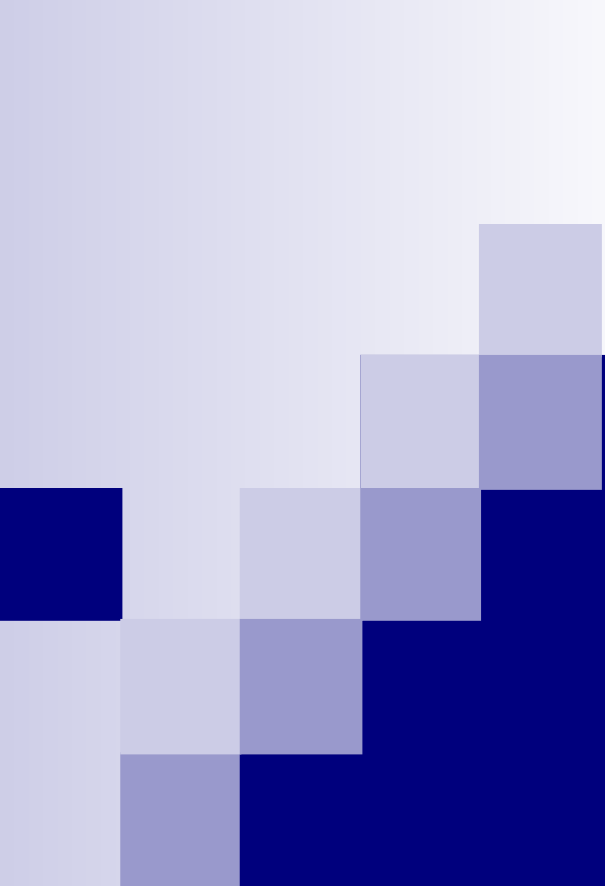




שפות פורמאליות אוטומטים

תורת הקומפילציה
אהרון נץ

מבוסס על השקפים של עומר ביהם
שמבוססים על שקפי הרצאה מהקורס אוטומטים ושפות פורמאליות
בטכניון, פרופ' שמואל זקס

הנושאים שנעבור

- שפות פורמאליות
- מכונות/אוטומטים
- דקדוקים

הגדרות - אלף בית, אות

■ **אלף בית Σ** , הינו קבוצה סופית כלשהי.
■ **דוגמאות:**

■ $\Sigma_1 = \{a,b,c\}$

■ $\Sigma_2 = \{0,1\}$

■ $\Sigma_3 = \{(x,y) \mid x \in \{0,1\}, y \in \{a,b,c\}\}$

■ **אות $\sigma \in \Sigma$** , הינה איבר מתוך א"ב נתון.

הגדרות - מילה

- מילה מעל א"ב Σ הינה סידרה סופית של אותיות.
- סימון מקובל w
- דוגמאות:
- aaa היא מילה מעל $\{a\}$
- $1313d3\$\$ \#$ היא מילה מעל $\{d,1,3,\$, \#, \%, \&\}$
- $(2,c)(1,b)(3,a)$ היא מילה מעל $\{(x,y) \mid x \in \{1,2,3\}, y \in \{a,b,c\}\}$

הגדרות - מילה

- תהי $w = s_1, s_2, \dots, s_n$. נסמן את אורכה ב- $|w|$, כלומר $|w| = n$
- מילה שאינה מכילה שום אות נקראת המילה הריקה
- נסמן ב- ε את המילה הריקה. $|\varepsilon| = 0$

שרשור מילים

■ בהנתן שתי מילים $w_1 = a_1 a_2 \dots a_n$, $w_2 = b_1 b_2 \dots b_m$

■ שרשור מילים (concatenation) מוגדר כהדבקת המילה השנייה למילה הראשונה זו לזו (חשיבות לסדר)

$$w_1 \cdot w_2 = a_1 a_2 \dots a_n b_1 b_2 \dots b_m \quad \blacksquare$$

□ שרשור המילים "פר" ו "פר" נותן "פרפר"

□ באותו אופן, שרשור המילים "אינ" ו "טל" נותן "אינטל"

חזקה והיפוך

■ חזקה של מילה w

- $w^i = \varepsilon$, if $i = 0$
- $w^i = w^{i-1} \cdot w$, otherwise

■ יותר אינטואיטיבי

- $w^n = ww...w$ (n times)

■ היפוך של מילה w מסומן w^R

■ אם $w = a_1a_2...a_n$ אז $w^R = a_na_{n-1}...a_1$

שפות פורמאליות

■ שפה פורמאלית L , מעל Σ היא קבוצת מילים מעל אותו Σ .

■ דוגמאות לשפות מעל $\Sigma = \{a, b\}$

- $\{a\}$, $\{aaab, b\}$
- $\{w \mid \#a(w) = \#b(w)\}$, $\{a^n b^n \mid n \in \mathbb{N}\}$
- $\{\}$, $\{\varepsilon\}$, $\emptyset$

פעולות על שפות

■ שפה הופכית L^R

■ $L^R = \{w^R | w \in L\}$

■ שרשור של שפות $L_1 \cdot L_2$

■ $L_1 \cdot L_2 = \{w \cdot v | w \in L_1, v \in L_2\}$

■ חזקה של שפה L^i

■ $L^i = \{\varepsilon\}$, if $i = 0$
■ $L^i = L^{i-1} \cdot L$, otherwise

פעולות על שפות

- איחוד של שפות L_1, L_2 מעל א"ב Σ
- $L_1 \cup L_2 = \{w | w \in L_1 \cup w \in L_2\}$
- סגור קליין של שפה L , המסומן L^* , מוגדר כשרשור של 0 או יותר עותקים של L
- L^+ , מוגדר כשרשור של 1 או יותר עותקים של L
- את שפת המילים מעל א"ב Σ מסמנים Σ^*

אוטומטים

■ אוטומט – אבסטרקציה למערכת המגיבה לקלטים.

■ קלט סופי, בקרה סופית, מעבר יחיד.

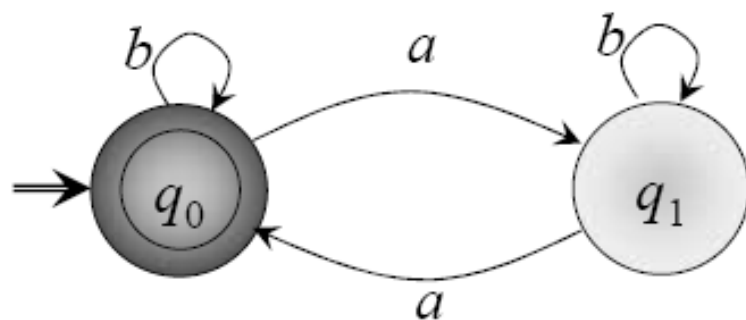
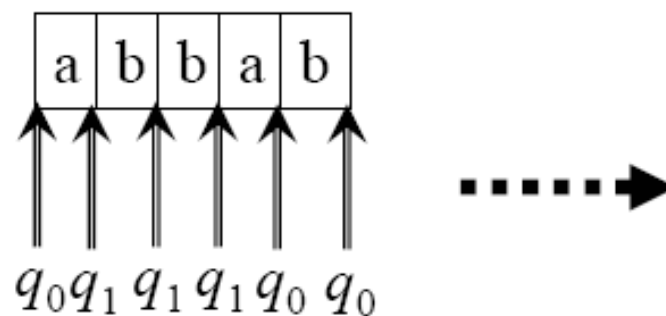
■ אופן הפעולה:

□ בכל שלב האוטומט נמצא באחד המצבים (התחלה מצב התחלתי)

□ עם קריאת אות קלט, ה"ראש הקורא" מתקדם לאות הקלט הבאה, והאוטומט משנה את מצבו כתלות (פונקציה של) במצב הנוכחי, ואות הקלט

□ פלט האוטומט מוגדר ע"י מצב האוטומט בסיום קריאת הקלט

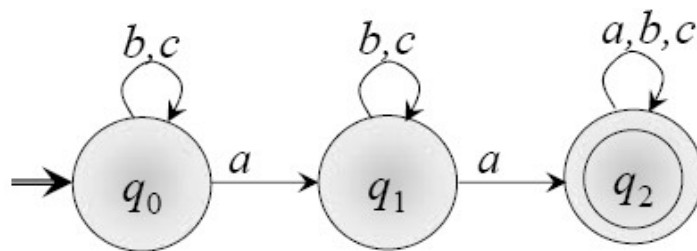
דוגמא



אינטואיציה - המשך

- אוטומט סופי הינו מודל מופשט המגיב לקלטים.
- תגובתו לקלטים (הפלט) מוגדר ע"י סוג המצב אליו מגיע האוטומט בסיום קריאת הקלט.
- אנו מגדירים שני סוגי מצבים:
 - מצבים מקבלים.
 - מצבים לא מקבלים.
- מכאן שאוטומט יכול להגיב באחת משתי צורות:
 - לקבל מילה.
 - לדחות מילה.
- תגובתו של אוטומט לאות קלט הינה פונקציה של המצב הנוכחי ואות הקלט הנוכחית. פונקציה זו קובעת מצב יחיד אליו האוטומט הדטרמיניסטי יעבור.

אוטומט סופי - הצגה ע"י טבלת מעברים



הטבלה מגדירה פונקציה

$$\delta : Q \times \Sigma \rightarrow Q$$

אשר בהנתן קלט של זוג (מצב ומילה)
מחזירה מצב חדש.

המצב החדש לאו דווקא שונה!

c	b	a	
Q_0	Q_0	Q_1	Q_0
Q_1	Q_1	Q_2	Q_1
Q_2	Q_2	Q_2	Q_2

אוטומט סופי דטרמיניסטי - הגדרה פורמאלית

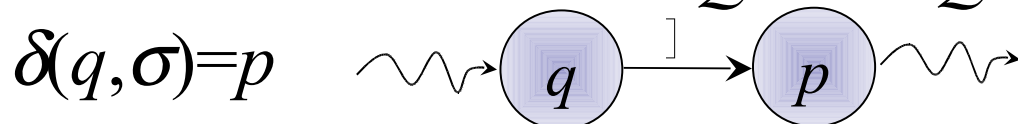
■ אוטומט סופי דטרמיניסטי הינו החמישייה $A=(Q,\Sigma,q_0,\delta,F)$

■ $Q=\{q_0, q_1, q_2, \dots\}$, קבוצה סופית של מצבים

■ $\Sigma=\{a,b,\dots\}$, א"ב הקלט (קבוצה סופית של סימנים)

■ q_0 , המצב ההתחלתי היחיד

■ δ , פונקצית המעברים: $\delta : Q \times \Sigma \rightarrow Q$



■ באופן כללי, $\delta(q,w)=p$, כאשר w היא מילה, q המצב לפני

קריאת המילה, ו- p המצב בסיום קריאת המילה ע"י האוטומט

■ F - קבוצת מצבים סופיים/מקבלים $F=\{q_0, q_3, \dots\} \subseteq Q$

השפה המתקבלת ע"י אוטומט

■ בהינתן אוטומט סופי דטרמיניסטי A , שפת האוטומט $L(A)$ היא קבוצת המילים המתקבלת על ידו

$$L(A) := \{w \in \Sigma^* \mid \delta(q_0, w) \in F\} \quad \blacksquare$$

- בכל שלב האוטומט נמצא באחד המצבים (התחלה מצב התחלתי)
- עם קריאת אות קלט, ה"ראש הקורא" מתקדם לאות הקלט הבאה, והאוטומט משנה את מצבו כתלות (פונקציה של) במצב הנוכחי, ואות הקלט
- פלט האוטומט מוגדר ע"י מצב האוטומט בסיום קריאת הקלט.

השפה המתקבלת ע"י אוטומט - הוכחה

■ ע"מ להוכיח כי אוטומט A מקבל שפה L , יש להוכיח כי

$$L(A) = L$$

■ עושים זאת בדרך כלל באינדוקציה באופן הבא:

■ כיוון א' של ההכלה $L(A) \subseteq L$

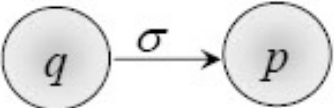
□ אינדוקציה על מספר צעדי החישוב של האוטומט

■ כיוון ב' של ההכלה $L \subseteq L(A)$

□ אינדוקציה על אורך המילה (או מבנה המילה)

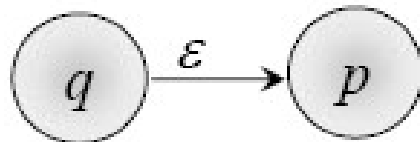
אי דטרמיניזם

■ עד כה, עבור מילה נתונה, מעבר האוטומט בתגובה

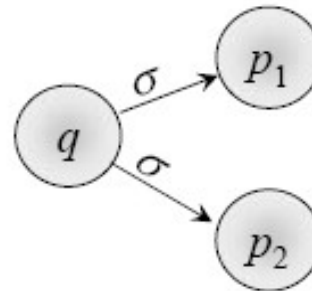
לקריאת אות היה מוגדר באופן חד-ערכי 

■ ניתן להגדיר את פעולת האוטומט כאי-דטרמיניסטית

כלומר בתגובה לקריאת אות, או אפילו בלי - מעבר ε ,
יתכנו מספר מעברים (כולל 0)



דוגמא למסע ε



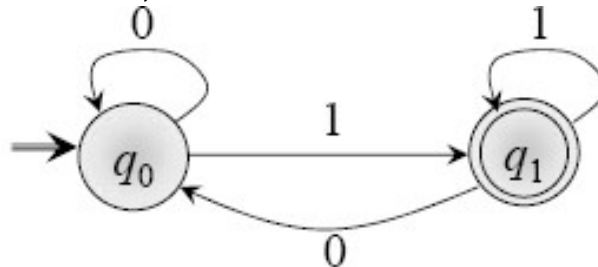
אי דטרמיניזם

- הכוונה במסע ε היא לכך שהאוטומט עובר ממצב q ל p ללא קריאת אף אות קלט.
- ניתן להסתכל על מסעי ε כעל מעבר האוטומט בין האותיות, לצורך התארגנות מחודשת
- אנו נאמר שמילה מתקבלת ע"י אוטומט אי-דטרמיניסטי אם "מ קיים עבודה חישוב המוביל למצב מקבל."

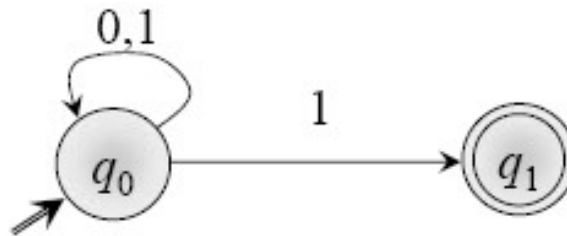
אי דטרמיניזם - דוגמא

■ נתבונן בשפה $L = \{w1 | w \in \{0,1\}^*\}$

□ להלן אוטומט סופי דטרמיניסטי המקבל אותה



□ להלן אוטומט סופי אי-דטרמיניסטי (לאותה השפה)



אוטומט סופי דטרמיניסטי - הגדרה פורמאלית

■ אוטומט סופי דטרמיניסטי הינו החמישייה $A=(Q,\Sigma,q_0,\delta,F)$

■ $Q=\{q_0, q_1, q_2, \dots\}$, קבוצה סופית של מצבים

■ $\Sigma=\{a,b,\dots\}$, א"ב הקלט (קבוצה סופית של סימנים)

■ q_0 , המצב ההתחלתי היחיד

השינוי היחיד לעומת
האוטומט הדטרמיניסטי

■ δ , פונקצית המעברים: $\delta : Q \times \Sigma \rightarrow 2^Q$

באופן כללי, $\delta(q,w)=P$, כאשר w היא מילה, q המצב לפני קריאת המילה, ו- P קבוצת המצבים האפשריים בסיום קריאת המילה ע"י האוטומט

■ F - קבוצת מצבים סופיים/מקבלים $F=\{q_0, q_3, \dots\} \subseteq Q$

שקילות המודלים

- שאלה מרכזית שמעניינת אותנו בעת הצגת מודל חישובי חדש היא מה כוחו ביחס למודלים שאנו מכירים.
- משמעות השאלה: האם קיימות שפות שמודל אחד יכול לקבל והשני לא ?
- מסתבר כי שני השינויים שהוצגו, אי-דטרמיניזם ומסעי ε אינם משנים את הכוח החישובי של האוטומט

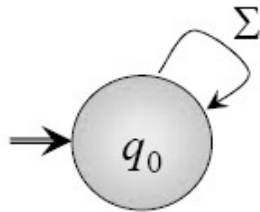
שפות רגולריות

- ראינו כי אוטומט סופי דטרמיניסטי מזהה (מקבל) שפות מעל א"ב נתון
- חלק משאלות בהן נתעניין
 - האם קיימת שפה שאס"ד אינו מסוגל לזהותה?
 - אילו תכונות יש לקבוצת השפות אותן יכול אס"ד לקבל?
 - אילו מודלים נוספים קיימים המקבלים את אותה קבוצת שפות?

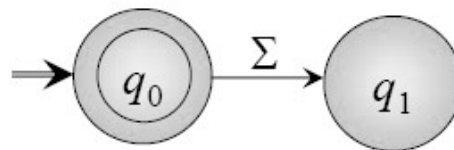
שפות רגולריות

■ שפה L נקראת רגולרית אם "מ קיים עבודה אס"ד A , כך
ש $L(A) = L$

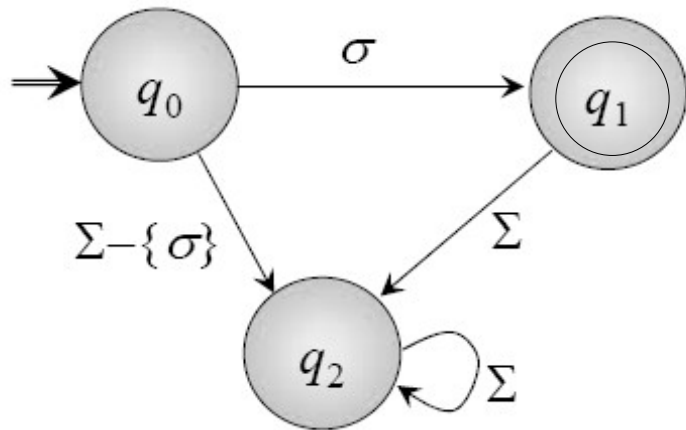
■ השפה הריקה, $\emptyset$, היא רגולרית.



■ השפה המכילה את המילה הריקה, $\{\epsilon\}$, היא רגולרית.

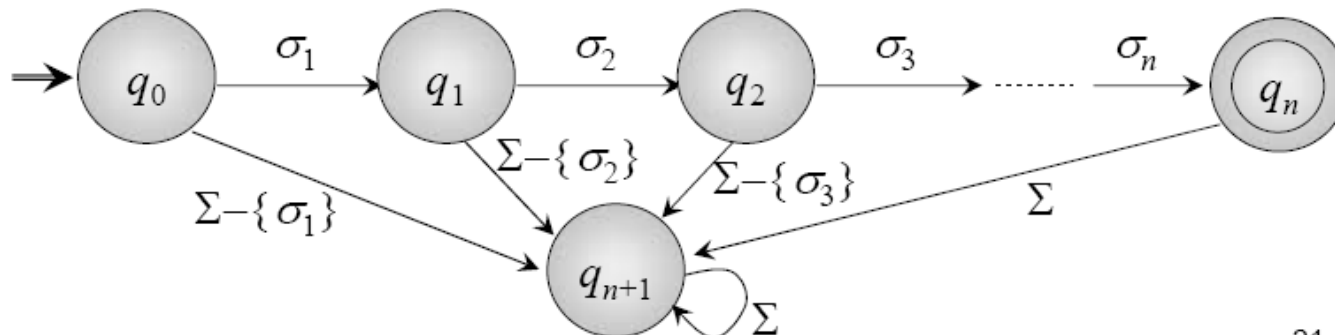


שפות רגולריות - דוגמאות



■ לכל $\sigma \in \Sigma$ השפה $\{\sigma\}$ רגולרית.

■ לכל $w \in \Sigma^*$ השפה $\{w\}$ רגולרית



21

סגירות השפות הרגולריות לפעולות

- השפות הרגולריות סגורות תחת חיתוך.
- השפות הרגולריות סגורות תחת השלמה.
- השפות הרגולריות סגורות תחת איחוד.
- כל שפה סופית היא רגולרית.
- השפות הרגולריות **אינן** סגורות תחת אינסופי.
- השפות הרגולריות **אינן** סגורות תחת חיתוך אינסופי.

ביטויים רגולרים

- השפות הרגולריות ניתנות לתיאור ע"י בניית אוטומט, וכן ע"י סימוני קבוצות רגילים כלומר תיאור תכונות שונות של המילים השייכות לקבוצה
- באמצעות תיאור כקבוצה ניתן כמובן לתאר גם שפות שאינן רגולריות, לדוגמא: $L = \{a^i b^i | i \geq 0\}$
- הוכחת רגולריות של שפה דורשת מציאת אוטומט המקבל אותה, או שימוש בתכונות סגירות
- נציג עתה דרך נוספת לתיאור שפות רגולריות. דרך סימון זו מבטיחה כי כל שפה הניתנת לתיאור באמצעותה היא רגולרית

ביטויים רגולרים

- ההגדרה הבאה היא תחבירית בלבד, אין לה משמעות
- אוסף הביטויים הרגולרים (מעל א"ב, Σ) המסומן R , מוגדר באינדוקציית מבנה באופן הבא:
- אטומים:

- $\emptyset, \varepsilon \in R$
- $\forall \sigma \in \Sigma, \sigma \in R$

■ פעולות יצירה:

- $r_1, r_2 \in R \Rightarrow (r_1 + r_2) \in R \text{ and } (r_1 \cdot r_2) \in R$
- $r \in R \Rightarrow (r^*) \in R$

ביטויים רגולריים

- לכל ביטוי רגולרי r נתאים משמעות: שפה $L[r]$
- $L[\emptyset] = \emptyset, L[\varepsilon] = \{\varepsilon\}, \forall \sigma \in \Sigma, L[\sigma] = \{\sigma\}$
- $r_1, r_2 \in R$
 - $L[(r_1 + r_2)] = L[r_1] \cup L[r_2]$
 - $L[(r_1 \cdot r_2)] = L[r_1] \cdot L[r_2]$
- $r \in R$
 - $L[(r)^*] = (L[r])^*$

- דוגמא: שפת המילים הזוגיות מעל $\{a,b\}$ מתאימה לביטוי הרגולרי $((a+b) \cdot (a+b))^*$

שקילות ביטויים רגולריים לאוטומטים

■ נזכור כי המוטיבציה להגדרת ביטויים רגולריים הייתה לתת צורת ייצוג (נוספת) לשפות רגולריות.

■ שני המשפטים הבאים מתקיימים:

□ לכל $r \in R$ מתקיים כי $L(R)$ היא שפה רגולרית

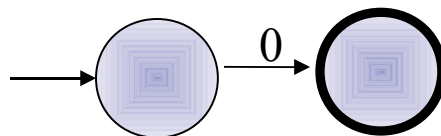
□ לכל שפה רגולרית L קיים ביטוי רגולרי r כך ש $L[r] = L$

בניית אוטומט מביטוי רגולרי

■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$r=0$

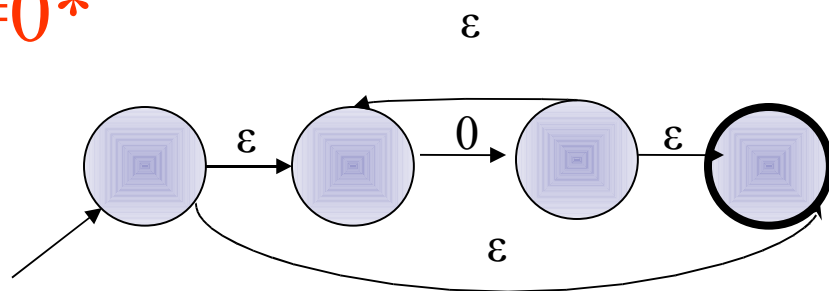


בניית אוטומט מביטוי רגולרי

■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$r=0^*$

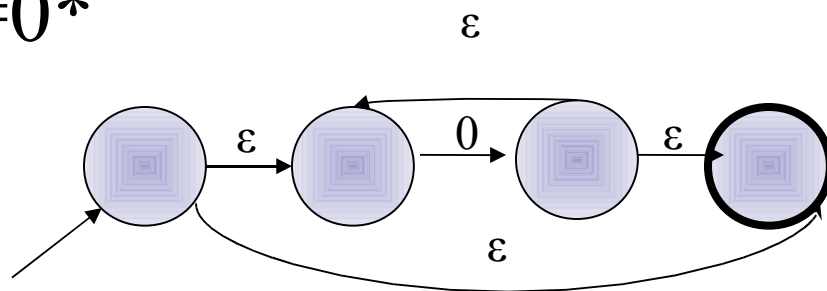


בניית אוטומט מביטוי רגולרי

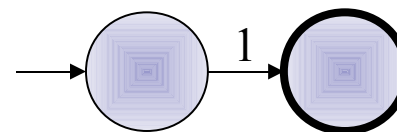
■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$r=0^*$



$r=1$

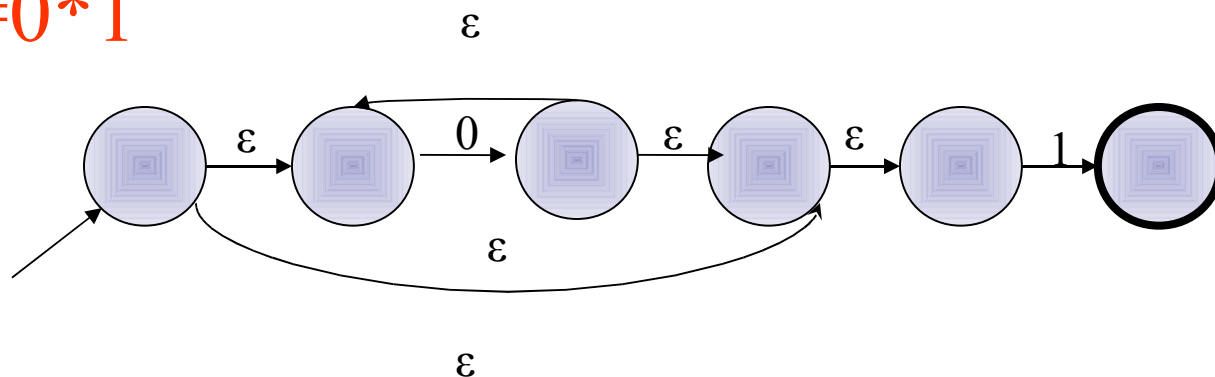


בניית אוטומט מביטוי רגולרי

■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$r=0^*1$

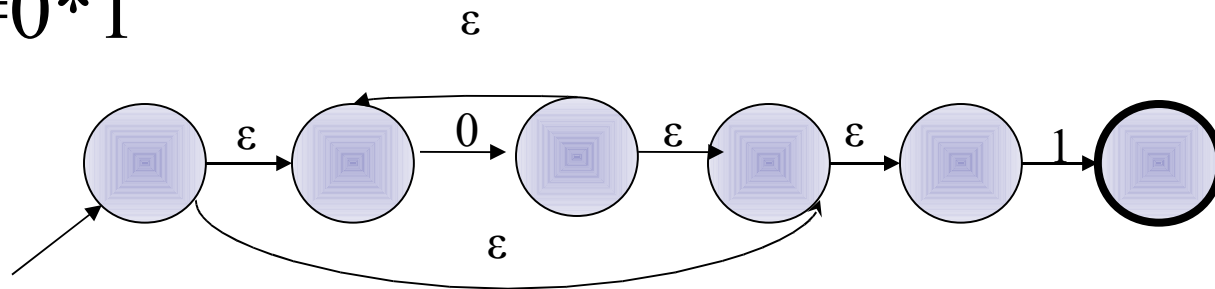


בניית אוטומט מביטוי רגולרי

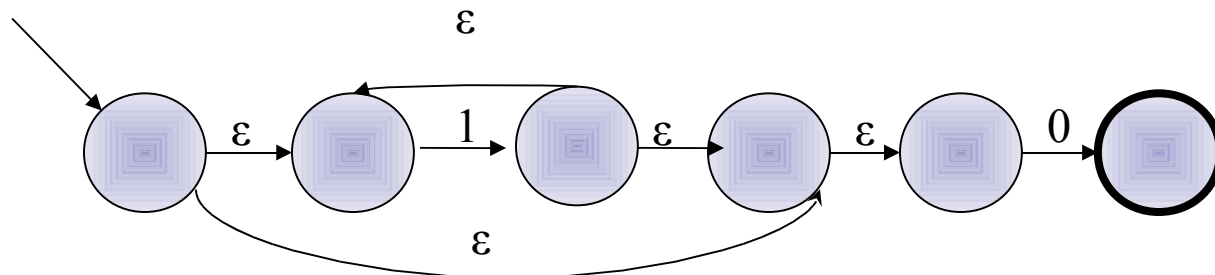
■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$r=0^*1$



$r=1^*0$

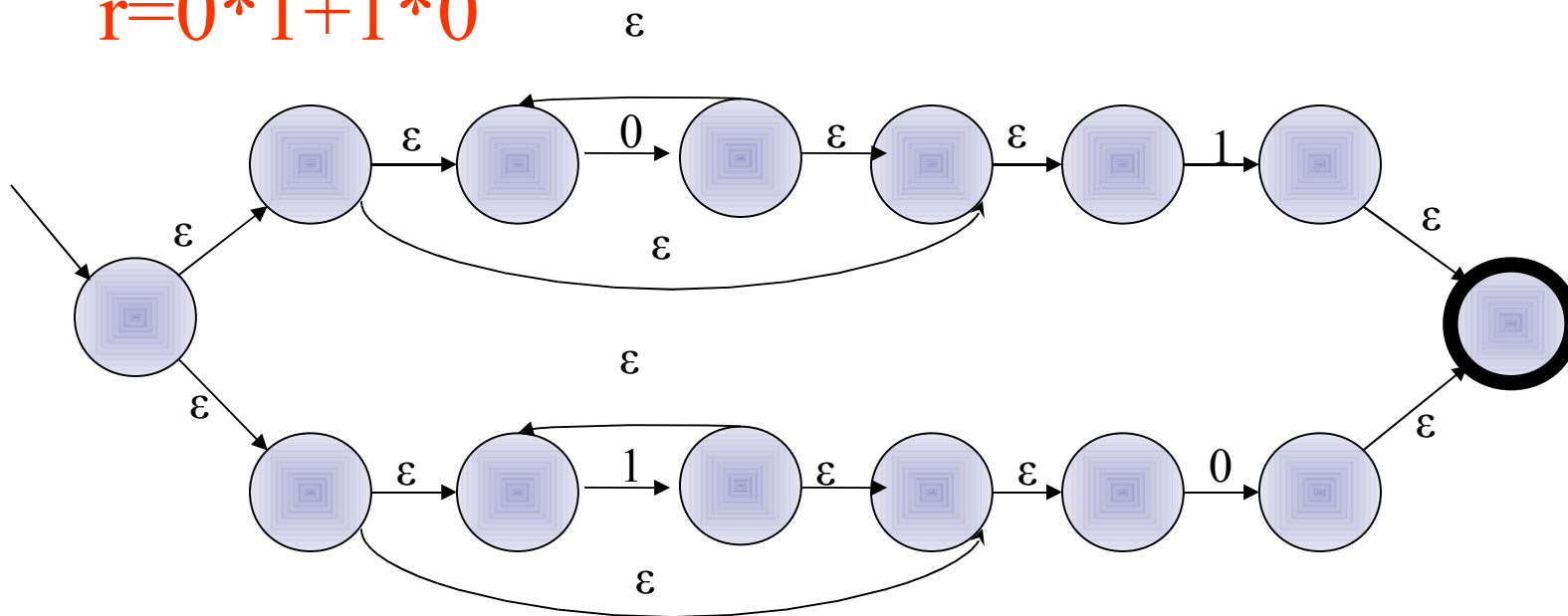


בניית אוטומט מביטוי רגולרי

■ נתבונן בביטוי 0^*1+1^*0

■ נדגים בנייה של אוטומט המקבל את השפה המתאימה

$$r=0^*1+1^*0$$



דוגמאות לביטויים רגולריים ומשמעותם

משמעות	השפה אותה הוא מייצג	הביטוי הרגולרי
השפה המוגדרת היא קבוצת בת 2 מילים $a b$	$\{a,b\}$	$a+b$
כיוון שלשרשור יש עדיפות גבוהה יותר מאופרטור "או", השפה המוגדרת מכילה מילים שהם שרשור של 2 אותיות.	$\{aa,ab,ba,bb\}$	$(a+b) \cdot (a+b)$
השפה המתקבלת היא השפה המכילה את כל המילים שמתקבלות ע"י שרשור של 0 או יותר פעמים של a	$\{\epsilon, a, aa, aaa, \dots\}$	a^*
שפת כל המילים שיש בהן a או b	$\{a, b, ab, aaab, bbb, baabab, \dots\}$	$(a+b)^*$