

חשוביות וסיבוכיות

תוכן העניינים

2	1 מכונות טיורינג
2	הגדרה היוריסטית של מכונת טיורינג
6	הגדרה פורמלית של מכונת טיורינג
19	טבלת המעברים
23	חישוב פונקציות

שיעור 1

מכונות טיורינג

1.1 הגדרה היוריסטית של מכונת טיורינג

הגדרה 1.1 מכונת טיורינג (הגדרה היוריסטית)

הקלט והסרט

מכונת טיורינג (מ"ט) קורא קלט. הקלט נמצא על סרט אינסופי. התווים של הקלט נמצאים במשבצות של הסרט. במכונת טיורינג אנחנו מניחים שהסרט אינסופי לשני הכיוונים. משמאל לתחילת הקלט לא כתוב כלום, ומימין לסוף הקלט לא כתוב כלום. אנחנו מניחים שיש תו הרווח $_$ שנמצא בכל משבצות שאינן משבצות קלט, משמאל לקלט ומימין לקלט.

...	_	_	a	b	b	b	a	a	_	_	_	...
-----	---	---	---	---	---	---	---	---	---	---	---	-----

הראש

במצב ההתחלתי הראש בקצה השמאלי של הקלט.

...	_	_	a	b	b	b	a	a	_	_	_	...
			↑									

הראש יכול לזוז ימינה על הסרט וגם שמאלה על הסרט. הראש יכול לקרוא את התוכן שנמצא במשבצת הסרט שבה הוא נמצא. הראש יכול לכתוב על המשבצת הסרט שבה הוא נמצא. הכתיבה נעשית תמיד במיקום הראש.

המצבים

בהתחלה הראש בקצה השמאלי של הקלט והמ"ט במצב התחלתי q_0 . הראש קורא את התו במשבצת הראשונה וכותב עליה לפי הפונקציית המעברים (שנגדיר בהגדרה 1.2). כעת המ"ט במצב חדש q_1 . הראש קורא את התו במשבצת השניה וכותב עליה לפי הפונקציית המעברים ואז המ"ט במצב חדש q_2 . התהליך ממשיך עד שהראש מגיע לקצה הימני של הקלט, ואז הוא ממשיך לקרוא ולכתוב על כל משבצת בכיוון שמאלה, עד שהוא מגיע לקצה השמאלי. במ"ט ניתן לטייל על הקלט שוב ושוב לשני הכיוונים. התהליך מסתיים כאשר המ"ט מגיע למצב מקבל q_{acc} או מצב דוחה q_{rej} .

דוגמה 1.1

נבנה מכונת טיורינג אשר מקבלת מילה אם היא בשפה

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

ז"א השפה המורכבת מכל המילים עם מספר שווה אותיות a ו b .

תיאור מילולי

- נסרוק את הקלט משמאל לימין ולכל a נחשפ b תואם.
 - נניח שראינו במשבצת הראשונה a , נסמן עליה \checkmark .
 - עכשיו נסרוק את יתרת הקלט ונחפש אות b מתאימה ל a שכבר ראינו.
 - אם לא מצאנו, המילה לא בשפה.
 - אם מצאנו, נסמן את ה- b התואם ב- \checkmark .
 - נחזור לתחילת הקלט ונעשה סריקה נוספת משמאל לימין.
 - במשבצת הראשונה יש \checkmark מסיבוב הראשון. הראש פשוט כותב עליה \checkmark , כלומר משבצת ראשונה נשארת ללא שינוי.
 - הראש z שמאלה למשבצת הבאה. נניח שמצאנו b . נסמן במשבצת \checkmark .
 - נסרוק את יתרת הקלט ונחפש אות a מתאימה ל b שכבר ראינו.
 - אם לא מצאנו, המילה לא בשפה.
 - אם מצאנו, נסמן את ה- a התואם ב- \checkmark .
 - בכל משבצת שיש \checkmark כותבים עליה \checkmark וממשיכים למשבצת הבאה הימני.
 - נחזור לתחילת הקלט ונעשה סריקה נוספת משמאל לימין.
 - חוזרים על התהליך שוב ושוב.
 - אם היה מעבר שבו לא מצאנו אות תואמת, המילה לא בשפה.
 - אם כולן היו תואמות ועשינו מעבר שבו הגכנו מקצה לקצה, מרווח לרווח, בלי לראות שום אות, אז המילה בשפה.
- כעת נתאר את המ"ט באמצעות המצבי המכונה והפונקציית המעברים.

מצבי המכונה

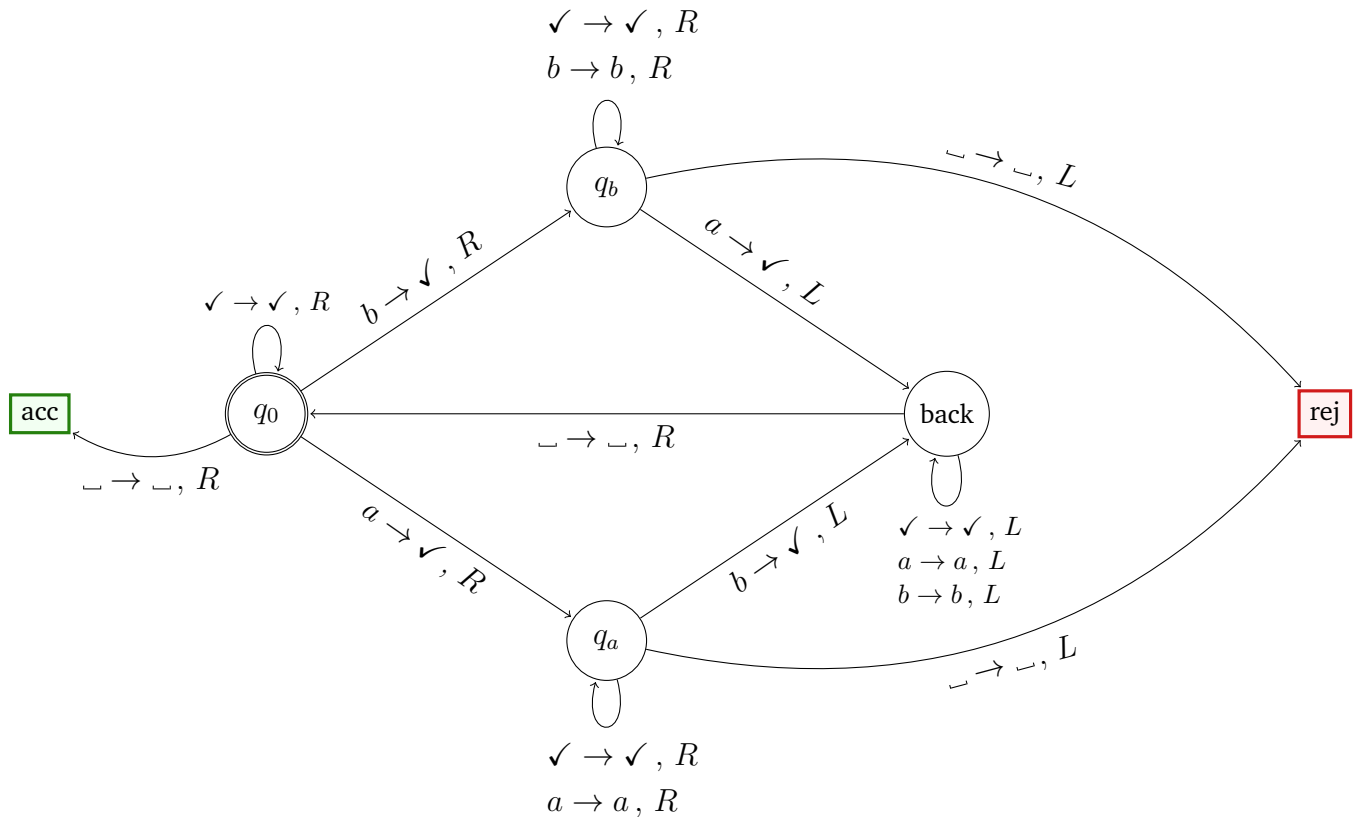
q_0	המצב ההתחלתי. אליו נחזור אחרי כל סבב התאמה של זוג אותיות.
q_a	מצב שבו ראינו a ומחפשים b תואם.
q_b	מצב שבו ראינו b ומחפשים a תואם.
back	מצב שנשתמש בו כדי לחזור לקצה השמאלי של הקלט ולהתחיל את הסריקה הבאה (סבב ההתאמה הבא).
acc	מצב מקבל.
rej	מצב דוחה.

- כאשר המכונה מגיעה למצב acc היא עוצרת.

עצירה במצב acc משמעותה קבלה.

- כאשר המכונה מגיעה למצב rej היא עוצרת.
עצירה במצב rej משמעותה דחייה.
- רק בשני מצבים אלו המכונה מפסיקה.
בכל מצב אחר המכונה בהכרח ממשיכה.

תרשים מצבים



- בכל צעד המכונה מבצעת שתי פעולות:

1. כותבת אות במיקום הראש
2. זזה צעד אחד שמאלה או צעד אחד ימינה.

- בכל צעד המכונה יכולה לעבור למצב אחר או להישאר באותו מצב.

דוגמה 1.2

בדקו אם המכונת טיורינג של הדוגמה 1.1 מקבלת את המילה $abbbbaa$.

פתרון:

␣	q_0	a	b	b	b	a	a	␣
␣	✓	q_a	b	b	b	a	a	␣
␣	back	✓	✓	b	b	a	a	␣
back	␣	✓	✓	b	b	a	a	␣
␣	q_0	✓	✓	b	b	a	a	␣

⌊	✓	q_0	✓	b	b	a	a	⌊
⌊	✓	✓	q_0	b	b	a	a	⌊
⌊	✓	✓	✓	q_b	b	a	a	⌊
⌊	✓	✓	✓	b	q_b	a	a	⌊
⌊	✓	✓	✓	back	b	✓	a	⌊
⌊	✓	✓	back	✓	b	✓	a	⌊
⌊	✓	back	✓	✓	b	✓	a	⌊
⌊	back	✓	✓	✓	b	✓	a	⌊
⌊	q_0	✓	✓	✓	b	✓	a	⌊
⌊	✓	q_0	✓	✓	b	✓	a	⌊
⌊	✓	✓	q_0	✓	b	✓	a	⌊
⌊	✓	✓	✓	q_0	b	✓	a	⌊
⌊	✓	✓	✓	✓	q_b	✓	a	⌊
⌊	✓	✓	✓	✓	✓	q_b	a	⌊
⌊	✓	✓	✓	✓	back	✓	✓	⌊
⌊	✓	✓	✓	back	✓	✓	✓	⌊
⌊	✓	✓	back	✓	✓	✓	✓	⌊
⌊	✓	back	✓	✓	✓	✓	✓	⌊
⌊	back	✓	✓	✓	✓	✓	✓	⌊
⌊	q_0	✓	✓	✓	✓	✓	✓	⌊
⌊	✓	q_0	✓	✓	✓	✓	✓	⌊
⌊	✓	✓	q_0	✓	✓	✓	✓	⌊
⌊	✓	✓	✓	q_0	✓	✓	✓	⌊
⌊	✓	✓	✓	✓	q_0	✓	✓	⌊
⌊	✓	✓	✓	✓	✓	q_0	✓	⌊
⌊	✓	✓	✓	✓	✓	✓	q_0	⌊
⌊	✓	✓	✓	✓	✓	✓	⌊	acc

דוגמה 1.3

בדקו אם המכונת טיורינג של הדוגמה 1.1 מקבלת את המילה aab.

פתרון:

⌊	q_0	a	a	b	⌊
⌊	✓	q_a	a	b	⌊
⌊	✓	a	q_a	b	⌊
⌊	✓	back	a	✓	⌊
⌊	back	✓	a	✓	⌊
back	⌊	✓	a	✓	⌊
⌊	q_0	✓	a	✓	⌊
⌊	✓	q_0	a	✓	⌊
⌊	✓	✓	q_a	✓	⌊
⌊	✓	✓	✓	q_a	⌊
⌊	✓	✓	rej	✓	⌊

1.2 הגדרה פורמלית של מכונת טיורינג

הגדרה 1.2 מכונת טיורינג

מכונת טיורינג (מ"ט) היא שביעה

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$$

כאשר:

Q	קבוצת מצבים סופיות
Σ	א"ב קלט סופי
Γ	א"ב סרט סופי
δ	פונקציית המעברים
q_0	מצב התחלתי
acc	מצב מקבל
rej	מצב דוחה

$$_ \notin \Sigma$$

$$\Sigma \subseteq \Gamma, _ \in \Gamma \text{ ref}$$

$$\delta : (Q \setminus \{\text{rej}, \text{acc}\}) \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$$

דוגמה 1.4 (המשך דוגמה 1.1)

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$$

$$Q = \{q_0, q_a, q_b, \text{back}, \text{rej}, \text{acc}\} .$$

$$\Sigma = \{a, b\} , \quad \Gamma = \{a, b, _, \checkmark\}$$

$$\delta(q_0, a) = (q_a, \checkmark, R) ,$$

$$\delta(q_0, b) = (q_b, \checkmark, R) ,$$

$$\delta(q_0, _) = (\text{acc}, _, R) ,$$

$$\delta(q_a, \checkmark) = (q_a, \checkmark, R) ,$$

$$\delta(q_a, a) = (q_a, a, R) ,$$

$$\delta(q_a, b) = (\text{back}, \checkmark, L) ,$$

$$\delta(q_b, \checkmark) = (q_b, \checkmark, R) ,$$

$$\delta(q_b, b) = (q_a, b, R) ,$$

$$\delta(q_b, a) = (\text{back}, \checkmark, L) ,$$

קל יותר לרשום את פונקציית המעברים δ בטבלה:

$Q \backslash \Gamma$	a	b	_	✓
q_0	(q_a, \checkmark, R)	(q_b, \checkmark, R)	$(\text{acc}, _, R)$	(q_0, \checkmark, R)
q_a	(q_a, a, R)	$(\text{back}, \checkmark, L)$	$(\text{rej}, _, L)$	(q_a, \checkmark, R)
q_b	$(\text{back}, \checkmark, L)$	(q_b, b, R)	$(\text{rej}, _, L)$	(q_b, \checkmark, R)
back	(back, a, L)	(back, b, L)	$(q_0, _, R)$	$(\text{back}, \checkmark, L)$

הגדרה 1.3 קונפיגורציה

תהי $M = (Q, q_0, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג.

קונפיגורציה של M הינה מחרוזת

$$\mu q \sigma \nu$$

כאשר משמעות:

$$\mu, \nu \in \Gamma^*, \quad \sigma \in \Gamma, \quad q \in Q.$$

q	מצב המכונה,
σ	הסימון במיקום הראש
μ	תוכן הסרט משמאל לראש,
ν	תוכן הסרט מימין לראש.

דוגמה 1.5 (המשך של דוגמה 1.3)

μ	q	σ	ν
␣	q_0	a	a b ␣
␣✓	q_a	a	b ␣
␣✓ a	q_a	b	␣
␣✓	back	a	✓ ␣
␣	back	✓	a ✓ ␣
␣	back	␣	✓ a ✓ ␣
␣	q_0	✓	a ✓ ␣
␣✓	q_0	a	✓ ␣
␣✓✓	q_a	✓	␣
␣✓✓✓	q_a	␣	␣
␣✓✓	rej	✓	␣

דוגמה 1.6

בנו מכונת טיורינג אשר מקבלת כל מילה בשפה

$$L = \{a^n \mid n = 2^k, k \in \mathbb{N}\}$$

ז"א מילים בעלי מספר אותיות a אשר חזקה של 2.

פתרון:

ראשית נשים לב:

$$\frac{n}{2^k} = 1 \text{ כלומר } k \text{ פעמים, ב- } 2 \text{ בדיוק של } n \text{ אחרי חילוק של } 1 \text{ אנחנו מקבלים}$$

לאור המשפט הזה נבנה אלגוריתם אשר מחלק את מספר האותיות במילה ב- 2 בצורה איטרטיבית. אם אחרי סבב מסויים נקבל מספר אי-זוגי גדול מ- 1 אז מספר האותיות a במילה לא יכול להיות חזקה של 2. אם אחרי כל הסבבים לא קיבלנו מספר אי-זוגי גדול מ- 1 אז מובטח לנו שיש מספר אותיות a אשר חזקה של 2.

- נתון הקלט

␣	␣	a	a	a	a	a	a	a	a	␣	␣
		↑									

נעבר על סרט הקלט. משמאל לימין.

- מבצעים מחקיה לסירוגין של האות a כלומר אות אחת נמחק ואות נשאיר וכן הלאה.

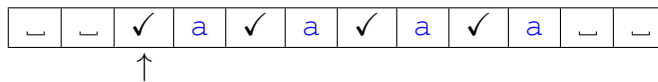
␣	␣	✓	a	✓	a	✓	a	✓	a	␣	␣
				↑							

אם אחרי סבב הראשון

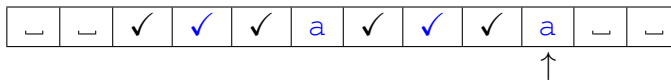
* יש ✓ בתו האחרון \Leftarrow קיבלנו מספר אי-זוגי של אותיות a אחרי חילוק ב- 2 \Leftarrow אין חזקה של 2 אותיות a במילה.

* יש a בתו האחרון \Leftarrow קיבלנו מספר זוגי של אותיות a אחרי חילוק ב- 2 ונמשיך לסבב הבא.

- הראש חוזר לתו הראשון של הקלט



- בסבב הבא חוזרים על התהליך של מחיקה לסירוגין של האות a (אות אחת נמחק ואות אחת נשאיר)

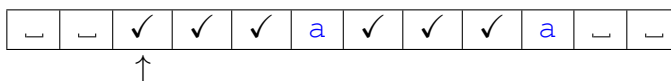


אם אחרי סבב השני

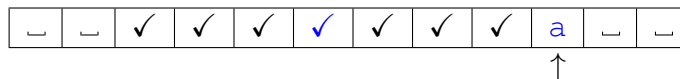
* יש ✓ בתו האחרון \Leftarrow קיבלנו מספר אי-זוגי של אותיות a אחרי חילוק ב- 2 \Leftarrow אין חזקה של 2 אותיות a במילה.

* יש a בתו האחרון \Leftarrow קיבלנו מספר זוגי של אותיות a אחרי חילוק ב- 2 ונמשיך לסבב הבא.

- הראש חוזר לתו הראשון של הקלט



- בסבב הבא חוזרים על התהליך של מחיקה לסירוגין של האות a (אות אחת נמחק ואות אחת נשאיר)



אם אחרי סבב השלישי

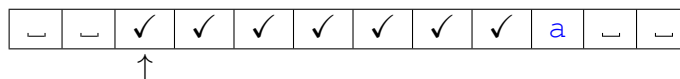
* יש ✓ בתו האחרון \Leftarrow קיבלנו מספר אי-זוגי של אותיות a אחרי חילוק ב- 2 \Leftarrow אין חזקה של 2 אותיות a במילה.

* יש a בתו האחרון \Leftarrow קיבלנו מספר זוגי של אותיות a אחרי חילוק ב- 2 ונמשיך לסבב הבא.

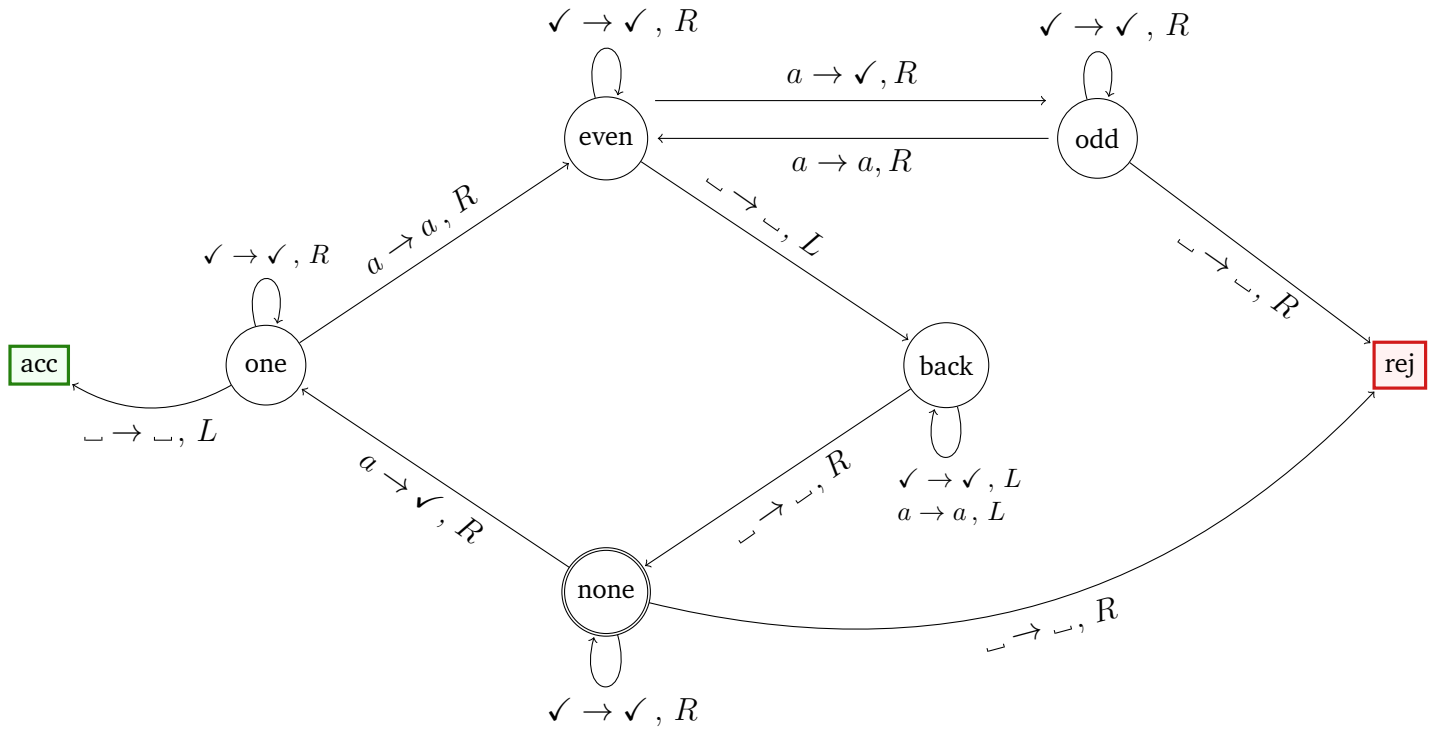
- הראש חוזר לתו הראשון של הקלט.

בסבב האחרון נשאיר רק אות a אחת.

לכן לפי המשפט למעלה מובטח לנו כי המילה מורכבת ממספר אותיות a אשר חזקה של 2.



המכונת טיורינג אשר מקבלת מילים בשפה שעובדת לפי האלגוריתם המתואר למעלה מתואר בתרשים למטה.



המצבים:

- מצב none: מצב התחלתי. עדיין לא קראנו a בסבב סריקה זה.
- מצב one: קראנו a בודד.
- מצב even: קראנו מספר זוגי של a.
- מצב odd: קראנו מספר אי-זוגי של a.
- מצב back: חזרה שלמאלה.

דוגמה 1.7

בדקו אם המילה

aaaa

מתקבלת על ידי המכונת טיורינג בדוגמה 1.6.

פתרון:

␣	none	a	a	a	a	␣
␣	✓	one	a	a	a	␣
␣	✓	a	even	a	a	␣
␣	✓	a	✓	odd	a	␣
␣	✓	a	✓	a	even	␣
␣	✓	a	✓	back	a	␣
␣	✓	a	back	✓	a	␣
␣	✓	back	a	✓	a	␣
␣	back	✓	a	✓	a	␣
back	␣	✓	a	✓	a	␣
␣	none	✓	a	✓	a	␣
␣	✓	none	a	✓	a	␣

⌊	✓	✓	one	✓	a	⌊
⌊	✓	✓	✓	one	a	⌊
⌊	✓	✓	✓	a	even	⌊
⌊	✓	✓	✓	back	a	⌊
⌊	✓	✓	back	✓	a	⌊
⌊	✓	back	✓	✓	a	⌊
⌊	back	✓	✓	✓	a	⌊
back	⌊	✓	✓	✓	a	⌊
⌊	none	✓	✓	✓	a	⌊
⌊	✓	none	✓	✓	a	⌊
⌊	✓	✓	none	✓	a	⌊
⌊	✓	✓	✓	none	a	⌊
⌊	✓	✓	✓	✓	one	⌊
⌊	✓	✓	✓	acc	✓	⌊

μ	q	σ	ν
⌊	none	a	aaa ⌊
⌊ ✓	one	a	aa ⌊
⌊ ✓ a	even	a	a ⌊
⌊ ✓ a ✓	odd	a	⌊
⌊ ✓ a ✓ a	even	⌊	⌊
⌊ ✓ a ✓	back	a	⌊
⌊ ✓ a	back	✓	a ⌊
⌊ ✓	back	a	✓ a ⌊
⌊	back	✓	a ✓ a ⌊
⌊	back	⌊	✓ a ✓ a ⌊
⌊	none	✓	a ✓ a ⌊
⌊ ✓	none	a	✓ a ⌊
⌊ ✓ ✓	one	✓	a ⌊
⌊ ✓ ✓ ✓	one	a	⌊
⌊ ✓ ✓ ✓ a	even	⌊	⌊
⌊ ✓ ✓ ✓	back	a	⌊
⌊ ✓ ✓	back	✓ a	⌊
⌊ ✓	back	✓	✓ a ⌊
⌊	back	✓	✓ ✓ a ⌊
⌊	back	⌊	✓ ✓ ✓ a ⌊
⌊	none	✓	✓ ✓ a ⌊
⌊ ✓	none	✓	✓ a ⌊
⌊ ✓ ✓	none	✓	a ⌊
⌊ ✓ ✓ ✓	none	a	⌊
⌊ ✓ ✓ ✓ ✓	one	⌊	⌊
⌊ ✓ ✓ ✓	acc	✓	⌊

דוגמה 1.8

בדקו אם המילה

aaa

מתקבלת על ידי המכונת טיורינג בדוגמה 1.6.

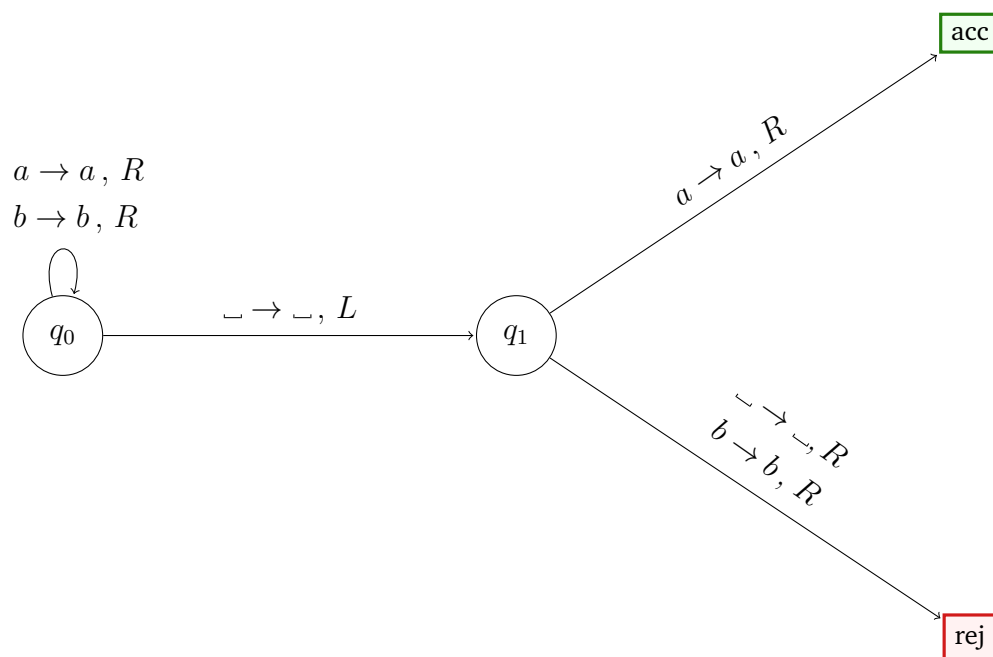
פתרון:

␣	none	a	a	a	␣
␣	✓	one	a	a	␣
␣	✓	a	even	a	␣
␣	✓	a	✓	odd	␣
␣	✓	a	✓	␣	rej

μ	q	σ	ν
␣	none	a	aa ␣
␣ ✓	one	a	a ␣
␣ ✓ a	even	a	␣
␣ ✓ a ✓	odd	␣	␣
␣ ✓ a ✓ ␣	rej	␣	␣

דוגמה 1.9

מהי שפת המכונה:



פתרון:

תיאור מילולי:

• במצב התחלתי q_0 :

* אם אנחנו רואים a , עוברים למשבצת הבאה לימין הראש.

* אם אנחנו רואים b , עוברים למשבצת הבהאה לשמאל הראש.

- ממשיכים כך עד שנגיע לתו רווח, כלומר לסוף המילה, ואז עוברים למשבצת לשמאל הראש, כלומר לתו האחרון של המילה.

* אם אנחנו רואים a , המילה מתקבלת. (ז"א התו האחרון הינו a).

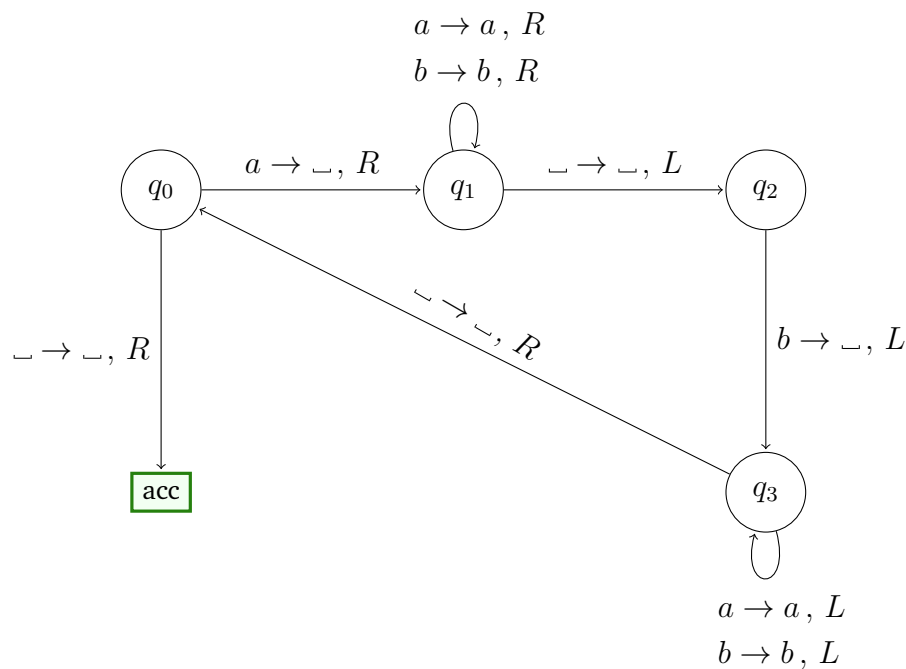
* אם אנחנו רואים b , המילה נדחית. (ז"א התו האחרון הינו b).

* אם אנחנו רואים תו-רווח המילה נדחית. (ז"א המילה הינה ריקה).

תשובה סופית: המכונה מקבלת שפת המילים המסתיימות באות a .

דוגמה 1.10

מהי שפת המכונה:



פתרון:

תיאור מילולי:

- במצב התחלתי q_0 :

- * אם אנחנו רואים b , המילה נדחית.

- * אם אנחנו רואים $_$, המילה מתקבלת.

- * אם אנחנו רואים a , כותבים עליה $_$ ועוברים למשבצת הבאה לימין הראש, והמ"ט עוברת למצב q_1 .

- במצב q_1 אנחנו ראינו a וכתבנו עליה $_$.

- * אם אנחנו רואים במשבצת הבאה a או b , ממשיכים למשבצת הבאה לימין והמ"ט נשארת במצב q_1 .

- * אם אנחנו רואים תו רווח (כלומר הגענו לסוף המילה) הראש z למשבצת השמאלי, כלומר לאות האחרונה של המילה והמ"ט עוברת למצב q_2 .

- במצב q_2 ראינו a בתו הראשון, כתבנו עליה $_$ והראש קורא התו האחרון.

- * אם אנחנו רואים a המילה נדחית.

- * אם אנחנו רואים $_$, המילה נדחית.

- * אם אנחנו רואים b כותבים עליה $_$ והמ"ט עוברת למצב q_3 .

- במצב q_3 קראנו a בתו הראשון ומחקנו אותה, קראנו b ומחקנו אותה.

- הראש z משבצת אחת שמאלה עד שיגיע לתו הראשון ומ"ט חוזרת למצב התחלת q_0 .

- המ"ט באופן איטרטיבי, עוברת על הקלט ובכל מעבר:

* אם יש a בתחילת המילה מורידה אותה ומחליפה אותה עם $_$, אחרת המילה נדחית,

* אם יש b בסופה של המילה מורידה אותה ומחליפה אותה עם $_$, אחרת המילה נדחית.

- אם לאחר מספר מעברים כאלו הסרט ריק, המ"ט מקבלת, וזה יתקיים לכל מילה ורק למילים בשפה

$$\{a^n b^n \mid n \geq 0\}.$$

תשובה סופית: המכונה מקבלת שפת המילים

$$\{a^n b^n \mid n \geq 0\}.$$

דוגמה 1.11

פתרון:

μ	q	σ	ν
$_ _$	q_0	a	aaabbbb $_ _$
$_ _ _$	q_1	a	aabbbb $_ _$
$_ _ _ a$	q_1	a	abbbb $_ _$
$_ _ _ aa$	q_1	a	bbbb $_ _$
$_ _ _ aaa$	q_1	b	bbb $_ _$
$_ _ _ aaab$	q_1	b	bb $_ _$
$_ _ _ aaabb$	q_1	b	b $_ _$
$_ _ _ aaabbb$	q_1	b	$_ _$
$_ _ _ aaabbbb$	q_1	$_$	$_$
$_ _ _ aaabbb$	q_2	b	$_ _$
$_ _ _ aaabb$	q_3	b	$_ _ _$
$_ _ _ aaab$	q_3	b	b $_ _ _$
$_ _ _ aaa$	q_3	b	bb $_ _ _$
$_ _ _ aa$	q_3	a	bbb $_ _ _$
$_ _ _ a$	q_3	a	abbb $_ _ _$
$_ _ _$	q_3	a	aabbb $_ _ _$
$_ _ _$	q_3	$_$	aaabbb $_ _ _$
$_ _ _$	q_0	a	aabbb $_ _ _$
$_ _ _ _$	q_1	a	abbb $_ _ _$
$_ _ _ _ a$	q_1	a	bbb $_ _ _$
$_ _ _ _ aa$	q_1	b	bb $_ _ _$
$_ _ _ _ aab$	q_1	b	b $_ _ _$
$_ _ _ _ aabb$	q_1	b	$_ _ _$
$_ _ _ _ aabbb$	q_1	$_$	$_ _$
$_ _ _ _ aabb$	q_2	b	$_ _ _$
$_ _ _ _ aab$	q_3	b	$_ _ _ _$
$_ _ _ _ aa$	q_3	b	b $_ _ _ _$
$_ _ _ _ a$	q_3	a	bb $_ _ _ _$
$_ _ _ _$	q_3	a	abb $_ _ _ _$

_ _ _	q_3	_	aabb_ _ _ _
_ _ _	q_0	a	abb_ _ _ _
_ _ _	q_1	a	bb_ _ _ _
_ _ _ a	q_1	b	b_ _ _ _
_ _ _ ab	q_1	b	_ _ _ _
_ _ _ abb	q_1	_	_ _ _ _
_ _ _ ab	q_2	b	_ _ _ _
_ _ _ a	q_3	b	_ _ _ _
_ _ _	q_3	a	b_ _ _ _
_ _ _	q_3	_	ab_ _ _ _
_ _ _	q_0	a	b_ _ _ _
_ _ _	q_1	b	_ _ _ _
_ _ _ b	q_1	_	_ _ _ _
_ _ _	q_2	b	_ _ _ _
_ _ _	q_3	_	_ _ _ _
_ _ _	q_0	_	_ _ _ _

הגדרה 1.4 גרירה בצעד אחד

תהי $M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג, ותהי c_1 ו- c_2 קונפיגורציות של M .
נסמן

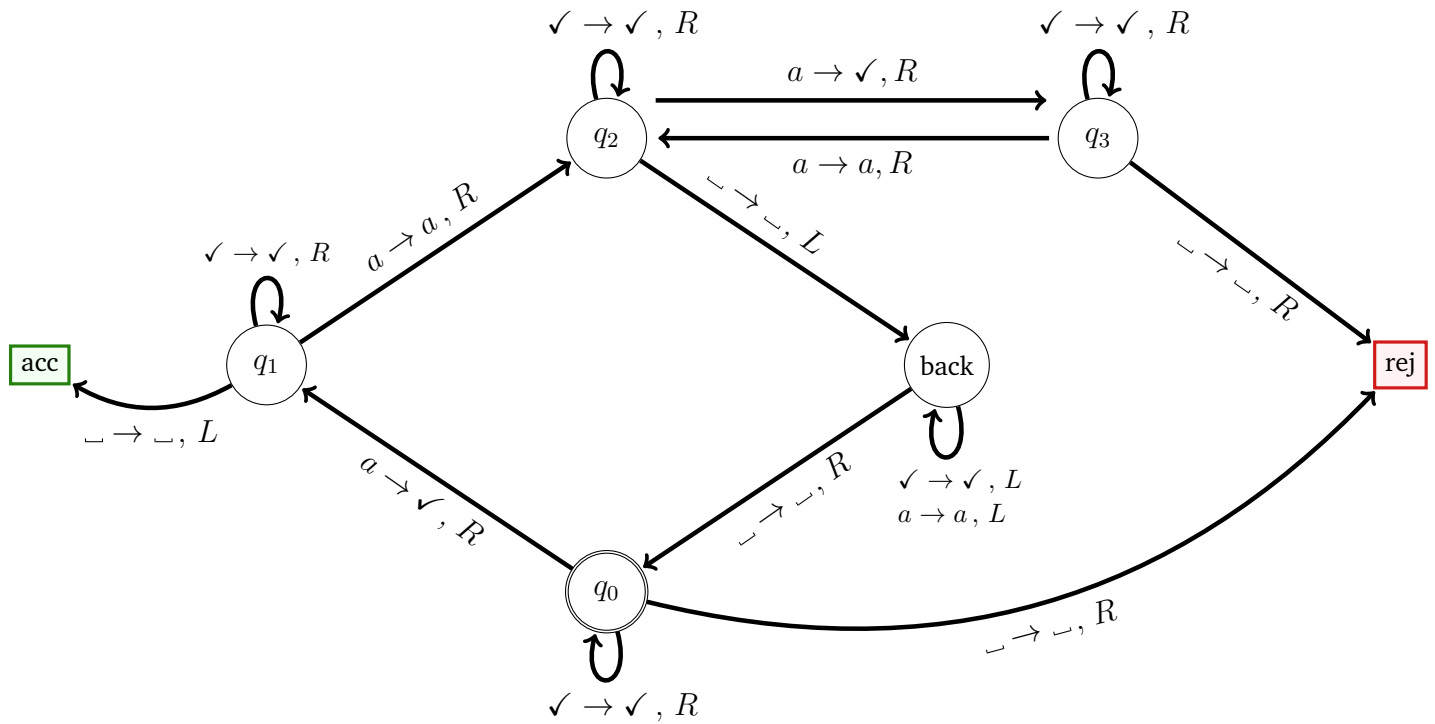
$$c_1 \vdash_M c_2$$

(במילים, c_1 גורר את c_2) אם כשנמצאים ב- c_1 עוברים ל- c_2 בצעד בודד.

דוגמה 1.12 (המשך של דוגמה 1.6)

במכונת טיורינג שמתואר בתרשים דמטה (אשר שווה למ"ט בדוגמה 1.6 רק עם סימנוים שונים למצבים) מתקיים

$$\checkmark q_0 a \checkmark a \vdash_M \checkmark \checkmark q_1 \checkmark a$$



הגדרה 1.5 גרירה בכללי

תהי $M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג, ותהי c_1 ו- c_2 קונפיגורציות של M . נסמן

$$c_1 \vdash_M^* c_2$$

(במילים, c_1 גורר את c_2) אם ניתן לעבור מ- c_1 ל- c_2 ב-0 או יותר צעדים.

דוגמה 1.13 (המשך של דוגמה 1.6)

במכונת טיורינג שמתואר בתרשים דוגמה 1.6 (אשר שווה למ"ט בדוגמה 1.6 רק עם סימנים שונים למצבים) מתקיים

$$\checkmark q_0 a \checkmark a \vdash_M^* \checkmark \checkmark \checkmark q_4 a$$

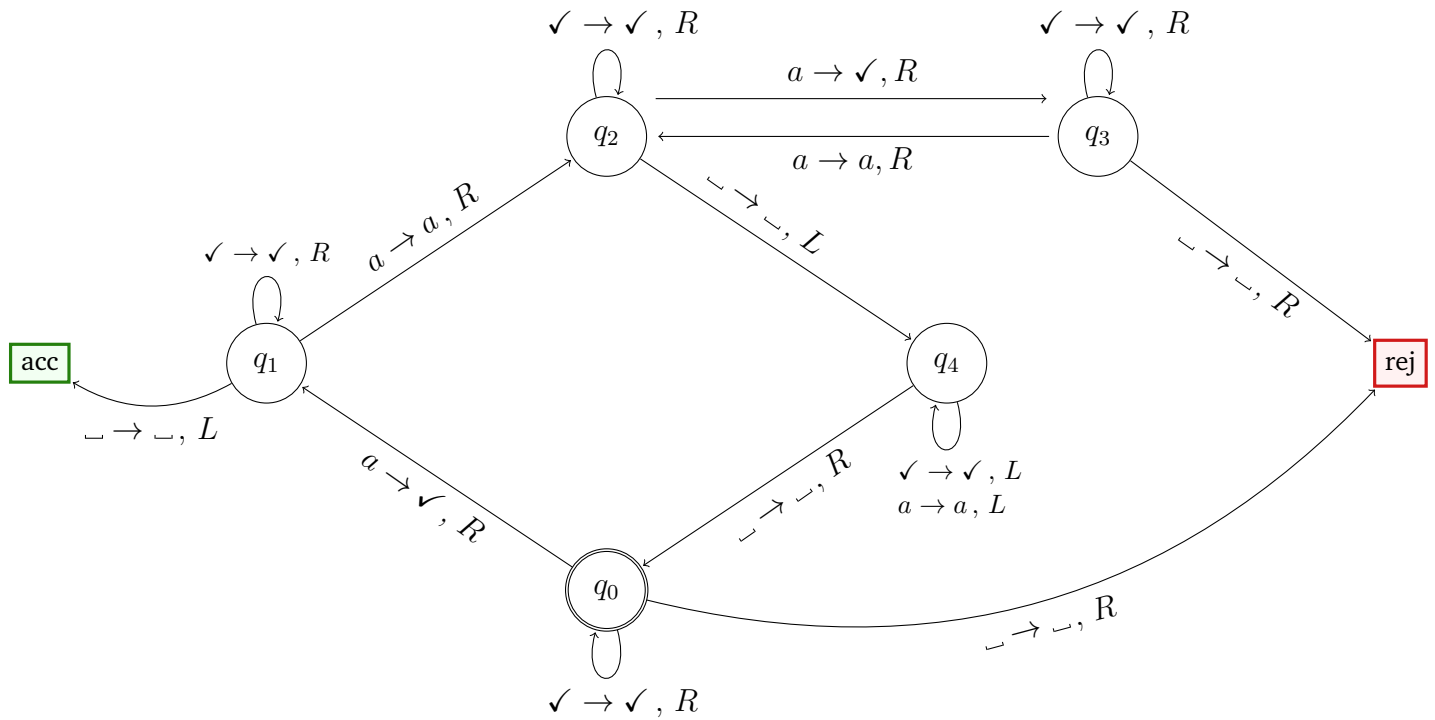
כי

$$\checkmark q_0 a \checkmark a \vdash_M \checkmark \checkmark q_1 \checkmark a$$

$$\vdash_M \checkmark \checkmark \checkmark q_1 a$$

$$\vdash_M \checkmark \checkmark \checkmark a q_2 _$$

$$\vdash_M \checkmark \checkmark \checkmark q_4 a .$$



הגדרה 1.6 קבלה ודחייה של מחרוזת

תהי

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$$

מכונת טיורינג, ו-

$$w \in \Sigma^*$$

מחרוזת. אומרים כי

• M מקבלת את w אם

$$q_0 w \vdash_M^* u \text{ acc } \sigma v$$

עבור $v, u \in \Gamma^*, \sigma \in \Gamma$ כלשהם,

• M דוחה את w אם

$$q_0 w \vdash_M^* u \text{ rej } \sigma v$$

עבור $v, u \in \Gamma^*, \sigma \in \Gamma$ כלשהם.

הגדרה 1.7 הכרעה של שפה

תהי

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$$

מכונת טיורינג, ו-

$$L \subseteq \Sigma^*$$

שפה. אומרים כי M מכריעה את L אם לכל $w \in \Sigma^*$ מתקיים

$$M \Leftarrow w \in L \text{ • } w \text{ מקבלת את } w$$

$$M \Leftarrow w \notin L \text{ • } w \text{ דוחה את } w$$

הגדרה 1.8 קבלה של שפה

תהי

$$M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$$

מכונת טיורינג, ו-

$$L \subseteq \Sigma^*$$

שפה. אומרים כי M מקבלת את L אם לכל $w \in \Sigma^*$ מתקיים• אם $w \in L$ אז M מקבלת את w .• אם $w \notin L$ אז M לא מקבלת את w .

במקרה כזה נכתוב ש-

$$L(M) = L.$$

1.3 טבלת המעברים

דוגמה 1.14

בנו מכונת טיורינג שמכריעה את השפה

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

פתרון:



מצב	סימון בסרט	מצב חדש	כתיבה	תזוזה	תנאי
$q.S$	σ	$q.(S \cup \{\sigma\})$	\checkmark	R	$\sigma \notin S$
$q.S$	σ	$q.S$		R	$\sigma \in S$
$q/\{a, b, c\}$	a, b, c, \checkmark	back		L	
$q.\emptyset$	\perp	acc		R	
back	a, b, c, \checkmark	back		L	
back	\perp	$q.\emptyset$		R	

דוגמה 1.15

בנו מכונת טיורינג שמכריעה את השפה

$$\{x_1 \dots x_k \# y_1 \dots y_k \# z_1 \dots z_k \mid x_i, y_i, z_i \in \{0, \dots, 3\}, \forall i, x_i \geq z_i \geq y_i\}$$

פתרון:

$$L = \{X_1 X_2 \# Y_1 Y_2 \# Z_1 Z_2 \mid X_i, Y_i, Z_i \in \{0,1,2,3\} \forall i \mid X_1 \geq X_2, Y_1 \geq Y_2\}$$



תנאי	תזוזה	כתיבה	מצב חדש	סימון בסרט	מצב
	R	✓	$X\sigma^*$	σ	X^{**}
	R	✓	X^{**}	✓	X^{**}
	R		$X\sigma^*$	$0, 1, \dots, 9, \checkmark$	$X\sigma^*$
	R		$Y\tau^*$	#	$X\tau^*$
	R		$Y\tau\sigma$	σ	$Y\tau^*$
	R		$Y\tau^*$	✓	$Y\tau^*$
	R		$Y\tau\sigma$	$0, 1, \dots, 9, \checkmark$	$Y\tau\sigma$
	R		$Z\tau_1\tau_2$	#	$Y\tau_1\tau_2$
	R		$Z\tau_1\tau_2$	✓	$Z\tau_1\tau_2$
	L	✓	back	σ	$Z\tau_1\tau_2$
	R		acc	\sqsubset	Z^{**}
	L		back	$0, 1, \dots, 9, \checkmark$	back
	R		X^{**}	\sqsubset	back

1.4 חישוב פונקציות

דוגמה 1.16 חיבור אונרי

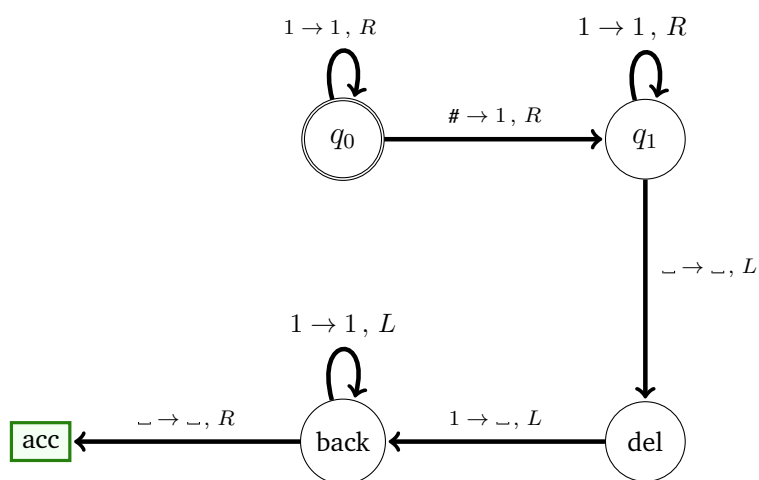
בנו מכונת טיורינג אשר מקבלת את הקלט

$$1^i \# 1^j$$

ומחזירה את פלט

$$1^{i+j}.$$

פתרון:



דוגמה 1.17 כפל אונרי

בנו מכונת טיורינג אשר מקבלת את הקלט

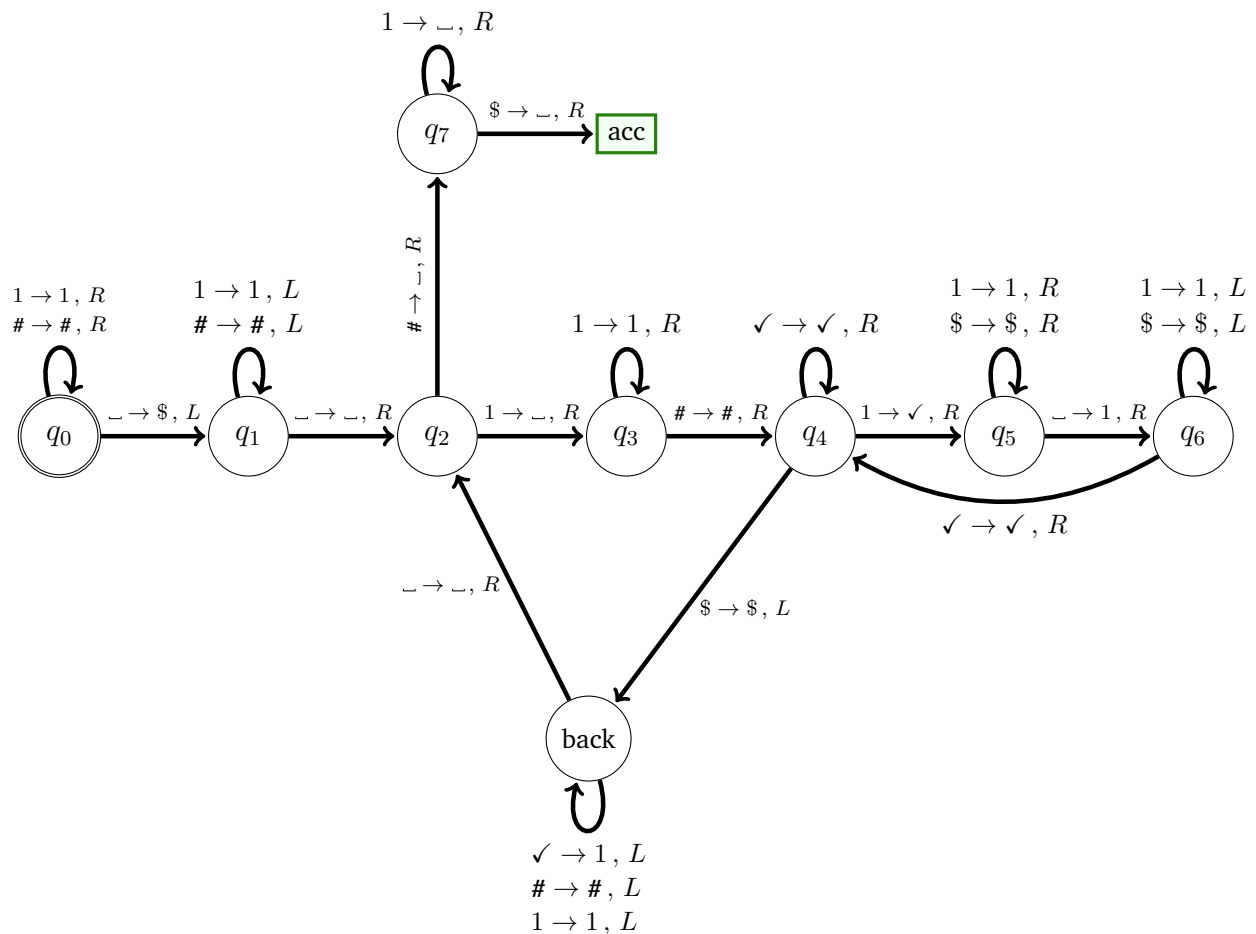
$$1^i \# 1^j$$

ומחזירה את פלט

$$1^{i \cdot j}.$$

פתרון:

- לדוגמה, נניח שהקלט הוא 2 כפול 2.
הקלט הוא 11#11.
- נרצה להבדיל בין הקלט לבין הפלט.
לכן בתחילת הריצה, נתקדם ימינה עד סוף הקלט ונוסיף שם את התו \$.
- לאחר מכן נחזור לתחילת הקלט.
על כל אות 1 במילה השמאלית נעתיק את המילה הימינית לאחר סימן ה-\$.
- לאחר מכן נשאיר רק את התווים שלאחר סימן ה-\$. כלומר, נמחק את כל מה שאינו פלט.



μ	q	σ	ν
$_$	q_0	1	1#11 $_$
$_11\#11$	q_1	$_$	$_$
$_11\#11$	q_1	\$	$_$
$_$	q_1	$_$	11#11\$
$_$	q_2	1	1#11\$
$_ _$	q_3	1	#11\$
$_ _1\#$	q_4	1	1\$
$_ _1\#\checkmark$	q_5	1	\$

1#✓1\$	q_5		
1#✓1\$1	q_6		
1#	q_6	✓	1\$1
1#✓	q_4	1	\$1
1#✓✓	q_5	\$	1
1#✓✓\$1	q_5		
1#✓✓\$11	q_6		
1#✓	q_6	✓	\$11
1#✓✓	q_4	\$	11
1#✓	back	✓	\$11
	back		1#11\$11
	q_2	1	#11\$11
	q_3	#	11\$11
#	q_4	1	1\$11
#✓	q_5	1	\$11
#✓1\$11	q_5		
#✓1\$111	q_6		
#	q_6	✓	1\$111
#✓	q_4	1	\$111
#✓✓	q_5	\$	111
#✓✓\$111	q_5		
#✓✓\$1111	q_6		
#✓	q_4	✓	\$1111
#✓✓	q_4	\$	1111
#✓	back	✓\$	1111
	back		#11\$1111
	q_2	#	11\$1111
	q_7	1	1\$1111
	q_7	\$	1111
	acc	1	111