

שיעור 5

התזה של צרץ טירינג ודקזוקים כלליים

5.1 היחס בין הכרעה וקבלה

משפט 5.1 כל שפה כריעה היא גם קבילה

כל שפה כריעת היא גם קבילה.

■ **הוכחה:** המכונה טירינג שמכריעה את L גם מקבלת אותה. נשאל שאלה. האם כל שפה קבילה היא גם כריעת? זאת שאלה שכרגע אין לנו מספיק כלים לענות עליה. נחזור לשאלת זו בפרק הבא. לבינתיים נוכיח טענה חלשה יותר.

משפט 5.2

תהי L שפה.

אם גם L וגם \bar{L} קבילות אז L כריעת.

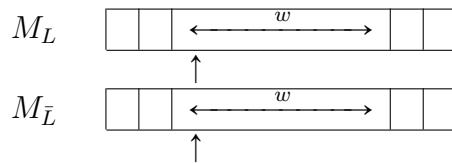
הוכחה: תהי M_L מ"ט שמקבלת את L , ותהי $M_{\bar{L}}$ מ"ט שמקבלת את \bar{L} .
נבנה מ"ט D_L שמכריעת את L .

כיצד תעבוד המ"ט D_L המכרייעת?

- נרים במקביל את M_L ואת $M_{\bar{L}}$.

- אם M_L מקבלת את המילה או נועבור ל-.acc.

- אם $M_{\bar{L}}$ מקבלת את המילה או נועבור ל-.re.



- הסימולציה מתבצעת ע"י סימולץ צעד צעד.

- * צעד במכונה M_L .

- * צעד במכונה $M_{\bar{L}}$.

- נמשיך בסימולציה המקבילה עד שאחת המכונות מגיעה למצב acc.

- * אם M_L מקבלת $\text{acc} \leftarrow$.
- * אם $M_{\bar{L}}$ מקבלת $\text{rej} \leftarrow$.
- לא יכול להיות מצב כי אף אחת מהמכונות לא מגיעה למצב acc כי כל מחרוזת $L \in w$ או $\bar{L} \in w$.

5.2 שיקולות של מכונת טיורינג ותוכנית מחשב

- מכונת טיורינג היא מודל חישובי למחשב.
- מחשב = תוכנית מחשב.
- תוכנית מחשב כתובה בשפת תכנות, למשל
 - * ג'אווה
 - * פיתון
 - C *
 - SIMPLE *

- המרכיבים של שפת תכנות הם
 - * משתנים
 - * פעולות
 - * תנאים
 - * זרימה

נוכיח כי מכונת טיורינג ותוכנית מחשב שקולים חישובי.

SIMPLE 5.3

משתנים

¹ i , j , k , . . .

- טבעיות:

מקבילים כערך מספר טבעי.

מערכות

- המערכתים אין-סופיים

¹ A [] ,
² B [] ,
³ C [] ,

- בכל תא ערך מתוק א"ב Γ.

אתחול

- כל המשתנים מאוחזרים ל- 0.
- הקלט נמצא בתאים הראשונים של []A.

פעולות

- השמה בקבוע:

```
1 i=3, B[i]="#"
```

```
1 i=k, A[k]=B[i]
```

```
1 x = y + z , x = y - z , x = y.z
```

- פעולות חשבו:

• השמה בין משתנים:

תנאים

- $B[i]==A[j]$ (מערכים).

- $y \geq x$ (משתנים טבעיות).

זרימה

- סדרה פקדות ממושפרות.

- goto : מותנה ולא מותנה.

- עזירה עם ערך חזרה.

```
1 one = 1
2 zero = 0
3 B[zero] = "0"
4 i=0
5 j=i
6 if A[i] == B[zero] goto 9
7 i=j + one
8 goto 3
9 C[one] = A[j]
10 if C[one] == A[zero] goto 12
11 stop(0)
12 stop(1)
```

כעת נגדיר את מושגי הקבלה והדחיה של מילים בשפה SIMPLE, ונגדיר את מושגי הכרעה והקבלה של שפות בשפה SIMPLE.

הגדרה 5.1 קבלה ודחיה של מחרוזות בשפה SIMPLE

עבור קלט w ותוכנית P בשפת SIMPLE. אומרים כי

- P מקבלת את w אם הריצה של P על w עוצרת עם ערך חזרה 1.
- P דוחה את w אם הריצה של P על w עוצרת עם ערך חזרה 0.

הגדרה 5.2 הכרעה וקבלה של מחרוזות בשפה SIMPLE

עבור שפה L ותוכנית P בשפת SIMPLE. אומרים כי

- P מבירעת את L אם היא מקבלת את המילים שב- L ודוחה את אלה שלא ב- L .
- P מקבלת את L אם היא מקבלת את כל וرك המילים ב- L .

משפט 5.3

המודלים של מכונת טיריניג ותוכנית SIMPLE שקולים.

הוכחה:

כיוון ראשון:

נוכיח כי לכל מ"ט M קיימת תוכנית P שcolaה.
בוצע סימולציה של מ"ט M במחשב P .

בלי להכנס פרטים, די ברור שבשפה עילית, כגון ג'אווה, ניתן להציג מבני נתונים עברור כל מרכיבי מכונת טיריניג:

- הסרט.
- המצביעים.
- מיקום הראש.
- טבלת המעברים.

ברור שנית לבצע סימולציה של פעילות המכונה.
ואם ניתן לעשות זאת בשפה עילית, ניתן לעשות זאת גם בשפת SIMPLE.

כיוון שני:

נוכיח כי לכל תוכנית P בשפה SIMPLE קיימת מ"ט Shcola.

אנחנו צריכים להראות כיצד ניתן למש את הרכיבים השונים של תוכניות SIMPLE במ"ט.

הרכיבים הם:

- משתנים.
- פעולה.
- תנאים.
- זרימה.

משתנים

לכל משתנה יהיה סרט משלו.
המספר שהמשתנה יחזק יוצג בסיס אונרי.
בהתחלת הסרט יהיה רק עם רוחחים, זה מייצג את המספר אפס בסיס אונרי.

לכל מערך יהיה סרט משלו.
בכל תא הסרט המערך תהיה אותן.
בהתחלת כל המערכות יהיו מאופסים למעט הסרט הראשון, שיחזק את הקלט.

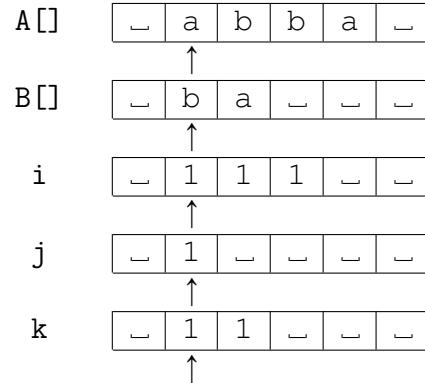
למשל ההשמה הבאה של משתנים בשפה SIMPLE:

```

1 A[1] = a, A[2] = b, A[3] = b , A[4] = a
2 B[1] = b, B[2] = a
3 i = 3
4 j = 1
5 k = 2

```

ניתן למשם במ"ט על ידי לכתוב על סרטים, שרט אחד לכל משתנה:

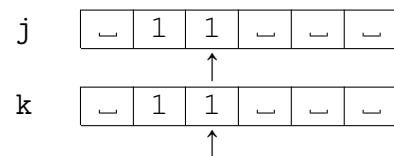
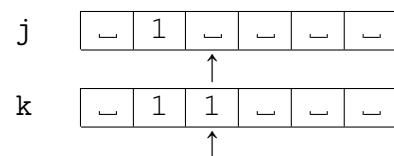


פעולות

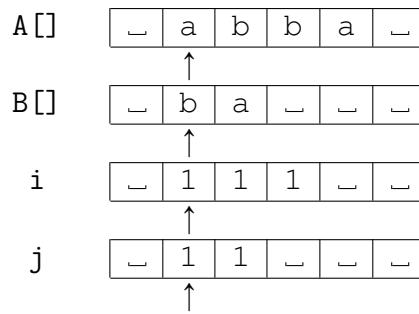
כעת נניח שנשים

```
1 j = k
```

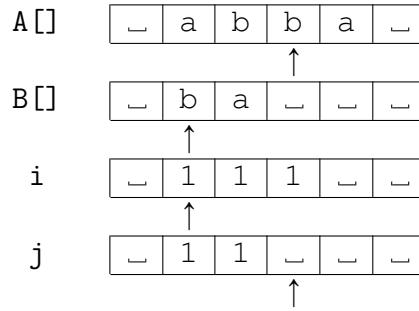
אפשר למשם את ההשמה הזאת על ידי להעתיק את תוכן הסרט של המשתנה k לסרט של המשתנה j.



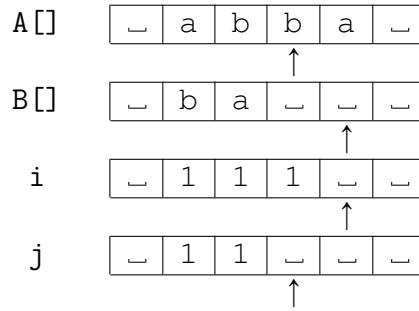
כעת נניח שנשים $[j]=A[i], [B[3]]=A[2], \dots$.
המשם זה במ"ט ע"י להעתיק את תוכן משבצת 2 בסרט של $[A]$ למשבצת 3 בסרט של $[B]$.



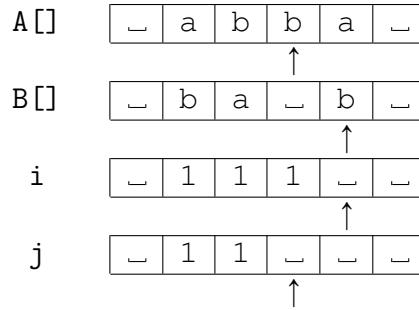
שלב 2



שלב 3



שלב 4



נניח עכשו שאנו רוצים לשים

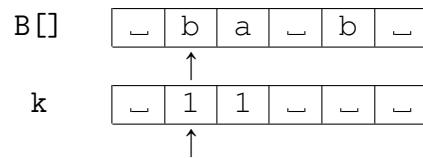
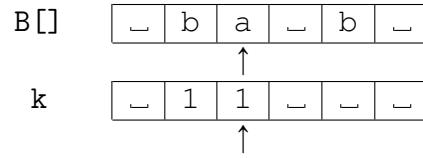
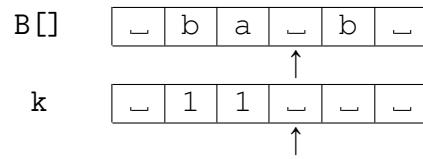
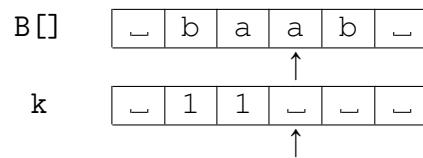
¹ B [k] <- "a"

א"

¹ B [3] <- "a"

נממש זה במת ע"י על ידי הפעולות הבאות עם הסרט של B [] והסרט של k.

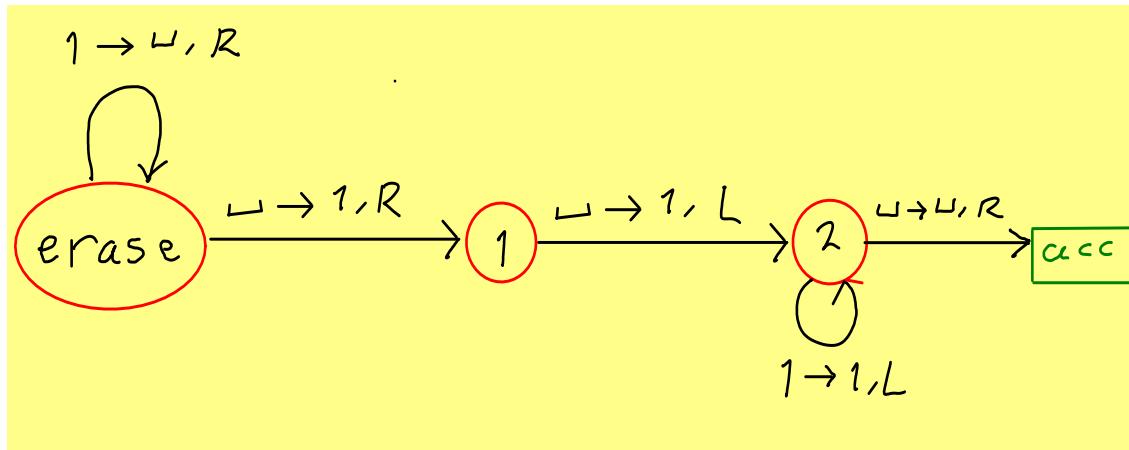
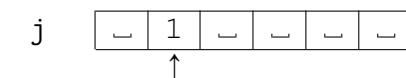
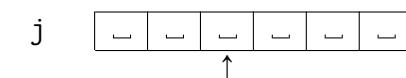
שלב 1

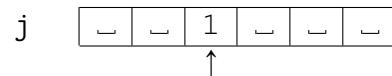
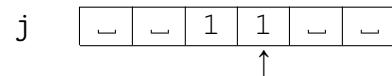
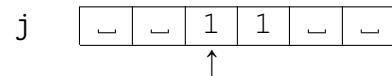
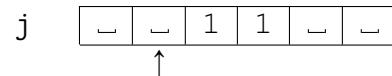
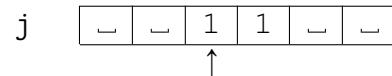
**שלב 2****שלב 3****שלב 4**

cut נניח שאנו רוצים לשים

1 j=2

אז נממש זה במת עם הפעולות הבאות:

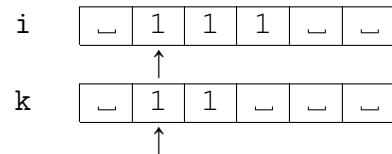
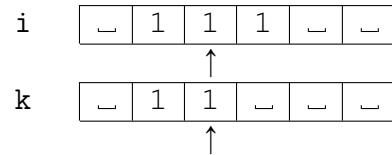
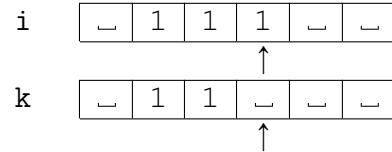
**שלב 1****שלב 2**

שלב 3)**שלב 4)****שלב 5)****שלב 6)****שלב 7)**תנאים

נניח שאנו רוצים למש את התנאי

`i >= k`

נitin לבדוק את התנאי במת' על ידי הפעולות הבאות:

שלב 1)**שלב 2)****שלב 3)****5.4 דקדוקים כלליים**

הגדרה 5.3 דקדוקים חסרי קשר

דקדוק חסר קשר הוא קבוצה

$$(V, \Sigma, R, S)$$

כאשר

- V קבוצה סופית של **משתנים** שמורכב מאותיות גדולות של אלףיבית.
- Σ קבוצה סופית של **טרמינלים** שמורכב מאותיות קטנות וסימנים של אלףיבית.
- R קבוצה של כללים. כל כלל הוא מצורה

$$\gamma \rightarrow u$$

כאשר $V \in \gamma$ משתנה בודד בצד שמאל ו- $* (\Sigma \cup u)$ מחרוזת של משתנים וטרמינלים בצד ימין
 $S \in V$ • המשתנה ההתחלתי.

דוגמה 5.1

נתון הדקדוק חסר קשר:

$$G_1 = (\{A, B\}, \{0, 1, \#\}, R, A)$$

הקבוצת משתנים היא $V = \{A, B\}$, הקבוצת טרמינלים היא $\{0, 1, \#\}$, המשתנה ההתחלתי הוא $S = A$ והכללים של הדקדוק הם

$$R = \begin{cases} A \rightarrow 0A1 \\ A \rightarrow B \\ B \rightarrow \# . \end{cases}$$

הגדרה 5.4 יצירה של מילה על ידי דקדוק חסר קשר

- 1) כתבו את המשתנה ההתחלתי S .
- 2) מצאו משתנה וכלל אשר מתחליל אם המשתנה זה, והחליפו אותו עם המחרוזת בצד ימין של הכלל.
- 3) חזרו על שלבים 1 ו- 2 עד שלא נשאר אף משתנים של V .

דוגמה 5.2הדקוק G_1 יוצר את המחרוזת 000#111

$$A \xrightarrow{A \rightarrow 0A1} 0A1 \xrightarrow{A \rightarrow 0A1} 00A11 \xrightarrow{A \rightarrow 0A1} 000A111 \xrightarrow{A \rightarrow B} 00B11 \xrightarrow{B \rightarrow \#} 000\#111$$

דוגמה 5.3

נתון את הדקדוק

$$G_2 = (\{S, T, F\}, \{(,), +, \times, a\}, R, S)$$

כאשר הכללים הם

$$R = \begin{cases} S \rightarrow S + T \\ S \rightarrow T \\ T \rightarrow T \times F \\ T \rightarrow F \\ F \rightarrow (S) \\ F \rightarrow a . \end{cases}$$

: $a + a$ יוצר את המילה: G_2

$$S \xrightarrow{S \rightarrow S+T} S + T \xrightarrow{S \rightarrow T} T + T \xrightarrow{T \rightarrow F} F + F \xrightarrow{F \rightarrow a} a + a$$

בדקdock כללי, גם בצד ימין וגם בצד שמאל יכולה להופיע מחרוזת של משתנים וטרמינליים. פורמלי:

הגדירה 5.5 דקדוקים כלליים

דקדוק חסר קשר הוא קבוצה

$$(V, \Sigma, R, S)$$

כאשר

- V קבוצה סופית של **משתנים** שמורכב מאותיות גדולות שלalfבית.
- Σ קבוצה סופית של **טרמינלים** שמורכב מאותיות קטנות וסימנים שלalfבית.
- R קבוצה של כללים. כל כלél הוא מצורה

$$\gamma \rightarrow u$$

כאשר $(\Sigma^* \cup V)^+$, $\gamma \in (V \cup \Sigma)^*$, $u \in \Sigma^*$. מחרוזת של משתנים וטרמינליים בצד ימין $S \in V$ המשתנה ההתחלתי.

דוגמה 5.4

נתון את הדקדוק

$$G = (\{S, [,]\}, \{a\}, R, S)$$

שבו הקבוצת משתנים היא $V = \{S, [,]\}$, הקבוצת טרמינליים היא $\{a\}$ והכללים הם

$$R = \begin{cases} S \rightarrow [S] \\ S \rightarrow a \\ [a \rightarrow aa[\\ [\rightarrow \varepsilon . \end{cases}$$

:aaaa יוצר את המילה: G

$$\begin{array}{ccccccc} S & \xrightarrow{S \rightarrow [S]} & [S] & \xrightarrow{S \rightarrow [S]} & [[S]] & \xrightarrow{S \rightarrow a} & [[a]] \\ & \xrightarrow{[\rightarrow \varepsilon } & [aa] & \xrightarrow{[a \rightarrow aa[} & aa[a] & \xrightarrow{[a \rightarrow aa[} & aa aa[] \\ & & & & & \xrightarrow{[\rightarrow \varepsilon } & aaaa \end{array}$$

דוגמה 5.5

נתון את הדקדוק

$$G = (\{S, [,]\}, \{a\}, R, S)$$

שבו הקבוצת משתנים היא $V = \{S, [,]\}$, הקבוצת טרמינליים היא $\{a\}$ והכללים הם

$$R = \begin{cases} S \rightarrow [S] \\ S \rightarrow a \\ [a \rightarrow aa] \\ [] \rightarrow \varepsilon . \end{cases}$$

מהן המילים שנitin נוצר בעזרת הדקדוק הזה,
או במילים אחרות: מהי השפה של הדקדוק?

פתרון:

תשובה:

$$L(G) = \{a^n \mid n = 2^k, k \geq 1\} .$$

הסבר:



דוגמה 5.6

בנו דקדוק כללי אשר יוצר את הפסה

$$L = \{w \in \{a, b\}^* \mid \#a_w = \#b_w\} .$$

פתרון:

$$G = (\{S\}, \{a, b\}, R, \{S\})$$

$$S \rightarrow abS , \tag{1}$$

$$ab \rightarrow ba , \tag{2}$$

$$ba \rightarrow ab , \tag{3}$$

$$S \rightarrow \varepsilon . \tag{4}$$

$$S \xrightarrow{1} abS \xrightarrow{1} ababS \xrightarrow{2} baabS \xrightarrow{4} baab$$

שימוש לב: בדקדוק כללי אנו מאפשרים גם כליל'יצרה בהם מצד שמאל יש רק טרמינלים.
לכן, ניתן גם שנמשיך ונפתח מחזורות של תרמינליים. למשל

$$S \xrightarrow{1} abS \xrightarrow{1} ababS \xrightarrow{4} abab \xrightarrow{2} baab$$

נשאל שאלה כללית:

- אלו שפות ניתנים לצורך בעזרת דקדוק כללי?
- האם יש שפות שלא ניתנים לצורך בעזרת דקדוק כללי?
- האם יש מודל חישובי שמקבל שפות שנוצרות ע"י דקדוקים כלליים?

דוגמה 5.7

בנו דקדוק כללי שיוצר את השפה

$$w = \{w \in \{a, b, c\}^* \mid w = a^n b^n c^n\}$$

פתרון:

נראה דקדוק כללי עבור שפה זו.

שפה זו אינה חסרת הקשר.

לכן, לא ניתן לבנות עבורה דקדוק חסר הקשר.

אנו נבנה לה דקדוק כלל.

נעזר את האותיות c, a, b יחד.

נעsha זאת בצורה כזו שכדי לסיים את תהליך הגזירה יש לסדר את האותיות בסדר הרצוי:

תחילה a ,אחר מכן b ,ובסוף c .

$$S \rightarrow S'] \quad (1)$$

$$S' \rightarrow aS'bC \mid \varepsilon \quad (2)$$

$$Cb \rightarrow bC \quad (3)$$

$$C] \rightarrow]c \quad (4)$$

$$] \rightarrow \varepsilon \quad (5)$$

$$\begin{array}{llllll} S & \xrightarrow{1} & S'] & \xrightarrow{2} & aS'bC] & \xrightarrow{2} aaS'bCbC] & \xrightarrow{2} aaaS'bCbCbC] \\ & \xrightarrow{3} & aaaS'bbCCbc] & \xrightarrow{3} & aaaS'bbCbCC] & \xrightarrow{3} & aaaS'bbbCCC] \\ & \xrightarrow{4} & aaaS'bbbCC]c & \xrightarrow{4} & aaaS'bbbC]cc & \xrightarrow{4} & aaaS'bbb]ccc \\ & \xrightarrow{5} & aaaS'bbbccc & \xrightarrow{1} & aaabbccc & & \end{array}$$

דוגמה 5.8

בנו דקדוק כללי אשר יוצר את שפת המיללים

$$L = \{ uu \mid u \in \{a, b\}^*\}$$

פתרון:

דוגמא זאת תמחיש כיצד דקדוק כללי יכול "לפעול בדומה" למוכנת טיריניג.

דקודק נשתמש במשתנים וכלי גירה שיאפשרו מעין תנעה על גבי המחרוזות הנוצרת, בדומה לתנועת הראש של מוכנת טיריניג על גבי הסרט.

S → [H {	כל גזירה ייחיד מהמשתנה ההתחלתי. המשתנה H ידמה את הראש של המ"ט ש"יזוז" מצד לצד על המחרוזות הנגזרות. הסוגר המרובע] מסמן את הקצה השמאלי של המילה השמאלית. הסוגר המסלול } מסמן את הקצה השמאלי של המילה הימנית.	1
[H → [aH _a	כל זה מאפשר הוספת אות a לקצה השמאלי של המילה השמאלית. המשתנה H מוחלף במשתנה H _a כדי "לזכור" שיש עכשו להוסיף a גם במחוזות הימנית. (בדומה לזכרון של מ"ט).	2
H _a a → aH _a	כל זה מאפשר בראש "יזוז" ימינה.	3
H _a { → H{a	כאשר המשתנה H _a "גינע" לסוגר המסלול, הוא יجوز אות a נוספת מימין לסוגר, שהוא הקצה השמאלי של המחרוזות הימנית. כך אפשרנו להוסיף שתי אותיות a: אחת מימין לסוגר] ואחת תואם ימין לסוגר } . כלומר אותן a בקצת השמאלי של כל אחת המחרוזות.	4
aH → Ha	cut ציריך לאפשר למשתנה H לחזור שמאלה על גבי האותיות שבין הסוגרים, עד הסוגר].	5

ברגע "שהראש" H חזר לתחילת המחרוזות ועומד ליד הסוגר] עברים על שלבים 5-2 שוב. בסבב הבא נחק במחשב גם יקרה של שתי אותיות b.

[H → [bH _b	כל זה מאפשר הוספת אות b לקצה השמאלי של המילה השמאלית. המשתנה H מוחלף במשתנה H _b כדי "לזכור" שיש עכשו להוסיף b גם במחוזות הימנית.	2'
H _a a → aH _a H _a b → bH _a H _b a → aH _b H _b b → bH _b	כללים האלהאפשרים בראש "יזוז" ימינה.	3'
H _b { → H{b	כאשר המשתנה H _b "גינע" לסוגר המסלול, הוא יجوز אות b נוספת מימין לסוגר, שהוא הקצה השמאלי של המחרוזות הימנית. כך אפשרנו להוסיף שתי אותיות b: אחת מימין לסוגר] ואחת תואם ימין לסוגר } . כלומר אותן b בקצת השמאלי של כל אחת המחרוזות.	4'
bH → Ha	Cut ציריך לאפשר למשתנה H לחזור שמאלה על גבי האותיות שבין הסוגרים, עד הסוגר].	5'

בכדי לסייע את הגזירה יש להפטר ממשתני העזר על ידי הכללים הבאים:

H → ε [→ ε { → ε	הכללים האלה אפשרים להעלים את המשתנים } , [, H	6
-------------------------	--	---

למשל:

$$\begin{array}{ccccccccc}
 S & \xrightarrow{1} & [H \{ & \xrightarrow{2} & [aH_a \{ & \xrightarrow{4} & [aH \{ a & \xrightarrow{5} & [Ha \{ a \\
 & & \xrightarrow{2} & & [aH_a a \{ a & \xrightarrow{3} & [aaH_a \{ a & \xrightarrow{4} & [aaH \{ aa & \xrightarrow{5} & [Haa \{ aa \\
 & & \xrightarrow{2} & & [bH_b a a \{ aa & \xrightarrow{3} & [baaH_b \{ aa & \xrightarrow{4} & [baaH \{ baa & \xrightarrow{5} & [Hbaa \{ baa \\
 & \xrightarrow{6} & baabaa & & & & & &
 \end{array}$$

5.5 דקדוקים כלליים ומכונת טיורינג

משפט 5.4 קדוקים כלליים ומכונת טיורינג

תהי L שפה. L קבילה אם ורק אם קיים דקדוק כללי G כך ש- $L = L(G)$.

הוכחה: **ביוון ראשון.**

נוכיח שאם קיים דקדוק כללי G אז $L(G)$ קבילה.

נניח שקיימים דקדוק כללי G . נוכיח כי $L(G)$ קבילה על ידי להוכיח שקיימת תוכנית מחשב P שמקבלת $P(L(G))$.

נתון דקדוק כללי G . נבנה תוכנית מחשב שמקבלת את $L(G)$.
יהי הקלט $w \in L(G)$, מילה בשפה G .

$w=S$ (1)

:repeat (2)

- פצל באופן לא דטרמיניסטי את w ל- xyz .
- בחר באופן לא דטרמיניסטי גזירה $v \rightarrow t$ של G .
- אם $t \neq y$ דחלה.
- $zv=x$
- אם $w==v$ קובל.

ביוון שני.

נוכיח שאם $L(G)$ קבילה אז קיים דקדוק כללי G .

צ"א, נניח שקיימות מ"ט M שמקבלת את השפה L . נוכיח שקיימים דקדוק כללי G כך ש- $L = L(G)$.
כלומר השפה המתקבלת על ידי M היא השפה של דקדוק כללי G .

נתונה מ"ט M בעלת הtablת המעברים להלן. נבנה דקדוק כללי G שممמש אותם צעדים.

תואזה	כתביה	מצב חדש	סימן	מצב
R	a	q_0	q_0	a
R	b	q_1	q_1	b
L	-	acc	-	-
L	a	q_0	q_0	a
L	b	q_1	q_1	b

לפי הtablת המעברים קיימים הצעדים

$q q_0 b a b \vdash_M aaq_1 ab$

נניח שבדוק כללי G קיים אותו הצעד

$$q q_0 bab \xrightarrow{G} aaq_1 ab$$

ניתן למעשה צעד זה על ידי הכלל

$$q_0 \xrightarrow{b} a \quad q_1$$

באופן כללי,

- עבור כל פונקציית המעברים של M שגוררת תזוזה ימינה מצורה

$$\delta(q, \sigma) = (p, \pi, R)$$

נமמש מעבר זה על ידי כלל של הדקדוק G מצורה

$$q\sigma \rightarrow \pi p .$$

- עבור כל פונקציית המעברים של M שגוררת תזוזה שמאליה מצורה

$$\delta(q, \sigma) = (p, \pi, L)$$

או כלל $\Gamma \in \tau$ ב- G נממש מעבר זה על ידי הכלל

$$\tau q\sigma \rightarrow p\tau\pi .$$

■

5.6 היררכיה של חומסקי

מודל חישובי	דקדוק	משפחה שפות
מכונת טיורינג	כללי	קבילות
אוטומט מחסנית	חסירות הקשר	אוטומטים רולריים וمتקובלות על ידי
אוטומט סופי	רגולרי	רגולריות

- היררכיה של חומסקי קושرت לנו בין משפחות של שפות דקדוקים ומודלים חישוביים.
- בתחום ההיררכיה נמצאות השפות הרגולריות שנוצרות על ידי דקדוקים רולריים וمتקובלות על ידי אוטומטים סופיים.
- מעלהן נמצאות השפות חסירות הקשר שנוצרות על ידי דקדוקים חסרי הקשר וمتקובלות על ידי אוטומטי מחסנית.
- מעלהן נמצאות השפות הקבילות שנוצרות על ידי דקדוקים כלליים וمتקובלות על ידי מכונות טיורינג.
- כל רמה בהיררכיה מכילה ממש את הרמה שמתחתה.
- * כל שפה רגולרית היא גם חסירת הקשר, אבל יש שפות חסירות הקשורין רגולריות.
- * כל שפה חסירת הקשר היא קבילה, אבל יש שפות קבילות שאין חסירות הקשר.

5.7 כל שפה חסירת הקשר הינה קריאה

לפי היררכיה של חומסקי אנחנו יודעים לקבוע שכל שפה חסירת הקשר היא קבילה.

האם כל שפה חסירת הקשר הינה קריאה?

משפט 5.5

יהי $G = (V, \Sigma, S, R)$ דקדוק חסר הקשר ו- $w \in L(G)$. אזי קיים עץ גזירה של w שעומקו לכל היותר $(|V| + 1)(|w| + 1)$.

הוכחה: יהי T עץ הגזירה הקטן ביותר (מבחינת מספר קודקודים) של w .
בשלילה נניח שב- T **יש מסלול מהשורש לעלה שמכיל לפחות** $(|V| + 1)(|w| + 1)$ **קודקודים פנימיים.**
 נסמן מסלול זה ב-

$$p = (u_1, u_2, \dots, u_m) .$$

עבור קודקוד u_i במסלול נסמן ב- (u_i) את תת-המחוזת של w שנוצרת מ- u_i .

מתקיים ש- s היא תת-מחוזת של (u_i) . אומרים שקודקוד u_i הוא **משמעותי** אם (u_i) מכיל ממש את s .

כל קודקוד **משמעותי** מוסיף לפחות אחת ל- w .
 לכן, ישנו לכל היותר $|w|$ קודקודים **משמעותיים**.
 לכן, ברגע הקודקודים הפנימיים (u_1, u_2, \dots, u_m) שאורכן לפחות $(|V| + 1)(|w| + 1)$, בהכרח ישנו תת רצף $(u_i, u_{i+1}, \dots, u_{i+|V|+1})$, שבו כל הקודקודים לא **משמעותיים**.
ברצף זה בהכרח ישנים שני קודקודים, נאמר u_j, u_k , $j < k$ **משמעותניים** עם אותו משתנה.
 לכן בעץ הגזירה, ניתן להחליף את הקודקוד u_j יחד עם כל תת העץ שמתחתיו - בקודקוד u_k , יחד עם כל תת העץ שמתחתיו.
 כיוון שכל הקודקודים שבין u_j ל- u_k (כולל) הם לא **משמעותיים**, **החלפה זו לא משנה את המחרוזות הנוצרת**.

כלומר, העץ החדש גם הוא עץ הגזירה עבור w .
 בסתיו להנחה המינימלית של העץ.

משפט 5.6

כל שפה חסרת הקשר היא כריעה.

הוכחה: בהינתן דקדוק חסר הקשר $G = (V, \Sigma, S, R)$, התוכנית הלא דטרמיניס הבאה מכריעת את $L(G)$.

קלט: מחרוזת w .
פלט: כן או לא.

- 1) נחש עץ גזירה של הדקדוק G בעומק לכל היותר $(|V| + 1)(|w| + 1)$.
 - 2) בדוק האם העץ יוצר את המחרוזת w . אם כן, החזר "כן" איתר החזר "לא".
- שני שלבי התוכנית בהכרח מסוימים. לכן, זו תוכנית להכרעה. ישנו חישוב שמחזיר "כן" אם ורק אם $w \in L(G)$.
- לכן זו תוכנית שמכריעת את $L(G)$.