Düzgün Deyimler Ve Düzgün Kümeler

Sonlu Otomatların Matematiksel Küme Gösterimi

2.1 Düzgün Kümeler

- Sonlu özdevinir (finite automata : FA) modeli, giriş alfabesindeki simgelerden oluşan dizgilerin bir kısmını tanıyan, bir kısmını ise tanımayan bir modeldir.
- Her sonlu özdevinirin tanıdığı bir dizgiler kümesi vardır. Bu küme, içerdiği dizgiler tek tek yazılarak, ya da içerdiği dizgilerin özellikleri belirtilerek tanımlanabilir.
- Sonlu özdevinirlerin tanıdığı kümelerin tanımları sözel olarak ya da matematiksel gösterimler kullanılarak yapılabilir.

Düzgün Küme Örnek:

- { 0, 1 } alfabesinde, sonlu bir özdevinir tarafından tanınan aşağıdaki kümeleri tanımlayabiliriz:
- $\mathbf{P_1}$= {1001,0110}
- $\mathbf{P_2}$= İçinde ardarda en az üç tane 1 bulunan (ya da 111 altdizgisi içeren) dizgiler kümesi
- $\mathbf{P_3}$= İlk ve son simgesi 1 olan en çok 5 uzunluğundaki dizgiler kümesi
- $\mathbf{P_4}$=$\{0^n1^m \mid n\geq1, m\geq1\}$

2.1 Düzgün Kümeler:

- Sonlu özdevinirler tarafından tanınan kümelere **düzgün kümeler (regular sets)** denir.
- Bir küme verildiğinde, bu kümenin düzgün bir küme olup/olmadığı, bu kümeyi tanıyan sonlu bir özdevinirin bulunup/bulunmaması tarafından belirlenir.
- **Eğer kümeyi tanıyan en az bir sonlu özdevinir varsa, bu küme düzgün bir kümedir**.
- Eğer verilen kümeyi tanıyan hiçbir sonlu özdevinir yoksa, bu küme düzgün bir küme değildir.
- $\mathbf{P_1}$, $\mathbf{P_2}$, $\mathbf{P_3}$ ve $\mathbf{P_4}$ kümeleri düzgün kümelerdir. Çünkü bu kümelerin her birini tanıyan en az bir sonlu özdevinir oluşturulabilir.

Düzgün Küme Değildir:

- $\mathbf{P_5}$=$\{0^n1^n \mid n \geq 1\}$
- $\mathbf{P_6}$= İçinde eşit sayıda 0 ve 1 bulunan dizgiler kümesi
- $\mathbf{P_5}$ ve $\mathbf{P_6}$ kümelerinin her ikisi de düzgün değildir. Çünkü bu kümeleri tanıyan sonlu özdevinirler oluşturulamaz.
- Tanımlanan bir kümenin düzgün olup olmadığını bulmak için, kümeyi tanıyan bir sonlu özdevinir oluşturulmaya çalışılır. Eğer bunda başarılı olunur ve kümeyi tanıyan sonlu bir özdevinir oluşturulabilirse, kümenin düzgün bir küme olduğu anlaşılır.

$P_5=\{0^n1^n \mid n\geq1\}$

- **P_5**'i tanıyacak bir makineyi (**M_5**) düşünelim.
- Bu makinenin, dizginin başındaki **0**'ların sayısını saklaması ve **1**'lerin sayısının **0**'ların sayısına eşit olduğunu denetlemesi gerekir.
- Küme tanımında, kümedeki dizgilerin n tane **0**'dan sonra n tane **1** içerdiği belirtilmekte ve n'nin birden büyük olduğu belirtilmektedir.
- Ancak n'nin üst sınırı yoktur.
- n çok büyük bir sayı olabilir. Bu nedenle, sonlu sayıda durumu bulunan, bu nedenle sonlu saklama kapasitesine sahip bir FA ile **P_5**'i tanımak mümkün değildir.
- **P_5** kümesini tanıyan bir makine (**M_5**) varsa, bu makine sonlu bir özdevinir değildir.

P_6= İçinde eşit sayıda 0 ve 1 bulunan dizgiler kümesi

- **P_6**'yı tanıyacak bir makinenin (**M_6**), dizgiyi taraması ve sürekli olarak, o ana kadar raslanan **0**'larla **1**'lerin sayılarının farkını saklaması gerekir.
- Eğer dizginin sonunda, **0**'larla **1**'lerin sayılarının farkı sıfırsa, dizgi **M_6** tarafından tanınacaktır.
  Ancak **0**'larla **1**'lerin sayılarının farkının bir üst sınırı yoktur. Bu fark, düşünülebilecek tüm sonlu sayılardan daha büyük olabilir. Bu nedenle de **M_6** bir sonlu özdevinir olamaz.
- **P_5** ve **P_6** kümelerini tanıyan sonlu özdevinirler bulunamadığı için, bu kümeler düzgün kümeler değildir.

2.2 Düzgün Deyimler

- **Düzgün deyimler**, düzgün kümeleri biçimsel olarak tanımlamak için kullanılan bir anlatım aracı, bir dildir. Her düzgün deyim, belirli bir alfabedeki simgelerden oluşturulabilecek dizgilerin bir altkümesini tanımlar.
- **Tanım 2.1**
- { a, b, c, ... } alfabesindeki düzgün deyimler özyineli olarak aşağıdaki gibi tanımlanır:
- Alfabedeki her simge düzgün deyimdir:
  - a={a},
  - b={b}, ...
- λ ve Φ birer düzgün deyimdir.
  - λ={λ}
  - Φ={ }

Tanım 2.1 devamı:

- Eğer P ve Q birer düzgün deyimse:
- P+Q   PQ   P* da birer düzgün deyimdir.
- Burada 3 tane küme işleci tanımlanmaktadır.
- "+" küme birleşim işlecidir. $P+Q= P\cup Q$
- "." veya boşluk( ) ardarda ekleme (concatenation) işlecidir:
- $PQ=\{pq \mid p\varepsilon P, q\varepsilon Q\}$
- " * " kapanış (closure) işlecidir.
- $P^*=\lambda+ P+ PP+ PPP+ PPPP+\ldots$
- Bu kuralların sonlu sayıda uygulanması ile oluşturulan deyimler düzgün deyimlerdir.

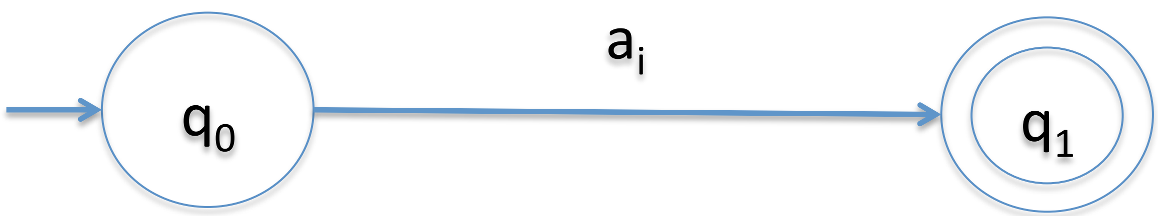
Örnekler:

Düzgün Deyim	Tanımladığı Küme
Φ	{}
λ	{ λ }
00+11	{00, 11}
a(b+c)	{ab, ac}
ab*	{a, ab, abb, abbb, abbbb, …}
a(bb+cc)d*	abb, abbd, abbdd,…, acc, accd, accdd,
a*	{λ, a, aa, aaa, aaaa,…}
(0+1)*	{λ, 0,1, 00, 01, 10, 11, 000, 001, 010, 011, 100, …}
a(b+cd*)*a	{aa, aba, acda, acdda, acddda, abca, acba, …}

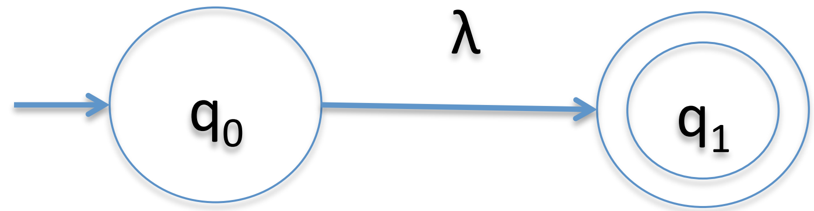
2.2.1. Düzgün Deyimlere Karşı Gelen Sonlu Özdevinirlerin Bulunması

- **Önerme 2.1.** Düzgün deyimlerle tanımlanan kümeler düzgün kümelerdir.
- Bu önermeye göre, bir düzgün deyim verildiğinde, bu düzgün deyimle tanımlanan kümeyi tanıyan bir sonlu özdevinir bulunabilir.
- Bu önerme, düzgün deyimlere karşı gelen sonlu özdevinirlerin geçiş çizeneklerinin nasıl bulunacağı gösterilerek ispat edilmeye çalışılacaktır.
- Bunun için de, düzgün deyimlerin özyineli tanımında yer alan her bir madde için ilgili geçiş çizeneğinin nasıl bulunacağı gösterilecektir.

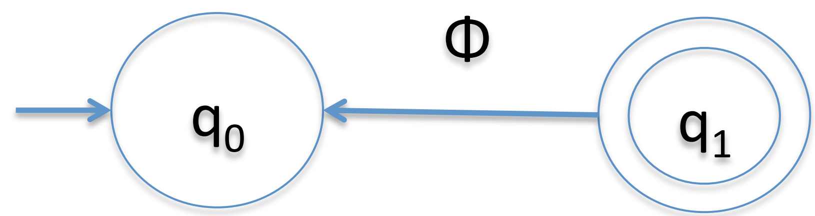
- $\mathbf{P_1 = a_i}$ düzgün deyimi ile tanımlanan küme tek bir giriş simgesi içermektedir. Bu kümeyi tanıyan sonlu özdevinirin geçiş çizeneği yandaki gibi oluşturulabilir.



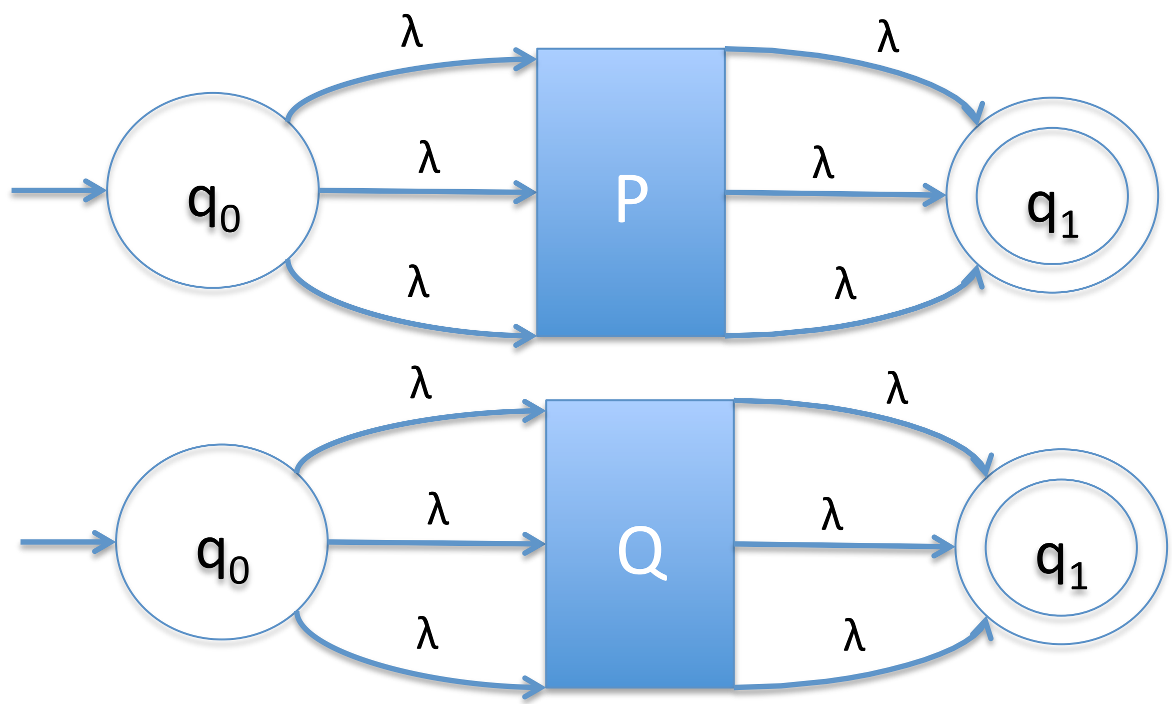
- **P_2** = düzgün deyimi ile tanımlanan küme yalnız boş simgeyi (λ) içeren kümedir. Bu kümeyi tanıyan sonlu özdevinirin geçiş çizeneği yandaki gibi oluşturulabilir



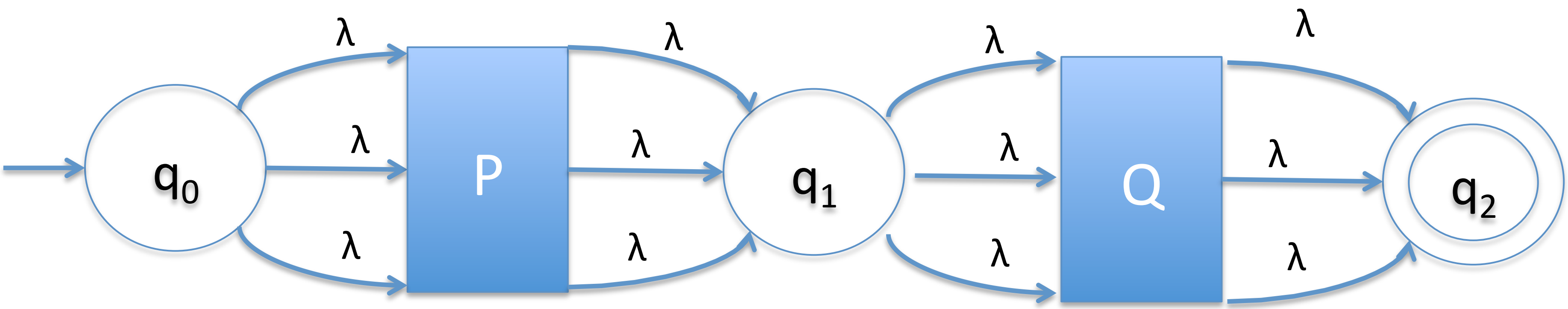
- **P_3** = düzgün deyimi ile tanımlanan küme boş kümedir. Bu kümeyi tanıyan sonlu özdevinirin geçiş çizeneği yandaki gibi oluşturulabilir. turulabilir.



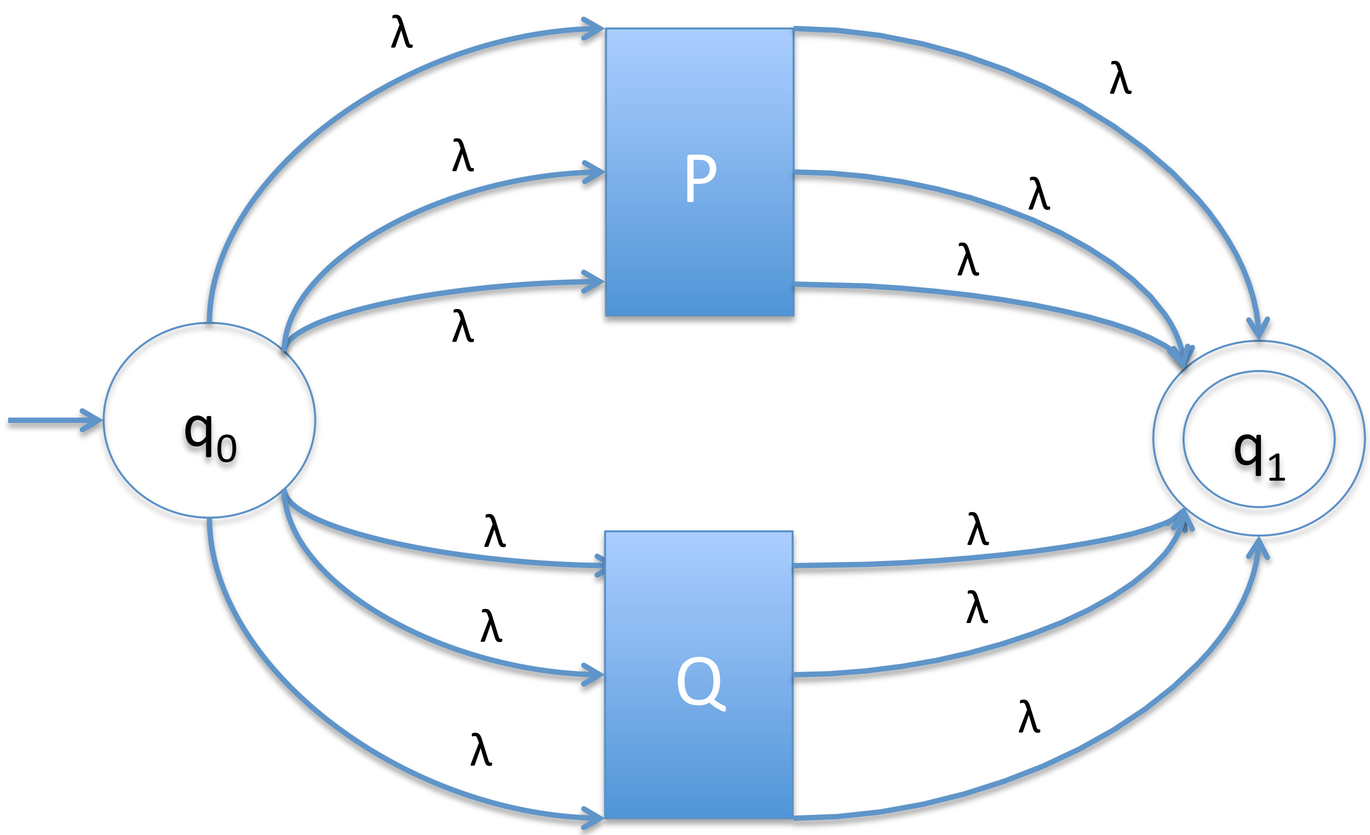
- **P** ve **Q** birer düzgün deyim olsun. Bu düzgün deyimlerle tanımlanan kümeleri tanıyan geçiş çizeneklerinin bilindiğini varsayalım.
Eğer **P** ve **Q**'yu tanıyan geçiş çizeneklerinde birden çok başlangıç ve/ya da uç durum varsa bu geçiş çizeneklerini bir başlangıç durumu, bir de uç durumu bulunan eşdeğer çizeneklere dönüştürebiliriz.
- Bunun için, eğer çizenekte birden çok başlangıç durumu varsa, çizeneğe yeni bir durum **(q_0)** eklenerek tek başlangıç durumu bu yeni durum yapılır ve bu yeni durum ile eski başlangıç durumları arasına birer λ - geçişi konulur.
- Eğer çizenekte birden çok uç durum varsa, çizeneğe yeni bir uç durum **(q_1)** eklenerek tek uç durum bu yeni durum yapılır ve eski uç durumlar ile bu yeni durum arasına birer λ-geçişi konulur.



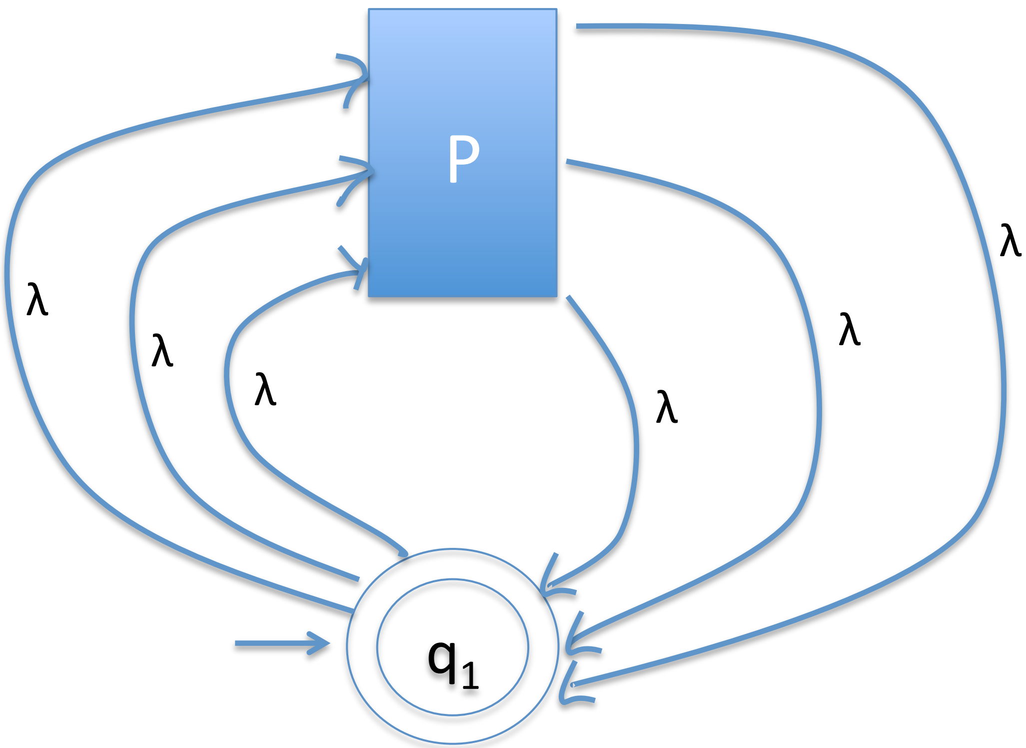
PQ (P.Q)



P+Q

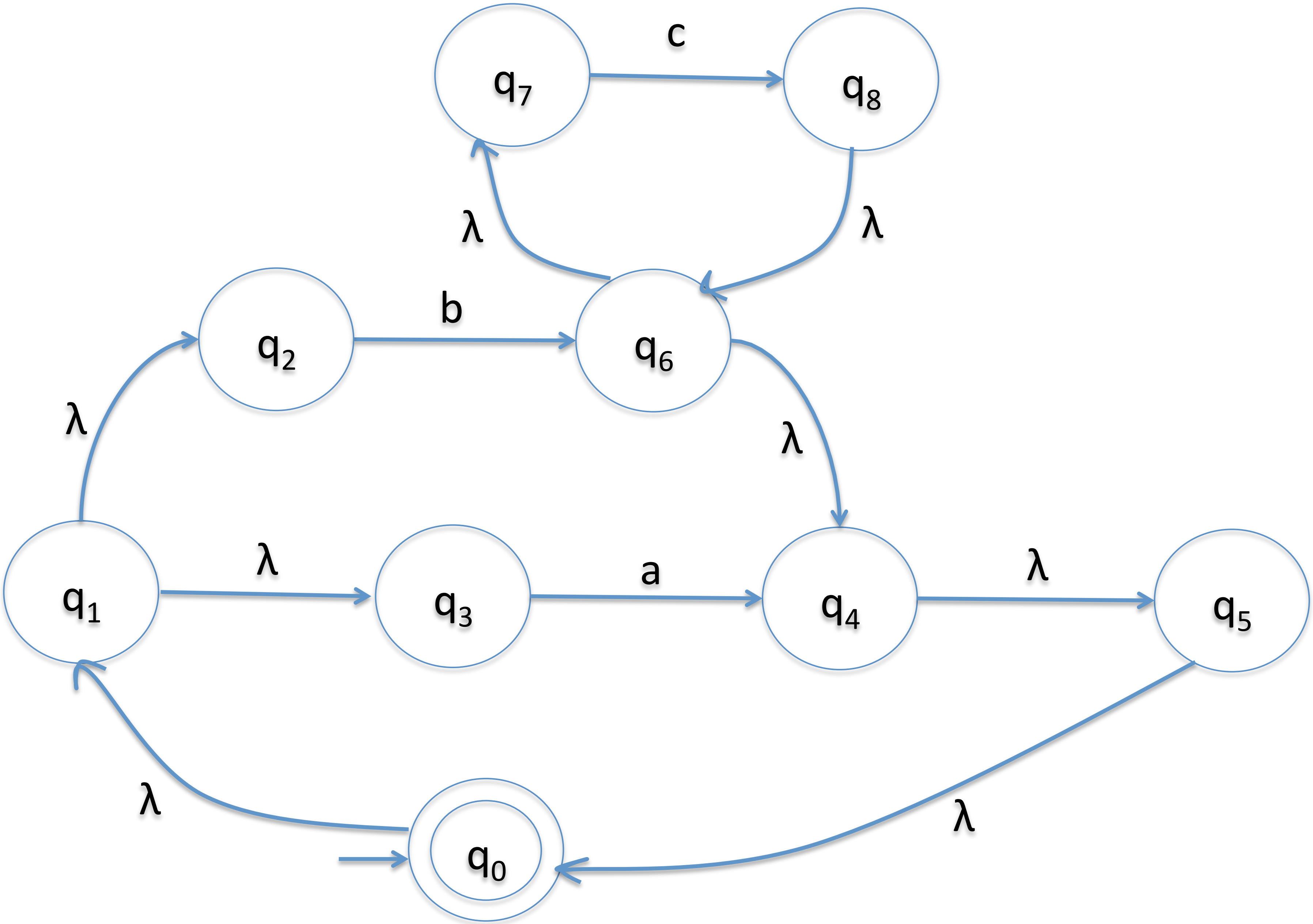


P*



$P=(a+bc^*)^*$ deyimine karşılık gelen sonlu otomat:

- Verilen bir düzgün deyimi tanıyan geçiş çizeneğinin biçimsel yaklaşımla oluşturulması çok uzun olabilir ve bu yöntemle oluşturulan geçiş çizeneği çok sayıda λ -geçişi içerebilir.
- **$P=(a+bc^*)^*$**
  düzgün deyimini tanıyan sonlu özdevinirin geçiş çizeneğini biçimsel yaklaşımla oluşturmak istediğimizde,
  - önce **(a)**'yı, **(b)**'yi ve **(c)**'yi tanıyan çizenekleri,
    sonra **(c^*)**'ı tanıyan çizeneği,
  - sonra **(bc^*)**'ı tanıyan çizeneği,
  - sonra **($a + bc^*$)**'ı tanıyan çizeneği,
  - son olarak da **($(a+ bc^*)^*$)**'ı tanıyan çizeneği oluşturmak gerekir
- Oysa **P**'yi tanıyan geçiş çizeneğini sezgisel yaklaşımla çok daha çabuk oluşturmak mümkün olabilir.


c
λ
λ
b
λ
λ
λ
a
λ
λ
λ
q_7
q_8
q_2
q_6
q_1
q_3
q_4
q_5
q_0

$P = (a + bc^*)^*$ Sezgisel Yaklaşım:

Sezgisel yaklaşımla oluşturulan geçiş çizeneği, biçimsel yaklaşımla oluşturulan geçiş çizeneğine oranla çok daha az λ –geçişi içeren, çok daha yalın bir çizenektir.

Geçiş çizeneklerinin sezgisel yaklaşımla oluşturulması sırasında dikkat edilmesi gereken önemli hususlardan birkaçı aşağıda yer almaktadır.

1. Eğer düzgün deyimde:

… $+ a^* b$… ya da $(a^* b \ldots.)^*$

gibi bir örüntü varsa, üzerinde a döngüsü bulunan düğüme λ-geçişi ile gelinmelidir.

2.Eğer düzgün deyimde:

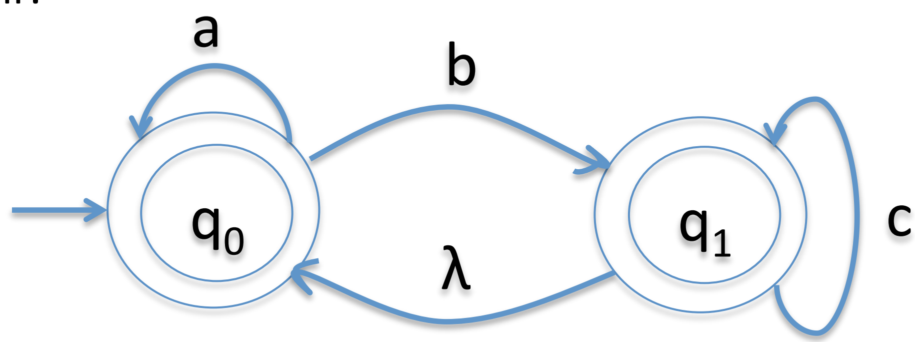
… $a b^* +$ … ya da … $a b^*)^*$

gibi bir örüntü varsa, üzerinde b döngüsü bulunan düğümden λ –geçişi ile çıkılmalıdır.

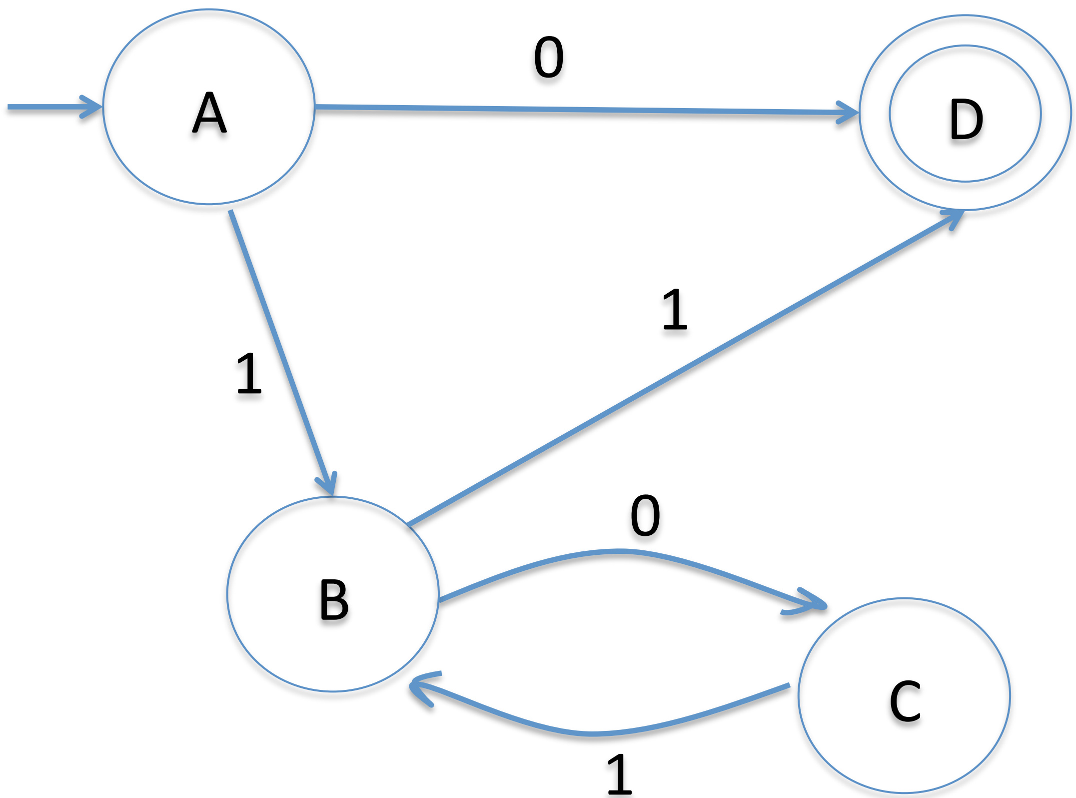
3. Eğer düzgün deyimde:

… $a^* b^*$…

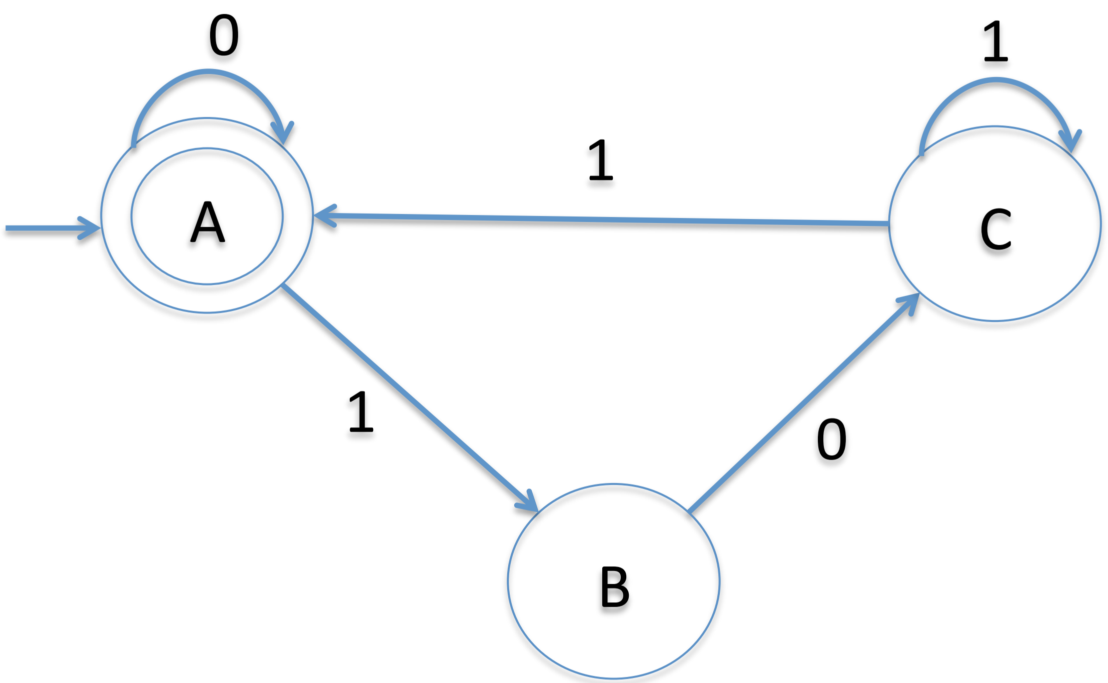
gibi bir örüntü varsa, üzerinde a döngüsü bolunan düğümden, üzerinde b döngüsü bolunan düğüme λ–geçişi ile geçilmelidir.



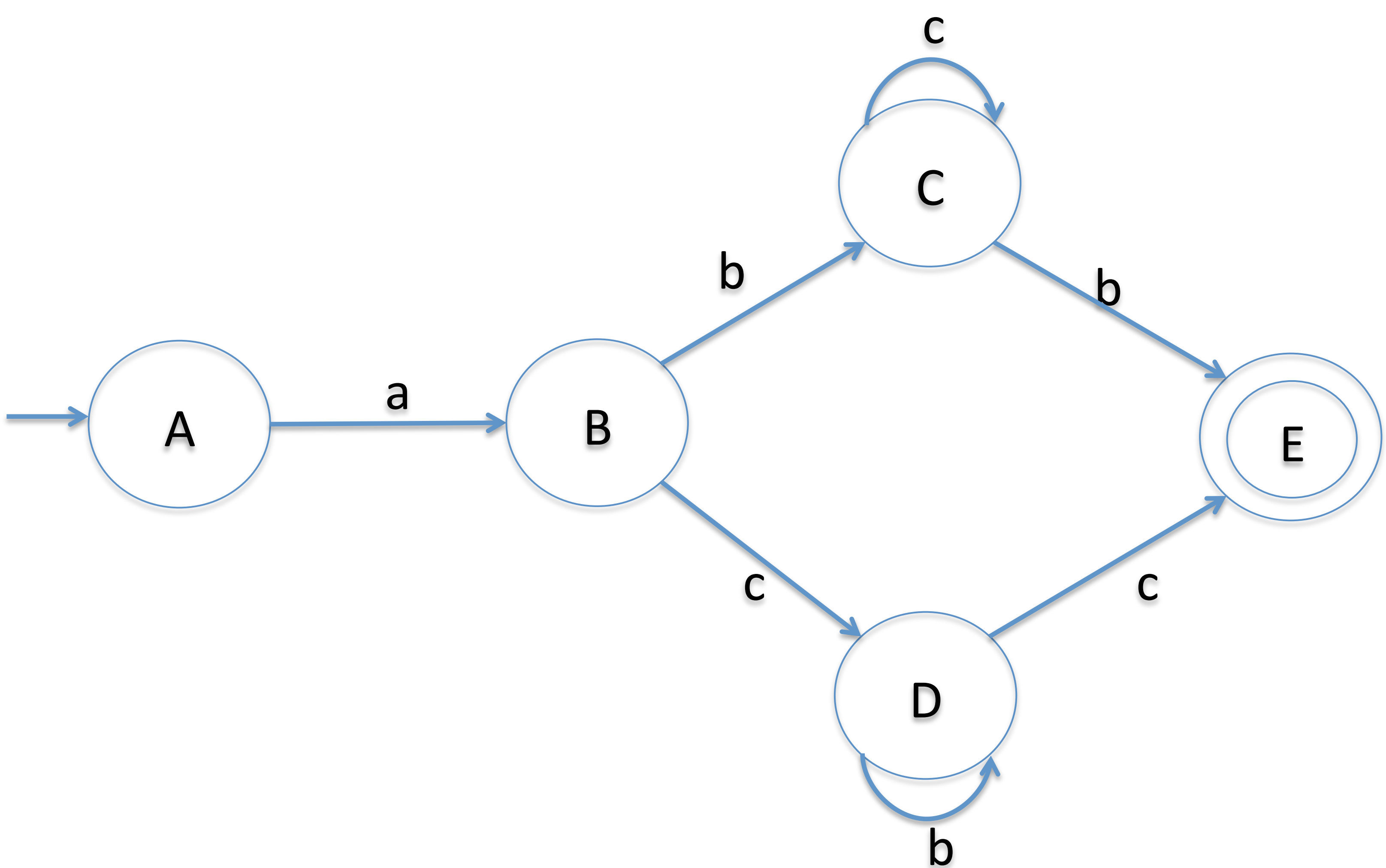
Örnek a) $P_1 = 0+1(01)^*1$



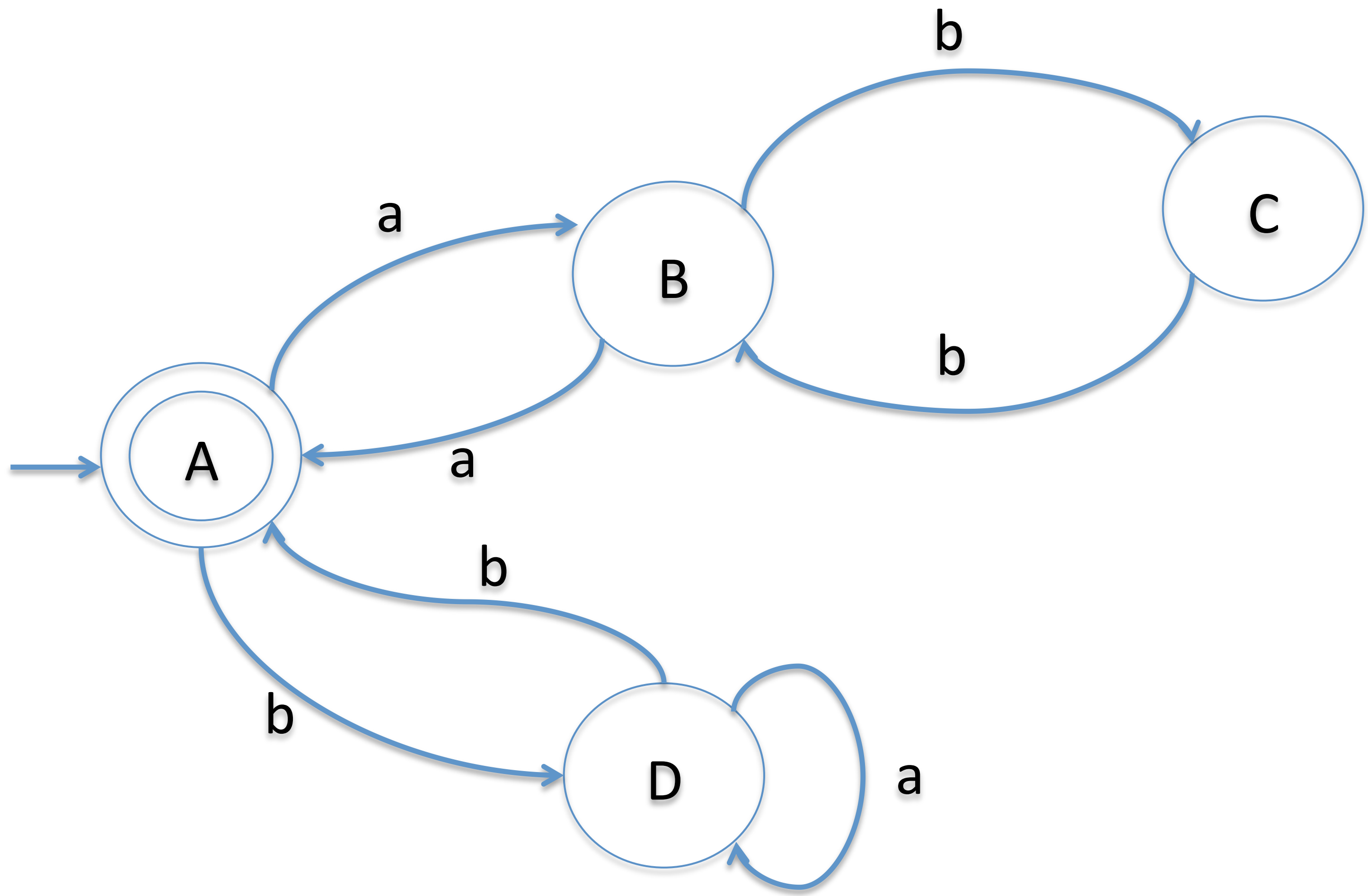
b) $P_2 = (0+101^*1)^*$



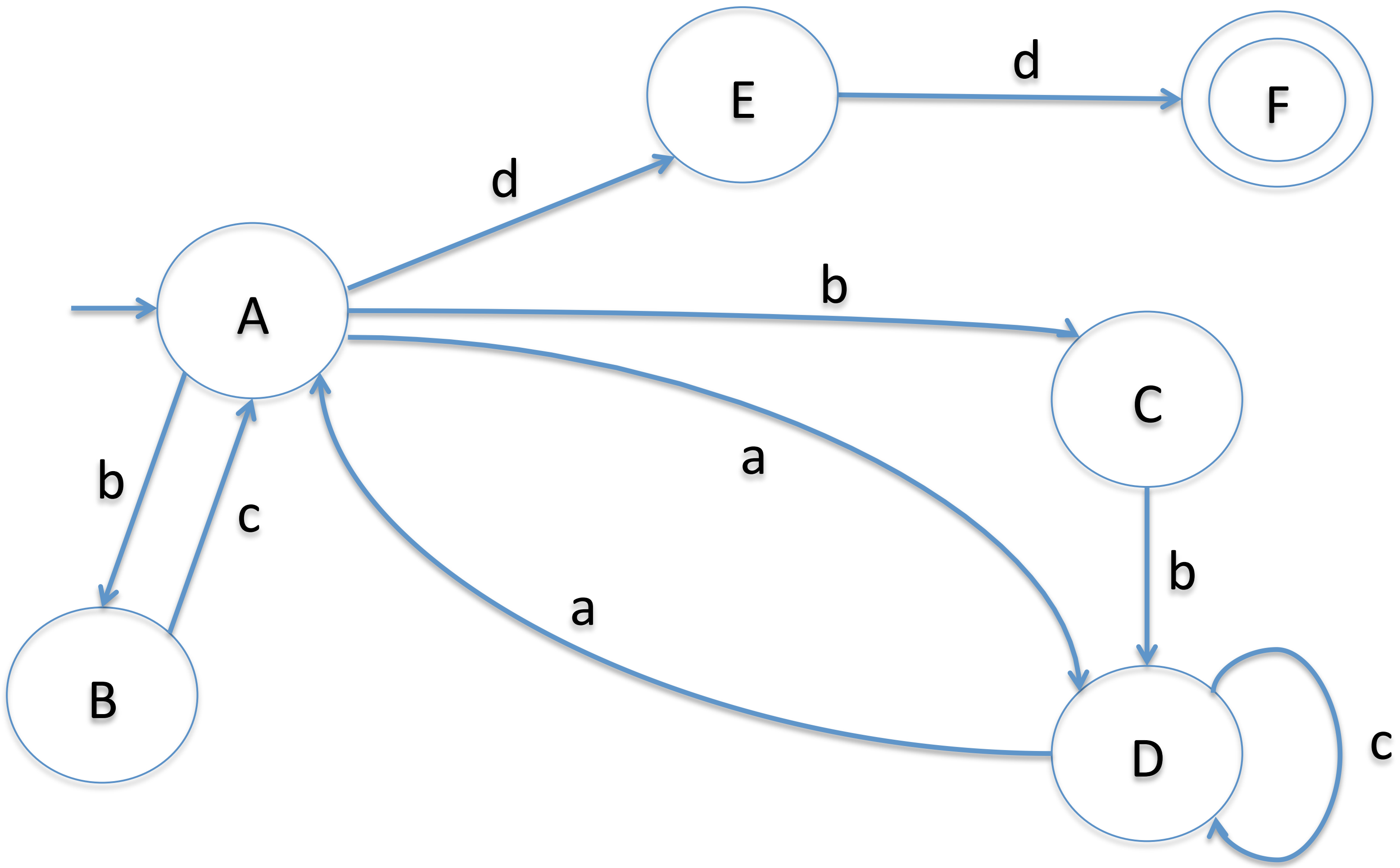
c) $P_3 = a(bc^*b+cb^*c)$



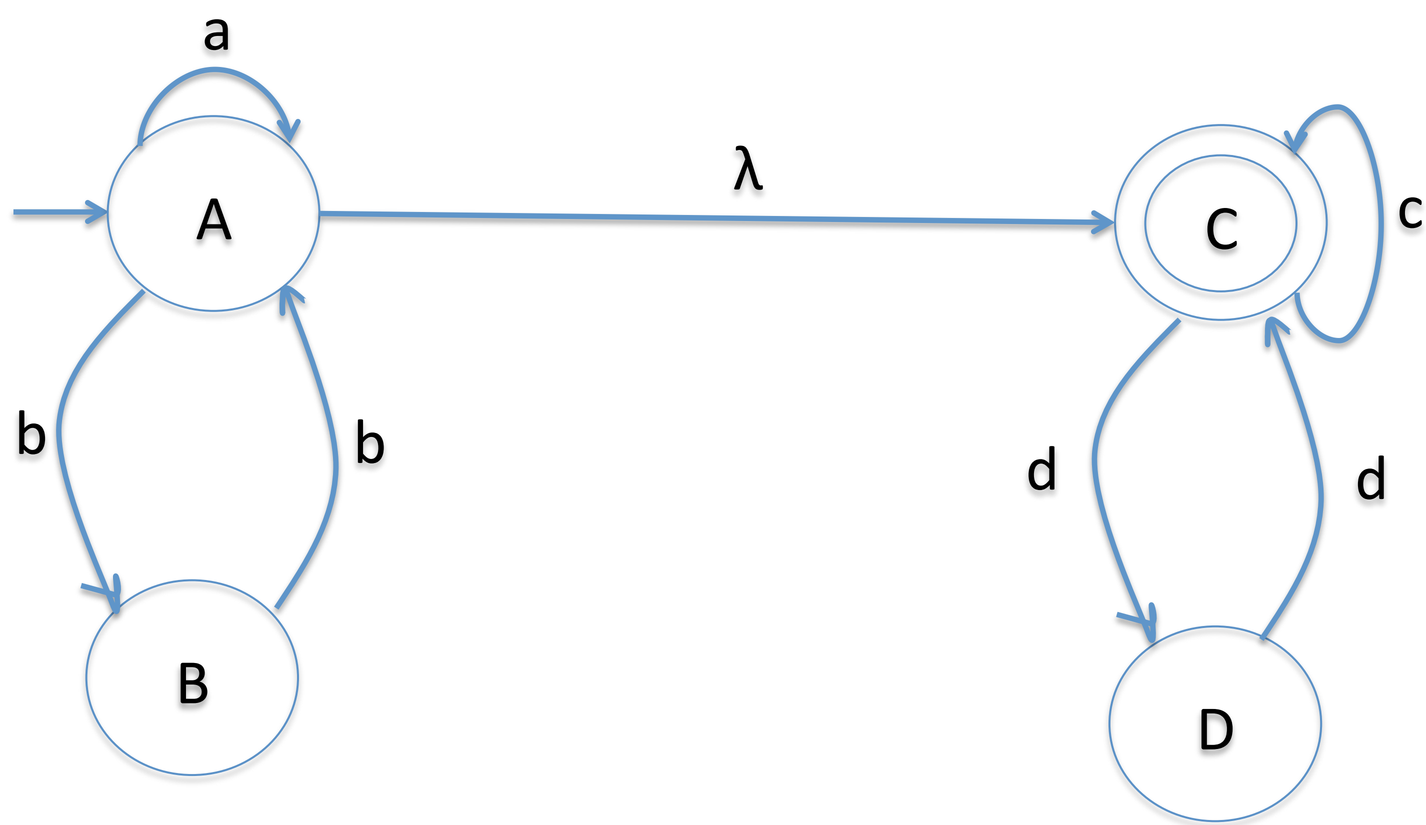
c) $P_4 = (a(bb)^*a+ba^*b)^*$



c) $P_5 = (bc+(a+bb)c^*a)^*dd$



c) $P_6 = (a+bb)^*(c+dd)^*$



2.2.2. Sonlu Özdevinirlerin Tanıdığı Kümelerin Birer Düzgün Deyim Olarak Bulunması

- **Teorem 2.2:** Her düzgün küme bir düzgün deyimle gösterilebilir. Düzgün kümeler, sonlu otomatlar tarafından tanınan kümeler olduğu için, verilen bir sonlu otomatın tanıdığı düzgün kümeyi bir düzgün deyimle göstermek mümkündür.
- Düzgün deyimin elde edilmesi için kullanılacak yöntem, denklem sistemi çözmeye dayalı olacaktır. Denklem sisteminin nasıl kurulacağı ve çözüleceğini örnekler üzerinde göstermeden önce, denklem sisteminin çözülmesinde yararlanılacak bir özelliği tanımlamak gereklidir.
- **Teorem 2.3:** P, Q ve R aynı alfabede tanımlanmış düzgün deyimler ise **P, λ'yı içermiyorsa:**
- **R=Q+RP ise denklemin tek çözümü: R=QP* dır.**

2.2.2. Sonlu Özdevinirlerin Tanıdığı Kümelerin Birer Düzgün Deyim Olarak Bulunması

- Tek başlangıç durumu bulunan bir geçiş çizeneği verildiğinde, her durum için aynı adı taşıyan bir düzgün deyim (küme değişkeni) tanımlanır.
- **A** durumu için tanımlanan **A** değişkeni, başlangıç durumundan başlayıp **A** durumunda son bulan tüm dizgileri içeren bir kümeye karşı gelir.
- Durumlar arası geçişler dikkate alınarak, her **A** durumu için, sol tarafında bu duruma karşı gelen değişken; sağ tarafında ise, **A** durumunda son bulan her

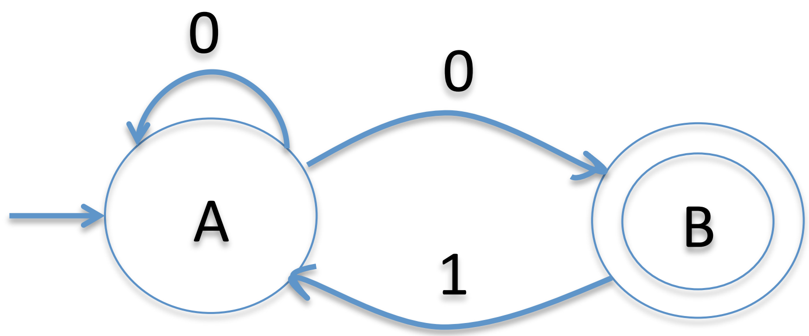
$$\delta(B, a_i) = A$$

- geçişi için Ba_i çarpımının yer aldığı

$$A = \Sigma Ba_i$$

- biçiminde bir denklem kurulur.
- Daha sonra da, teorem 2.3'den yararlanılarak denklem sistemi çözülür ve her değişken için bir düzgün deyim elde edilir. Sonlu özdevinirin tanıdığı küme, uç durumlara karşı gelen düzgün deyimlerin küme birleşimi **(+)** ile gösterilir.

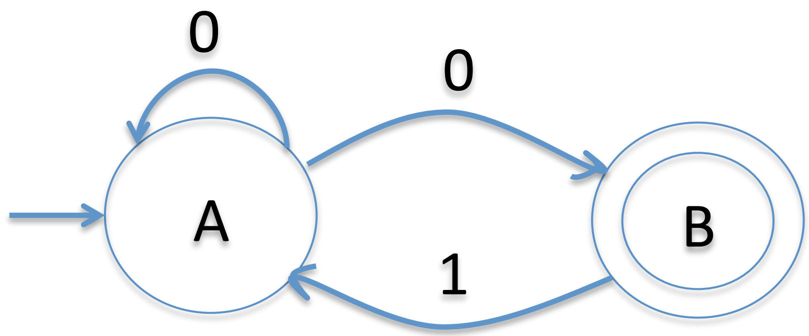
Örnek 2.1



- Durum çizeneği yandaki çizimde görülen sonlu özdevinirin tanıdığı kümeyi bir düzgün deyim olarak bulalım. Makinenin A ve B adlı iki durumu bulunduğu için iki değişkenli aşağıdaki denklem sistemi kurulur:

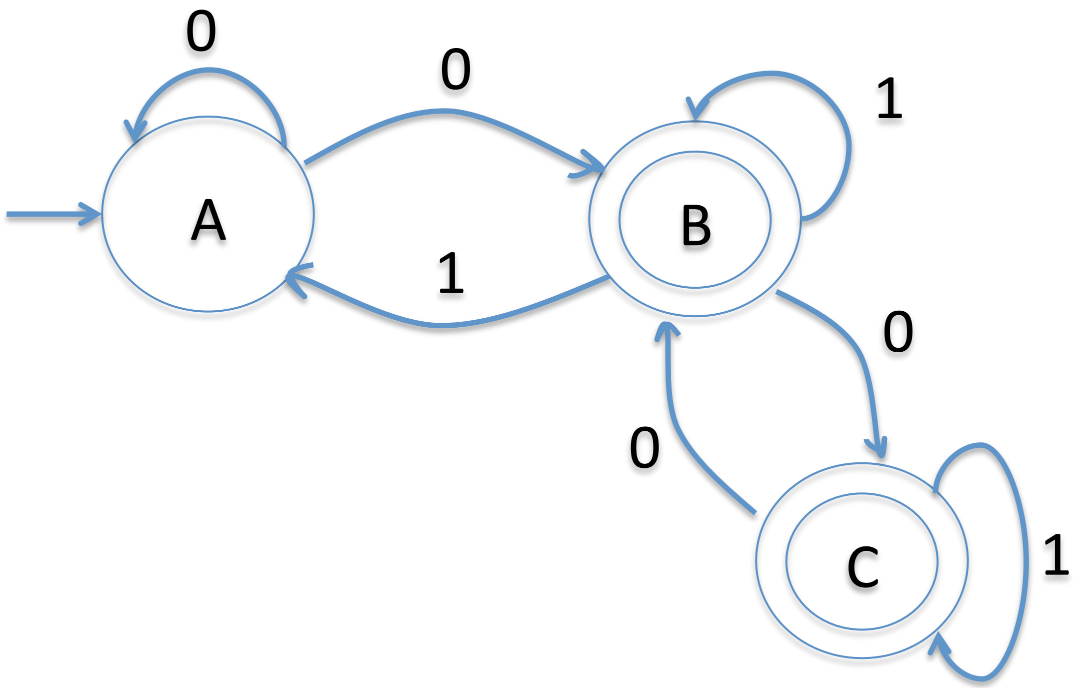
A = λ+ A0 + B1 ________________ (1)

B = A0 __________________________(2)



- Denklem (1)’de **B**’nin yerine denklem (2)’deki **A0** değeri konulduğunda:
- **A = λ + A0 + A01 = λ + A(0 + 01) _______**(3) elde edilir.
- Denklem (3), **teorem 2.3 (R=Q+RP ise denklemin tek çözümü: R=QP* dır.)** uygulanarak çözüldüğünde **A**’ya karşı gelen aşağıdaki düzgün deyim elde edilir (R-> A; Q= λ; P=0+01):
- **A = λ(0 + 01)* = (0 + 01)* ______________** (4)
- Denklem (2)’de **A**’nın yerine (4)’de elde edilen düzgün deyim konulduğunda, **B**’ye karşı gelen aşağıdaki düzgün deyim elde edilir:
- **B = A0 = (0 + 01)*0 _____________________** (5)
- Makinenin tek uç durumu **B** olduğu için, aşağıdaki düzgün deyim makinenin tanıdığı kümeye karşı gelir:
- **$T(M_{2.1}) = (0 + 01)^*0$**

Örnek 2.2



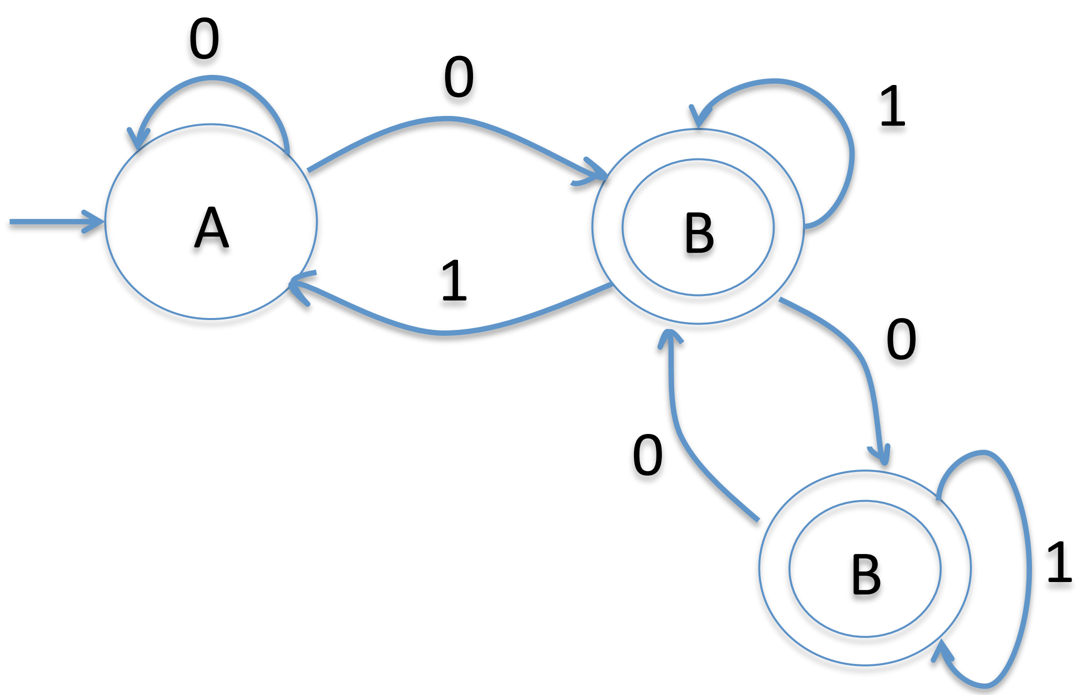
- Makinenin **A, B** ve **C** adlı üç durumu bulunduğu için üç değişkenli aşağıdaki denklem sistemi kurulur:
- **A = λ+ A0 + B1**__________________________ (1)
  B = A0 + B1 + C0_________________________ (2)
  C = B0 + C1 ______________________________(3)

- Makinenin **A, B** ve **C** adlı üç durumu bulunduğu için üç değişkenli aşağıdaki denklem sistemi kurulur:
- $\mathbf{A = \lambda + A0 + B1}$ ______________________ (1)
  $\mathbf{B = A0 + B1 + C0}$ ______________________ (2)
  $\mathbf{C = B0 + C1}$ __________________________(3)
- Üçüncü denkleme **Teorem 2.3** uygulandığında:
- $\mathbf{C = B01^*}$ ______________________________(4)
- elde edilir.
- İkinci denklemde **C**’nin yerine $\mathbf{B01^*}$ değeri konularak da:
- $\mathbf{B = A0 + B1 + B01^*0 = A0 + B(1 + 01^*0)}$ ___(5)
- elde edilir.
- **Teorem 2.3** kullanılarak yukarıdaki denklem çözüldüğünde ise:
- $\mathbf{B = A0(1 + 01^*0)^*}$ ____________________(6)
- elde edilir.

- **A = λ+ A0 + B1**____________________ (1)
  B = A0(1 + 01*0)* ___________________(6)
- Birinci deklemde **B**'nin yerine bulunan değer konulduğunda ise birinci denklem aşağıdaki biçime dönüşür:
- **A = λ + A0 + A0(1 + 01*0)*1 =λ + A[0 + 0(1 + 01*0)*1]** _________(7)
- Teorem 2.3 kullanılarak yukarıdaki denklem çözüldüğünde ise **A**'ya karşı gelen aşağıdaki düzgün deyim elde edilir:
- **A = λ[0 + 0(1 + 01*0)*1]*= [0 + 0(1 + 01*0)*1]***_________________(8)

- Denklem (6)'da **A**'nın yerine (8)'de bulunan değeri konularak **B**'ye karşı gelen aşağıdaki düzgün deyim elde edilir.
- **B = [0 + 0(1 + 01*0)*1]* 0(1 + 01*0)***________(9)
- Son olarak da denklem (4)'de **B**'nin yerine (9)'da bulunan değeri konularak **C**'ye karşı gelen düzgün deyim elde edilir.
- **C = [0 + 0(1 + 01*0)*1]* 0(1 + 01*0)* 01***

- $B = [0 + 0(1 + 01^*0)^*1]^* \, 0(1 + 01^*0)^*$
- $C = [0 + 0(1 + 01^*0)^*1]^* \, 0(1 + 01^*0)^* \, 01^*$
- Denklem sistemi çözülüp, her bir duruma karşı gelen düzgün deyim elde edildikten sonra, makinenin tanıdığı kümeye karşı gelen düzgün deyim aşağıdaki gibi elde edilir.
- $T(M_{2.2}) = B + C = B + B01^* = B(\lambda + 01^*)$
- $T(M_{2.2}) = [0 + 0(1 + 01^*0)^*1]^* \, 0(1 + 01^*0)^* \, (\lambda + 01^*)$



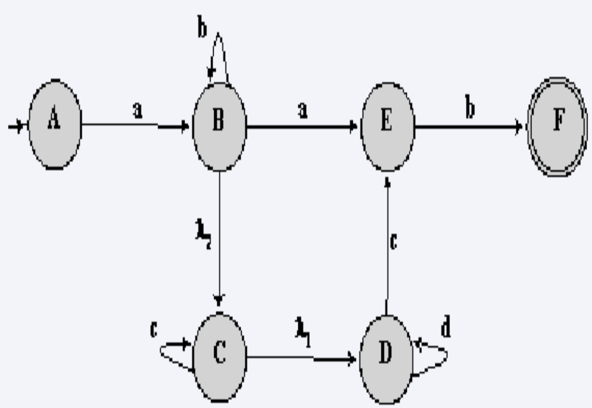
Örnek 1

L $_{2.1}$ = ab*(a + c*d*c)b

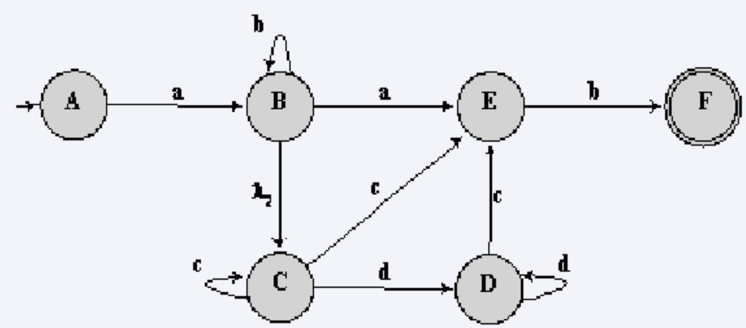
a) **L $_{2.1}$**'i tanıyan NFA'nın geçiş çizeneğini oluşturunuz. Oluşturduğunuz geçiş çizeneği λ-geçişleri içerebilir.

b) Eğer oluşturduğunuz geçiş çizeneği λ-geçişleri içeriyorsa, bu geçişleri tek tek yok ederek λ-geçişi içermeyen denk geçiş çizeneğini bulunuz.

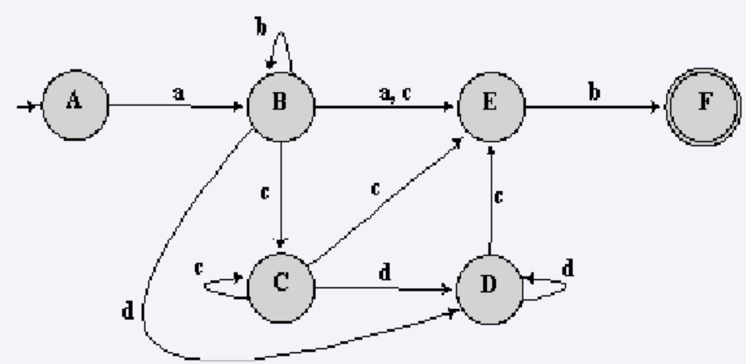
a)



b) λ_1 yok edildikten sonra:



λ_2 yok edildikten sonra:



Örnek 2

Sonlu durumlu **$M_{2.2}$** makinesi **{ 0, 1 }** alfabesinde **11, 101** ya da **110** ile biten dizgileri tanıyan makine olarak tanımlanıyor.

a) **$M_{2.2}$**'nin tanıdığı düzgün dili **($L_{2.2}$)** bir düzgün deyim olarak yazınız.

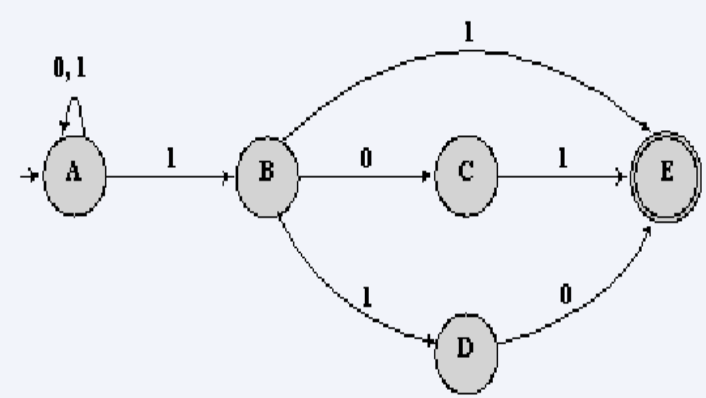
b) **$M_{2.2}$**'nin deterministik olmayan **(NFA)** geçiş çizeneğini olabildiğince az durum kullanarak oluşturunuz.

c) Durumları **S_0, S_1, S_2, ...** diye adlandırarak **$M_{2.2}$**'nin deterministik **(DFA)** geçiş çizelgesi ile geçiş çizeneğini oluşturunuz.

d) Makinenin denklik bölümlemesini bulunuz. Eğer oluşturduğunuz makinenin denk durumları varsa, eşdeğer en küçük makinenin durum çizelgesini bulunuz.

a) $L_{2.2} = (0 + 1)^* \, 1 \, (1 + 01 + 10)$

b) NFA'nın Geçiş Çizeneği:

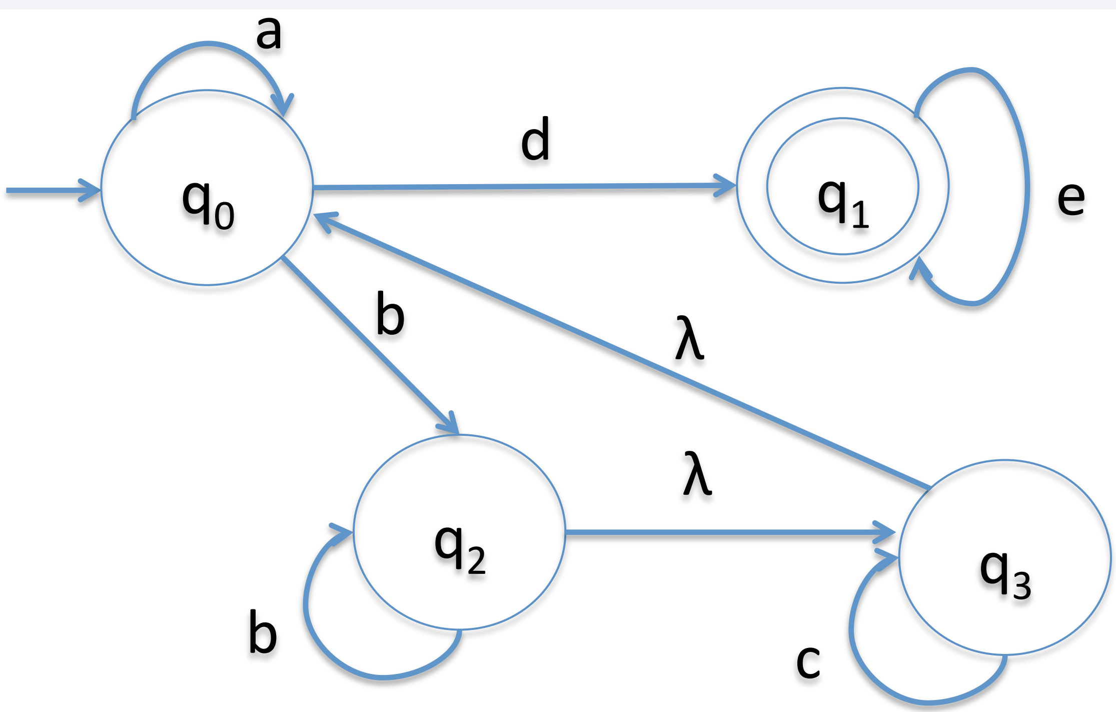


Örnek 3

L $_{2.3}$ dili aşağıdaki düzgün deyimle tanımlanıyor.
L $_{2.3}$ **= (a + bb*c*)*de***

a) L $_{2.3}$'ü tanıyan NFA'nın geçiş çizeneğini oluşturunuz. Oluşturduğunuz geçiş çizeneği (**G**$_0$ olarak adlandıralım) λ-geçişleri içerebilir.

b) Eğer **G**$_0$ λ–geçişleri içeriyorsa, λ–geçişi içermeyen eşdeğer geçiş çizeneğini bulmanız isteniyor. Bunun için **G**$_0$'da, eğer birden çok λ–geçişi varsa, bu geçişlerin hangi sırada yok edilmesi gerektiğini belirtiniz ve yok edilme sıralarına göre bu geçişleri **λ_1, λ_2, λ_3...** olarak adlandırız. Daha sonra λ_1'i yok ederek **G**$_1$'i; eğer varsa λ_2'yi yok ederek **G**$_2$'yi, ..., bulunuz.



Örnek 4

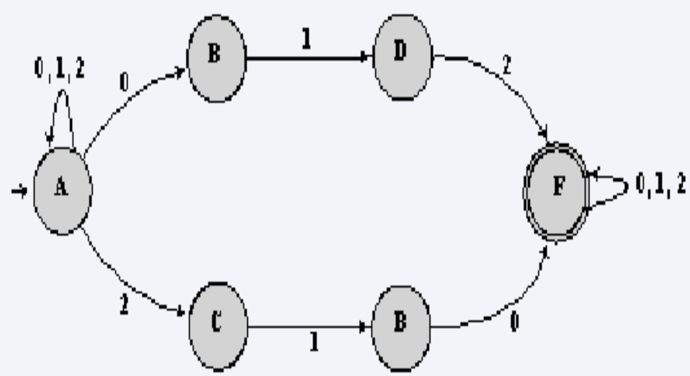
L $_{2.4}$ dili, **{ 0, 1, 2 }** alfabesinde, içinde **012** ya da **210** aldizgisi (ikisinden en az biri) bulunan dizgiler kümesi olarak tanımlanıyor. **L** $_{2.4}$'ü tanıyan, en az durumlu **DFA**'yı bulmanız isteniyor. Bunun için:

a) **L** $_{2.4}$'ü bir düzgün deyimle gösteriniz ve **L** $_{2.4}$'ü tanıyan **NFA**'nın geçiş çizeneğini oluşturunuz. Oluşturacağınız geçiş çizeneği λ-geçişleri içermesin.

b) Durumlarını **S_0, S_1, ...** diye adlandırarak **L** $_{2.4}$'ü tanıyan **DFA**'nın geçiş çizelgesini bulunuz.

c) Bulduğunuz **DFA**'yı ingirgeyerek, **L** $_{2.4}$'ü tanıyan en az durumlu **DFA**'nın geçiş çizelge ve çizeneğini (durumları **S, A, B, C, ...** diye adlandırarak) bulunuz.

a) $L_{2.4} = (0 + 1 + 2)^* (012 + 210) (0 + 1 + 2)^*$



Örnek 5

Aşağıda sözlü olarak tanımlanan kümelerden her birinin biçimsel tanımını bir düzgün deyimle veriniz.

a) { a, b, c } alfabesinde, içindeki her **a**'dan önce ve her **b**'den sonra en az bir **c** bulunan dizgiler kümesi.

b) { a, b, c } alfabesinde, içindeki **b**'lerin sayısı ile **c**'lerin sayısının toplamı 3 olan dizgiler kümesi.

c) { a, b, c } alfabesinde, içindeki **a**'ların sayısı 3 olan dizgiler kümesi.

d) { a, b, c } alfabesinde, içinde **aa** altdizgisi bulunmayan dizgiler kümesi.

e) { a, b } alfabesinde, içinde **aaa** altdizgisi bulunmayan dizgiler kümesi.

f) { a, b, c } alfabesinde, içindeki **a**'ların sayısı ikinin katı (2, 4, 6, ...) olan dizgiler kümesi.

a) $P_1 = (ca + bc + c)^*$

b) $P_2 = a^*(b + c)a^*(b + c)a^*(b + c)a^*$

c) $P_3 = (b + c)^*a(b + c)^*a(b + c)^*a(b + c)^*$

d) $P_4 = (b + c + ab + ac)^*(a + \lambda)$

e) $P_5 = (b + ab + aab)^*(a + aa + \lambda)$

f) $P_6 = [\, (b + c)^*a(b + c)^*a(b + c)^*]^*(b + c)^*a(b + c)^*a\,(b + c)^*$

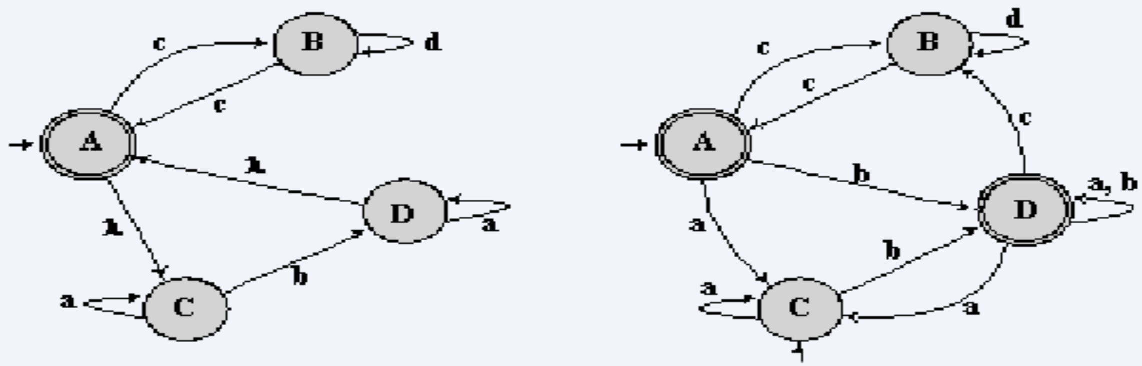
Örnek 6

Aşağıdaki düzgün deyimlerle tanımlanan her küme için, kümeyi tanıyan **NFA**'nın geçiş çizeneğini oluşturunuz. Eğer oluşturduğunuz çizenek λ-geçişleri içeriyorsa, λ-geçişsiz denk geçiş çizeneğini bulunuz.

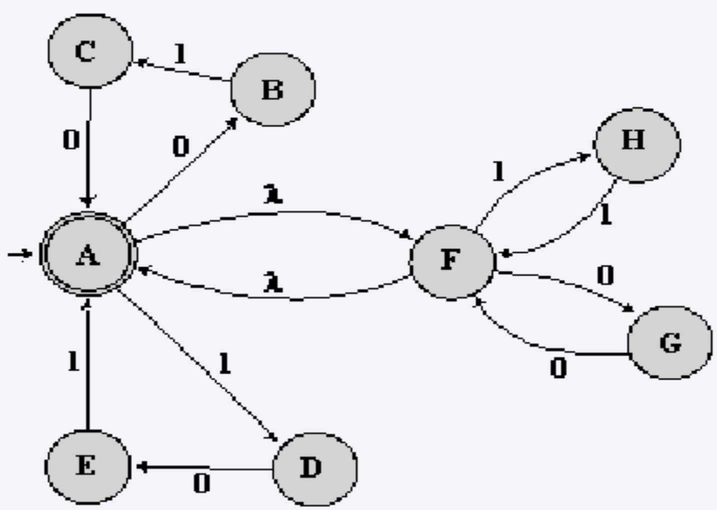
a) $P_1 = (a^*ba^* + cd^*c)^*$

b) $P_2 = [010 + 101 + (00 + 11)^*]^*$

a)



b)



Örnek 7

$L_{2.7}$ dili, **{ 0, 1 }** alfabesinde, içinde **0100** ya da **1011** altdizgisi (en az bir kez) bulunan dizgiler kümesi olarak tanımlanıyor.

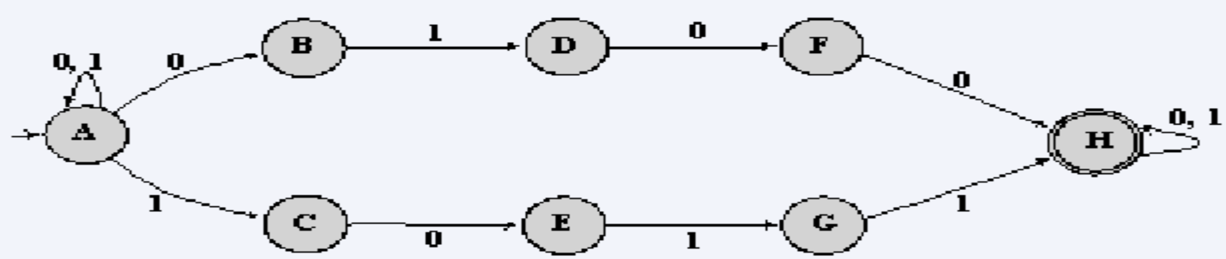
a) $L_{2.7}$ dilini bir düzgün deyimle gösteriniz.

b) $L_{2.7}$ dilini tanıyan **NFA**'nın geçiş çizeneğini oluşturunuz. Geçiş çizeneğinde λ–geçişleri yer almasın. Deterministik olmayan bu çizenekte durumları **A, B, C, ..** diye adlandırınız.

c) Bulduğunuz **NFA**'ya denk **DFA'**nın geçiş çizeneğini oluşturunuz. Deterministik çizenekte durumları $S_0, S_1, S_2, \ldots$ diye adlandırınız. Deterministik çizeneği indirgeyiniz ve en az durumlu deterministik çizeneği bulunuz.

a) $L_{2.7} = (0 + 1)^*(0100 + 1011)(0 + 1)^*$

b)



Örnek 8

Aşağıdaki kümelerden her birini tanımlayan bir düzgün deyim yazınız.

a) { a, b, c } alfabesinde, içinde iki (ve yalnız iki) tane **a** bulunan dizgiler kümesi.

b) { a, b } alfabesinde, içindeki **a**'ların sayısı üçün katı (3, 6, 9, …) olan dizgiler kümesi.

c) { a, b } alfabesinde, içinde **aa** altdizgisi bulunmayan dizgiler kümesi.

Durumların anlamları

a) $P_1 = (b + c)^*a(b + c)^*a(b + c)^*$

Durumların anlamları

b) $P_2 = (b^*ab^*ab^*ab^*)^*b^*ab^*ab^*ab^*$

Durumların anlamları

c) $P_3 = (b + ab)^*(a + \lambda)$

Örnek 9

L $_{2.9}$ = (a*bc + ab*c + abc*)*

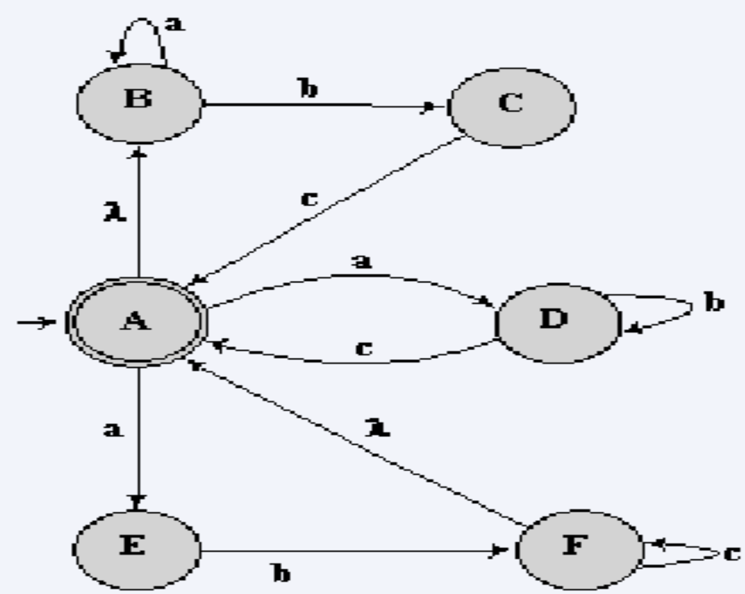
Yukarıdaki düzgün deyimle tanımlanan **L $_{2.9}$** dilini tanıyan bir DFA bulmanız isteniyor. Bunun için:

a) Durumları **A, B, C, ….** diye adlandırarak **L $_{2.9}$**'u tanıyan NFA'nın geçiş çizeneğini oluşturunuz.

b) Oluşturduğunuz çizenekte λ-geçişleri varsa, bu geçişleri yok ederek λ–geçişsiz denk çizeneği bulunuz.

c) Durumları **q_0, q_1, q_2, …** diye adlandırarak denk DFA'nın geçiş çizelgesini oluşturunuz.

a)



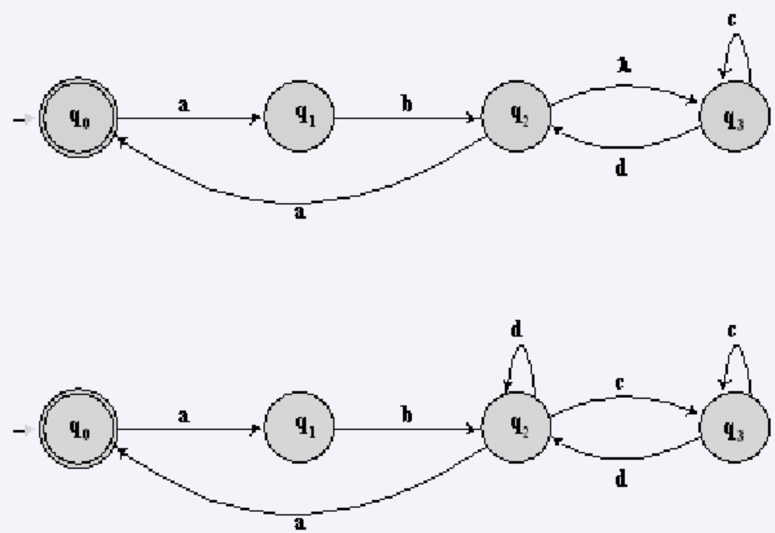
Örnek 10

Aşağıdaki düzgün deyimlerle tanımlanan her küme için, kümeyi tanıyan sonlu özdevinirin (NFA) geçiş çizeneğini oluşturunuz. Oluşturduğunuz çizenekte λ-geçişleri varsa, bu λ–geçişleri yok ederek λ-geçişsiz denk çizeneği bulunuz.

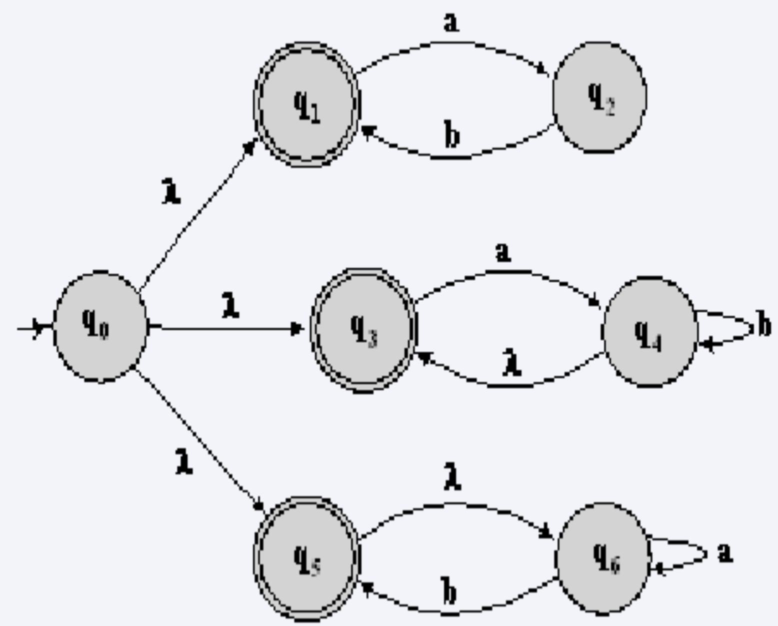
a) $L_{2.10.1} = (ab(c^*d)^*a)^*$

b) $L_{2.10.2} = (a\,b)^* + (ab^*)^* + (a^*b)^*$

a)



b)

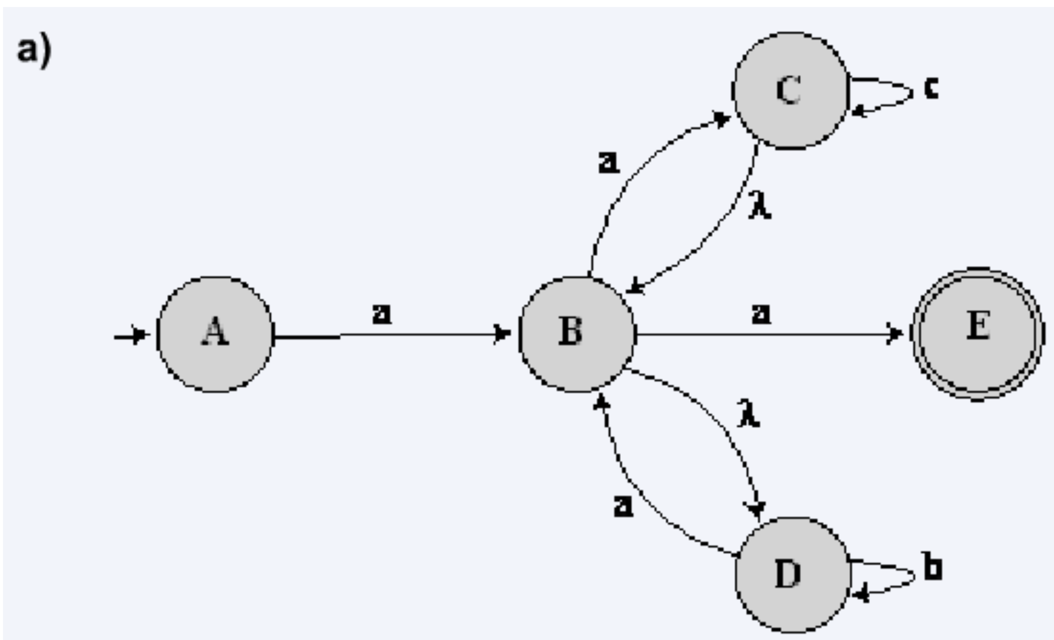


Örnek 11

$L_{2.11} = a(ac^* + b^*a)^*a$

Yukarıdaki düzgün deyimle tanımlanan $L_{2.11}$ dilini tanıyan bir DFA bulmanız isteniyor. Bunun için:

- **a)** Durumları **A, B, C, ….** diye adlandırarak $L_{2.11}$'i tanıyan NFA'nın geçiş çizeneğini oluşturunuz.
- **b)** Oluşturduğunuz çizenekte λ-geçişleri varsa, bu geçişleri yok ederek λ-geçişsiz denk çizeneği bulunuz.
- **c)** Durumları q_0, q_1, q_2, … diye adlandırarak denk DFA'nın geçiş çizelgesini oluşturunuz.



Örnek 12

$L_{2.12} = \{ (a + bb)^k (c + dd)^m (e + ff)^n \mid k, n \geq 0, \; m \geq 1 \}$

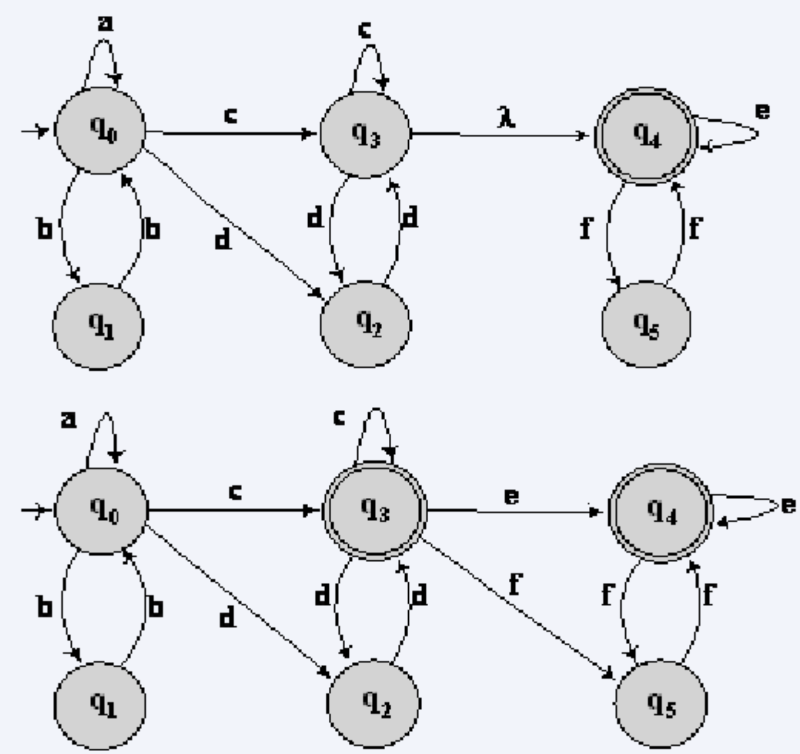
a) Yukarıda küme tanımı verilen düzgün dili bir düzgün deyimle gösteriniz.

b) Dili tanıyan NFA'nın geçiş çizeneğini oluşturunuz.

c) Oluşturduğunuz NFA'ya denk DFA'nın geçiş çizelgesini oluşturunuz.

a) $L_{2.12} = (a + bb)^*(c + dd)(c + dd)^*(e + ff)^*$

b)



Örnek 13

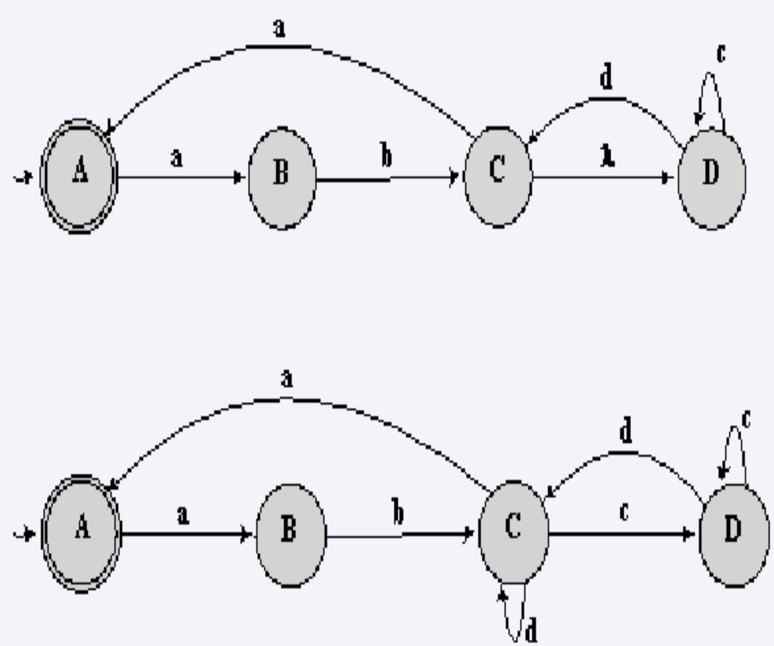
L $_{2.13}$ dili aşağıdaki düzgün deyimle tanımlanıyor.

L $_{2.13}$ **= (ab(c*d)* a)***

a) L $_{2.13}$'ü tanıyan bir NFA'nın geçiş çizeneğini oluşturunuz. Oluşturduğunuz çizenek λ-geçişi/geçişleri içeriyorsa, bu geçişleri yok edip λ-geçişsiz denk çizeneği bulunuz (bu çizenekte durumları **A, B, C, ...** diye adlandırınız).

b) Yukarıda bulduğunuz NFA'ya denk DFA'nın geçiş çizeneğini bulunuz. Deterministik çizenekte durumları **S_0, S_1, S_2, ...** diye adlandırınız. Bu son çizenek indirgenebilir mi? Evet ise çizeneği indirgeyerek en az durumlu denk çizeneği bulunuz.

a)



Örnek 14

L $_{2.14}$ dili **{ a, b }** alfabesinde **a** ile başlayıp **a** ile biten, ilk ve son simgeler arasında da **aa** altdizgisini içeren (dolayısıyla uzunluğu en az 4 olan) dizgiler kümesi olarak tanımlanıyor. **L** $_{2.16}$'da yer alan dizgilerden kimi örnekler aşağıda görülmektedir.

L $_{2.14}$ **= { aaaa, abaaa, aaaaa, ababbaaba, … }**

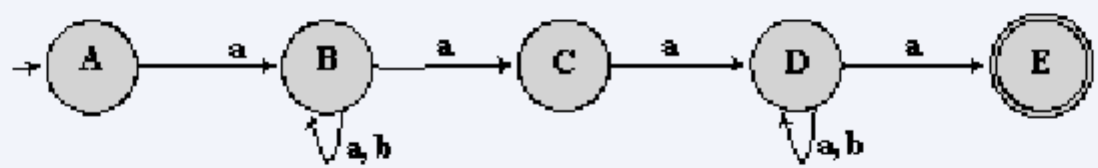
a) L $_{2.14}$ dilini tanımlayan bir düzgün deyim yazınız.

b) L $_{2.14}$ dilini tanıyan NFA'nın geçiş çizeneğini oluşturunuz. Eğer geçiş çizeneği λ-geçişleri içeriyorsa, bu geçişleri yok ederek λ–geçişsiz denk çizeneği bulunuz.

c) Bulduğunuz NFA'ya denk DFA'nın geçiş çizeneğini oluşturunuz.

a) L $_{2.14}$ **= a(a + b)*aa(a + b)*a**

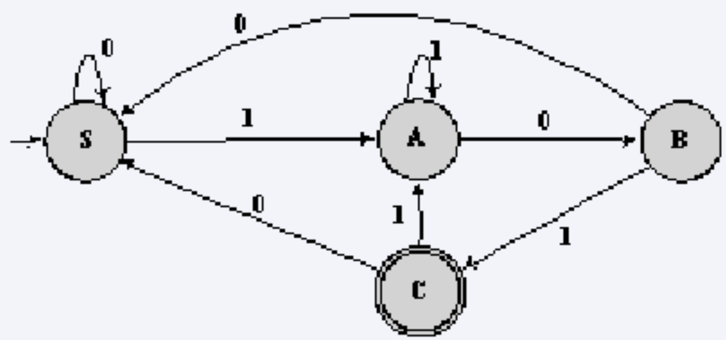
b)



Örnek 15

Yandaki geçiş çizelgesi ile tanımlanan sonlu özdevinirin tanıdığı dili bir düzgün deyimle gösteriniz.

	0	1
→S	S	A
A	B	A
B	S	C
(C)	S	A



Denklem Sistemi:

$S = \lambda + S0 + B0 + C0$

$A = S1 + A1 + C1$

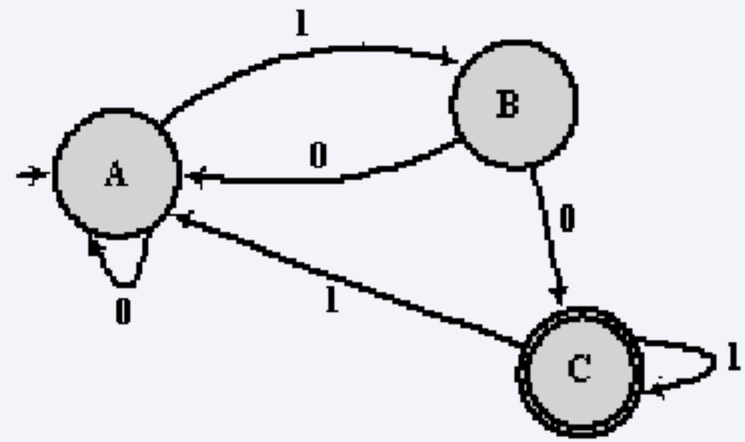
$B = A0$

$C = B1$

$L_{2.15} = [0 + 1(1 + 011)^*(00 + 010)]^*1(1 + 011)^*01$

Örnek 16

Aşağıda geçiş çizeneği verilen NFA'nın tanıdığı dili bir düzgün deyim olarak yazınız. NFA'ya denk bir DFA bulunuz. DFA'nın durumlarını **S_0, S_1, S_2, …** diye adlandırarak geçiş çizelgesi ve geçiş çizeneğini oluşturunuz.



Denklem Sistemi:

$A = \lambda + A0 + B0 + C1$

$B = A1$

$C = B0 + C1$

$L_{2.16} = (0 + 10 + 101^*1)^*101^*$

DFA'nın Geçiş Çizelge ve Çizeneği:

	0	1
→ S_0	S_0	S_1
S_1	S_2	S_3
S_2	S_0	S_4
S_3	S_3	S_3
S_4	S_2	S_4

Durumların karşılıkları:

S_0 : A

S_1 : B

S_2 : AC

S_3 : Φ

S_4 : ABC