz olması, şerit sayısı, iz sayısı ve okuma kafası sayısı ile ilgili olarak Turing makinesi modelinde bir dizi değişiklik yapılabilir. Bu değişiklikler modelin kullanımında esneklik ve kolaylıklar sağlar. Ancak modelin gücünde hiçbir değişiklik yapılmaz.
- Başka bir deyişle, hangi değişik model kullanılırsa kullanılsın, yapılan hesaplamayı temel model bir Turing makinesi ile yapmak ya da tanınan dili temel model bir Turing makinesi ile tanımak mümkündür.

6.3. TURING MAKİNESİ MODELİNDE DEĞİŞİKLİKLER

- Turing makinesi modelinde yapılabilecek değişikliklerden başlıcaları aşağıdaki gibi sıralanabilir:
- **1. İki yönlü sonsuz şerit kullanan Turing makinesi**
- Temel modelde sol ucu sınırlı, sağ ucu sınırsız olan şeridin bu modelde her iki ucu da sonsuzdur. Başlangıçta şerit üzerinde belirli bir giriş dizgisi kayıtlıdır. Şerit üzerinde giriş dizgisinin sol ve sağ taraflarının sonsuz sayıda B ile dolu olduğu varsayılır.

- **2. Çok şeritli Turing makinesi**

Bu modelde makinenin bir sonlu denetim birimi, ancak n adet şeridi ve n adet okuma kafası vardır. Başlangıçta giriş dizgisi şeritlerden birinde kayıtlıdır. Diğer şeritler ise boştur. Her harekette, Turing makinesi her şeritten bir simge okur, her şeride bir simge yazar ve okuma kafalarının her biri bir sağdaki ya da bir soldaki hücreye geçer. Şeritler üzerindeki yazma ve sağa/sola geçme hareketleri birbirinden bağımsızdır. Ancak sonlu denetim birimi tek olduğu için durum geçişleri ortaktır.

6.3. TURING MAKİNESİ MODELİNDE DEĞİŞİKLİKLER

- **3. Çok izli Turing makinesi**

Bu modelde makinenin bir sonlu denetim birimi, n izli bir şeridi, n izli şerit üzerinde okuma ve yazma yapabilen bir de okuma kafası vardır. Her harekette makine her izden bir tane olmak üzere n simge okur, her ize bir tane olmak üzere n simge yazar ; okuma kafası bir sağa ya da bir sola, sonlu denetim birimi de yeni bir duruma geçer.

- **4. Birden çok okuma kafası bulunan Turing makinesi**

Bu modelde makinenin bir sonlu denetim birimi, tek izli bir şeridi, birden çok da okuma kafası bulunur. Belirli bir anda okuma kafalarının her biri şeridin belli bir hücresi üzerindedir. Her harekette okuma kafalarının her biri bulunduğu hücredeki simgeyi okur, yerine bir simge yazar ve bir sağdaki/soldaki hücreye geçer.

6.3. TURING MAKİNESİ MODELİNDE DEĞİŞİKLİKLER

- **5. Çok boyutlu Turing makinesi**

Temel modelde tek boyutlu olan okuma şeridi bu modelde çok boyutludur. Örneğin k boyutlu olan şerit $2k$ yönde sonsuzdur. Her harekette okuma kafası üzerinde bulunduğu hücredeki simgeyi okuyup, yerine yeni bir simge yazdıktan sonra $2k$ yönden birinde ilerler.

- **6. Off-line Turing makinesi**

Çok şeritli modelde olduğu gibi bu modelde de makinenin birden çok şeridi vardır. Ancak şeritlerden biri giriş şerididir ve bu şerit üzerinde yalnız okuma yapılabilir. Başlangıçta giriş şeridi üzerinde bir dizgi kayıtlıdır, diğer şeritler ise boştur. Makine her hareketinde her şeritten bir simge okur, giriş şeridi dışındaki şeritlere birer simge yazar ve her şerit üzerindeki okuma kafası bir sağ ya da sol hücreye geçer.

- **7. Deterministik olmayan Turing makinesi**

Turing makinesinin temel modeli deterministik bir modeldir. Deterministik modelde her anlık tanıma bir hareket eşlenir. Deterministik olmayan modelde ise her anlık tanıma sıfır, bir ya da birden çok hareket eşlenebilir.

6.4. CHOMSKY SIRADÜZENİ

- Sonlu özdevinir, yığıtlı özdevinir (PDA) ve Turing makinesi adlarıyla sunulan üç modelden birincisi sonlu bellekli, diğer ikisi ise sonsuz bellekli modellerdir.
- Sonlu özdevinir modelinde bilgi saklamak için kullanılan açık bir birim yoktur. Modelin saklama kapasitesi durumlarla sınırlıdır. Durum sayısı sınırlı olduğu için de model sonlu bellekli bir modeldir.
- Yığıtlı özdevinir ve Turing makinesi modellerinde ise bilgi saklamak için kullanılan sonsuz kapasiteli birer birim (yığın ve şerit) vardır. Sonsuz bellekli modellerin işlem gücünün sonlu bellekli modelden daha büyük olduğu açıktır.
- Sonsuz bellekli modellerde ise, Turing makinesinin gücü yığıtlı özdevinirden daha büyüktür. Yığıtlı özdevinir ve Turing makinesi modellerinin işlem gücü açısından farkı bellek kapasitesinden değil, modelin yapısından kaynaklanmaktadır. Turing makinesi modeli, yığıtlı özdevinir modeline göre daha güçlü bir modeldir.

6.4. CHOMSKY SIRADÜZENİ

- Biçimsel diller ve biçimsel dillere karşı gelen makine modellerini aşağıdaki gibi sıralayabiliriz :

Biçimsel Dil sınıfı	Makine Modeli
Tür-3 ya da düzgün dil	Sonlu Özdevnir (FA)
Tür-2 ya da bağlamdan-bağımsız dil	Yığıtlı Özdevnir (PDA)
Tür-1 ya da bağlama-bağımlı dil	Doğrusal-Sınırlı Özdevnir (LBA)
Tür-0 ya da kısıtlamasız dil	Turing Makinesi (TM)

6.4. CHOMSKY SIRADÜZENİ

Yukarıdaki çizelgede yer alan biçimsel dil sınıflarının tümü ile makine sınıflarının üçü bu derste incelenmektedir. Doğrusal-sınırlı özdevinirler (linear-bounded automata – LBA) olarak adlandırılan makine modeli ise, diğer üç model kadar yaygın kullanılan bir model olmadığı için bu ders kapsamına alınmamıştır. Çok kısa olarak LBA modeli, Turing makinesi modelinin şerit kapasitesi sınırlı bir biçimdir ve şerit kapasitesindeki sınırlama giriş dizgisi tarafından belirlenir.

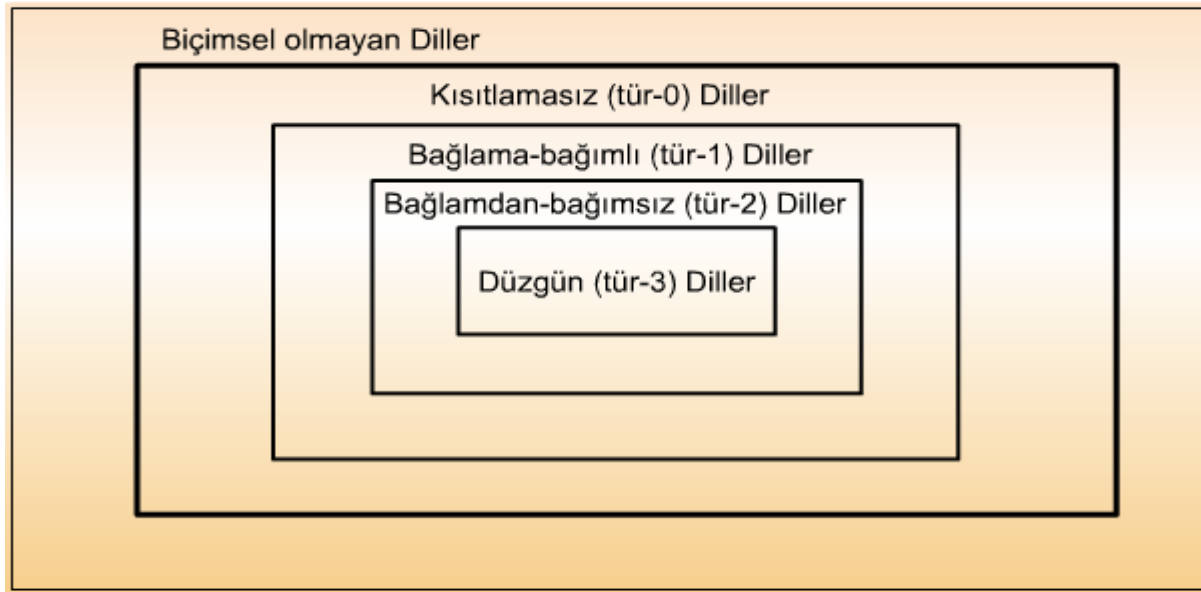
Biçimsel dil ve makine sınıfları arasındaki ilişki ise aşağıdaki gibi özetlenebilir :

1. Sonlu özdevinirler düzgün dilleri tanıyan makinelerdir.
2. Yığıtlı özdevinirler bağlamdan-bağımsız dilleri tanıyan makinelerdir.
3. Doğrusal-sınırlı özdevinirler bağlama-bağımlı dilleri tanıyan makinelerdir.
4. Turing makineleri ise kısıtlamasız dilleri tanıyan makinelerdir.

6.4. CHOMSKY SIRADÜZENİ

Yukarıda sıralanan dil sınıfları birbirinden bağımsız değildir. Biçimsel dil sınıfları arasındaki sıradüzen Chomsky sıradüzeni (Chomsky hierarchy) olarak bilinir ve kısaca aşağıdaki gibi ifade edilebilir (Çizim 6.2):

1. Her bağlama-bağımlı dil aynı zamanda bir kısıtlamasız dildir. Başka bir deyişle bağlama-bağımlı dil sınıfı kısıtlamasız dil sınıfının bir alt sınıfıdır.
2. Her bağlamdan-bağımsız dil aynı zamanda bir bağlama-bağımlı dildir. Başka bir deyişle bağlamdan-bağımsız dil sınıfı, bağlama-bağımlı dil sınıfının bir alt sınıfıdır.
3. Her düzgün dil aynı zamanda bir bağlamdan-bağımsız dildir. Başka bir deyişle düzgün diller sınıfı, bağlamdan bağımsız diller sınıfının bir alt sınıfıdır.

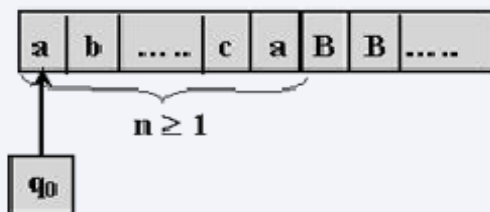


$L_{S.6.1}$ dili, $\{a, b, c\}$ alfabesinde, içindeki b ve c 'lerin sayılarının toplamı, a 'ların , sayısının iki katı olan dil olarak tanımlanıyor. Bu dil λ içermiyor. Dilin tümcelerinden birkaç örnek aşağıda görülmektedir.

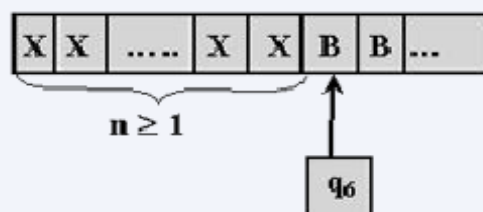
$$L_{S.6.1} = \{ abb, cab, aacbbb, cbabca, baacacbbc, \dots \}$$

Aşağıdaki Turing makinesinin tanımını, $L_{S.6.1}$ dilini tanıyan bir Turing makinesi alışturacak biçimde tamamlamanız isteniyor.

Başlangıç konfigürasyonu:



Bitiş konfigürasyonu:



$$M_{S.6.1} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q = \{ q_0, q_1, q_2, q_3, q_4, q_5, q_6 \}$$

$$\Sigma = \{ a, b, c \}$$

$$\Gamma = \{ a, b, c, X, Y, B \}$$

q_0 : Başlangıç durumu

$$F = \{ q_6 \}$$

a) $L_{S.6.1}$ dilinin tümceleri $3n$ uzunluğundadır ($n \geq 1$). Bir tümce verildiğinde, eğer tümce geçerli bir tümce ise, $M_{S.6.1}$ makinesi tümceyi soldan sağa en çok n kez, sağdan solda en çok $(n+1)$ kez tarayarak bitiş konfigürasyonuna ulaşacaktır. Bu bağlamda, $M_{S.6.1}$ 'in çalışma ilkesini açıklayınız.

b) Makinenin biçimsel tanımını tamamlayınız. Bu bağlamda makinenin hareketlerini bir geçiş çizeneği ile gösteriniz.

a	b	c	B
---	---	---	------	---

b	a	c	B
---	---	---	------	---

c	b	a	B
---	---	---	------	---

a	b	c	c	c	a	B
---	---	---	---	---	---	------	---

b	b	c	c	a	a	B
---	---	---	---	---	---	------	---

6.5.1. Cevap1

b) Makinenin biçimsel tanımı:

$$M_{5.6.1} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q = \{ q_0, q_1, q_2, q_3, q_4, q_5, q_6 \}$$

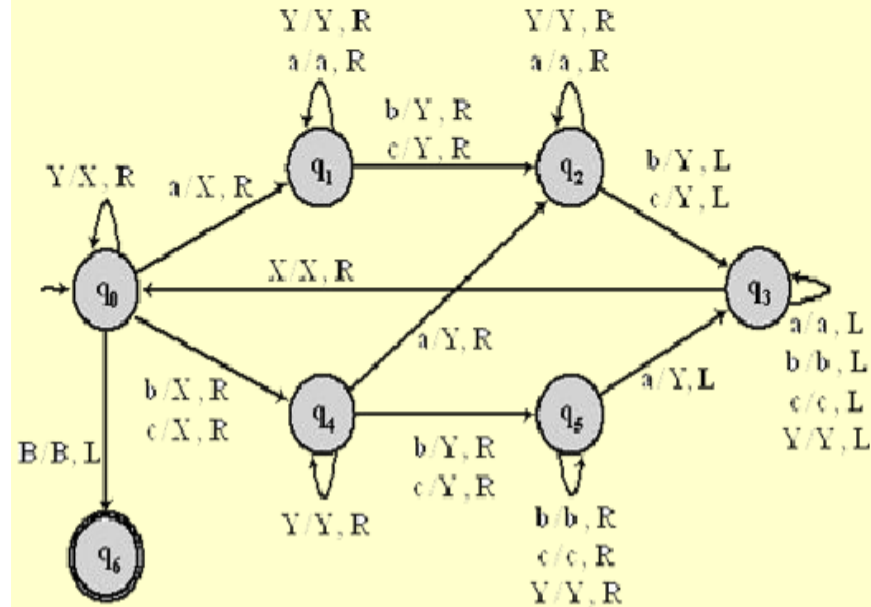
$$\Sigma = \{ a, b, c \}$$

$$\Gamma = \{ a, b, c, X, Y, B \}$$

q_0 : Başlangıç durumu.

$$F = \{ q_6 \}$$

δ :



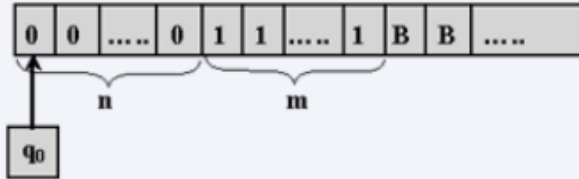
a) Makinenin çalışma ilkesi:

- Makine dizgiyi soldan sağa doğru tararken bir tane **a** ile iki tane **b** ya da **c**'yi siler. Bu üç simge herhangi bir sırada olabilir.
- Silinen üç simgeden ilkinin yerine **X**, diğer ikisinin yerine de **Y** yazılır.
- Dizgi sağdan sola, ilk **X** bulununcaya kadar taranır.
- Tüm **a**, **b** ve **c** simgeleri bittikten sonra tüm **Y**'ler **X** yapılır.

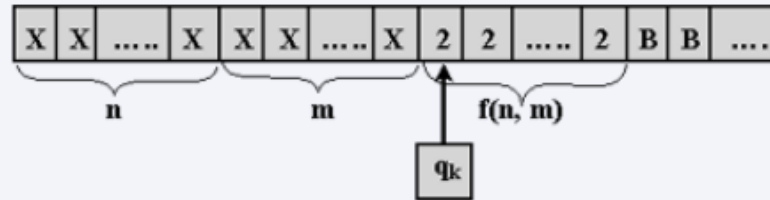
$$f(n, m) = \lceil (n + m) / 2 \rceil \quad n \geq 1, \quad m \geq 0$$

Yukarıdaki işlevin değerini hesaplayan bir Turing makinesini tasarlamamız isteniyor.

Başlangıç konfigürasyonu:



Bitiş konfigürasyonu:



- Makinenin nasıl çalışacağını sözel olarak açıklayınız
- Makinenin biçimsel tanımını veriniz. Bu bağlamda makinenin hareketlerini bir çizenek ile gösteriniz.

$$M_{S6.10} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q = \{ q_0, q_1, q_2, q_3, q_4, q_5, q_6 \}$$

$$\Sigma = \{ 0, 1 \}$$

$$\Gamma = \{ 0, 1, B, X, \epsilon, \$ \}$$

$$F = \{ q_6 \}$$

δ :

	ϵ	0	1	\$	X	B
q_0	-	-	(q_1, X, R)	-	-	-
q_1	-	$(q_1, 0, R)$	$(q_1, 1, R)$	$(q_2, \$, R)$	-	-
q_2	-	-	$(q_2, 1, R)$	-	-	$(q_3, 1, L)$
q_3	-	$(q_3, 0, L)$	$(q_3, 1, L)$	$(q_3, \$, L)$	(q_4, X, R)	-
q_4	-	$(q_5, 0, L)$	(q_1, X, R)	$(q_5, \$, L)$	-	-
q_5	(q_6, ϵ, R)	-	-	-	$(q_5, 1, L)$	-
q_6	-	-	-	-	-	-

a) Aşağıdaki başlangıç konfigürasyonlarından her biri için makinenin çıkış konfigürasyonunu bulunuz.

1) $\epsilon \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 1 \ \$ \ B \ B \ \dots$
 \uparrow

2) $\epsilon \ 1 \ 1 \ 1 \ 1 \ \$ \ B \ B \ \dots$
 \uparrow

3) $\epsilon \ 0 \ 1 \ 1 \ 1 \ 0 \ 1 \ \$ \ B \ B \ \dots$
 \uparrow

4) $\epsilon \ 1 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ \$ \ B \ B \ \dots$
 \uparrow

b) Makinenin çalışmasını sözel olarak açıklayınız.

6.5.10. Cevap 10

a)

$w_1 : \quad \epsilon \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 1 \quad \$ \quad \textcircled{1} \quad B \quad \dots$
 $\quad \quad \quad \uparrow$

$w_2 : \quad \epsilon \quad 1 \quad 1 \quad 1 \quad 1 \quad \$ \quad \textcircled{1 \ 1 \ 1 \ 1} \quad B \quad \dots$
 $\quad \quad \quad \uparrow$

$w_3 : \quad \epsilon \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad \$ \quad B \quad B \quad \dots$
 $\quad \quad \quad \uparrow$

$w_4 : \quad \epsilon \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 0 \quad \$ \quad \textcircled{1 \ 1 \ 1} \quad B \quad \dots$
 $\quad \quad \quad \uparrow$

b)

Turing makinesi şerit üzerinde ϵ ve $\$$ simgeleri arasında yer alan sözcüğün salt 1'lerden oluşan önekini $\$$ 'den sonraya kopyalıyor. Yapılan işlem düzgün deyim notasyonu ile aşağıdaki gibi özetlenebilir.

Başlangıç görünümü :

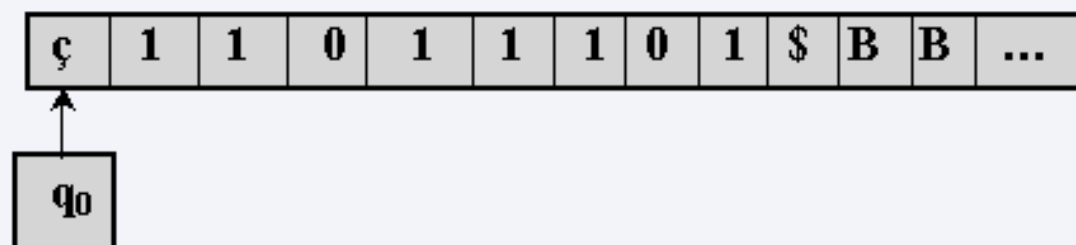
$$\epsilon \ w_1 \ w_2 \ \$ \ B \ B \ B \ \dots \quad w_1 = 1^*, \quad w_2 = 0(0+1)^* + \lambda$$

Bitiş görünümü:

$$\epsilon \ w_1 \ w_2 \ \$ \ w_1 \ B \ B \ B \ \dots$$

Örnek 6.11

$\{0, 1\}$ alfabetinde, uzunluğu en az iki olan ve içinde çift sayıda **0** ile çift sayıda **1** bulunan dizgiler kümesini tanıyan bir Turing makinesi tasarlamamız isteniyor. Örnek bir giriş dizgisi için makinenin başlangıç konfigürasyonu aşağıda görülmektedir



- a) Turing makinesinin nasıl çalışacağını açıklayınız.
- b) **TM**'nin biçimsel tanımını, hareketleri hem bir çizelge, hem de bir çizenek ile göstererek, veriniz.

$$M_{S.6.12} = \langle Q, \Sigma, \Gamma, \delta, q_0, B, F \rangle$$

$$Q = \{q_0, q_1, q_2, q_3, q_4\}$$

$$\Sigma = \{0, 1\}$$

$$\Gamma = \{0, 1, B\}$$

$$F = \{q_4\}$$

δ :

	0	1	B
q_0	(q_1, B, R)	-	(q_4, B, R)
q_1	$(q_1, 0, R)$	$(q_1, 1, R)$	(q_2, B, L)
q_2	-	(q_3, B, L)	-
q_3	$(q_3, 0, L)$	$(q_3, 1, L)$	(q_0, B, R)
q_4	-	-	-

Başlangıç konfigürasyonu:



- Yukarıda biçimsel tanımı verilen Turing makinesinin çalışmasını sözel olarak açıklayınız ve tanıdığı dilin küme tanımını veriniz
- Turing makinesinin tanıdığı dilin türü nedir? Bu dili türeten bir dilbilgisi tanımlayınız.
- $w = 011$ için anlık tanımlar (ID) dizisini oluşturarak makinenin dizgiyi tanıyıp tanımadığını bulunuz

6.5.12. Cevap 12

a) Biçimsel tanımı verilen Turing makinesi ($M_{S.6.12}$) giriş dizgisinden en soldaki 0 ile en sağdaki 1'i siliyor. Bu işlemin belirli sayıda tekrarlanmasıyla tüm simgeler silinebiliyorsa dizgi tanınıyor.

$$L_{S6.12} = \{ 0^n 1^n \mid n \geq 1 \}$$

b) $L_{S6.12}$ bağlamdan-bağımsız bir dildir.

$$G_{S6.12} = \langle V_N, V_T, P, S \rangle$$

$$V_N = \{ S, A \}$$

$$V_T = \{ 0, 1 \}$$

$$P : \quad S \Rightarrow 0A1$$

$$A \Rightarrow 0A1 \mid \lambda$$

c) $w = 011$:

$$\begin{aligned} (q_0, 011) &\vdash (B, q_1, 11) \vdash (B1, q_1, 1) \vdash (B11, q_1) \\ &\vdash (B1, q_2, 1) \vdash (B, q_3, 1B) \vdash (q_3, B1B) \\ &\vdash (B, q_0, 1B) \end{aligned}$$

w $M_{S6.12}$ tarafından tanınmaz.