

# חישוביות

---

מרצה – יונתן אומן(שעות קבלה יום א' 14<sup>00</sup>-13<sup>00</sup>, [aumann@cs.biu.ac.il](mailto:aumann@cs.biu.ac.il))

ספר – Introduction to the Theory of Computation/Sipser, (פרקים 4-6 רלוונטיים)

אתר הקורס – [www.yairdombb.com/comp2011](http://www.yairdombb.com/comp2011)

הקורס הזה מסתכל על המגבלות של מדעי המחשב – מה אפשר לחשב ומה אי אפשר לחשב.  
השאלה הבסיסית היא

- אילו בעיות ניתנות לפתרון בעזרת מחשב?
- האם ישנן בעיות שלא ניתנות לפתרון בעזרת מחשב?
- אילו בעיות "קשה מאוד" לפתור בעזרת מחשב?

# מושג ה"בעיה"

## הגדרה

א"ב (מסומן בד"כ ב $\Sigma$ ) הינה קבוצה סופית של תווים או אותיות. דוגמאות:

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

$$\Sigma = \{a, b, c, d, \dots, z\}$$

## הגדרה

יהי  $\Sigma$  אזי מחרוזת/מילה מעל  $\Sigma$  הינה סדרה סופית של תווים מתוך  $\Sigma$ .

בד"כ נסמן מילים באותיות קטנות מסוף הא"ב:  $\Sigma = \{a, b\}$ ,  $w = abbaba$ .

$\epsilon$  – המחרוזת הריקה.

תהי  $w$  מילה. נסמן  $|w|$  את מספר התווים ב $w$ , האורך של  $w$

$$|abaa| = 4$$

$$|\epsilon| = 0$$

עבור א"ב  $\Sigma$  נסמן ב $\Sigma^*$  את קבוצת כל המילים מעל  $\Sigma$ .  $\emptyset^* = \{\epsilon\}$ . נשים  $\epsilon \neq \emptyset$

## דוגמה – האם מספר מתחלק ב3?

$$\Sigma = \{0,1,2,3,4,5,6,7,8,9\}$$

$\Sigma^*$  מכיל את כל המחרוזות שאפשר לשלוח – כלומר את כל המספרים.

נגדיר  $L \subseteq \Sigma^*$  – כל המחרוזות שעליהן צריך להחזיר "כן"

## הגדרה

תהי  $\Sigma$  א"ב. שפה מעל  $\Sigma$  הינה תת קבוצה  $L \subseteq \Sigma^*$

## דוגמאות לשפות

$$L = \{a, ab, aba\}$$

$$L = \{w \in \{a, b\}^* \mid w \text{ מתחיל ב} a\}$$

$$L = \{w \in \{a, b\}^* \mid |w| \text{ זוגי}\}$$

## פעולות על שפות

איחוד, חיתוך, הפרש, משלים, מכפלה קרטזית...

## הגדרה

תהיה  $v, w$  מחרוזות. השרשור של  $v$  עם  $w$   $vw = wv$  הינה המילה שמתקבלת ע"י כתיבת

האותיות של  $v$  ולאחריהן אותיות של  $w$

$$v = aba, w = ba$$

$$vw = ababa, wv = baaba, w\epsilon = w$$

## פעולת החזקה

תהי  $w$  מילה עבור  $k$  טבעי. נגדיר  $w^k$ :

$$w^0 = \epsilon, w^{k+1} = w^k w$$

## הגדרה

תהיינה  $L_1, L_2$  שפות.

$$L_1 \circ L_2 = \{v \circ w \mid v \in L_1, w \in L_2\}$$

## דוגמה

$$L_1 = \{w \in \{a, b\}^* \mid a \text{ מתחיל עם } w\}$$

$$L_2 = \{w \in \{a, b\}^* \mid b \text{ מסתיים עם } w\}$$

$L_1 \circ L_2$  – מחזורות שמתחילות ב  $a$  ומסתיימות ב  $b$ .

$L_2 \circ L_1$  – מחזורות שמכילות  $ba$

## הגדרה

תהי  $L$  שפה,  $k$  טבעי. נגדיר  $L^k$ :

$$L^0 = \{\epsilon\}, L^{k+1} = L^k \circ L$$

## הגדרה

תהי  $L$  שפה. הסגור הקליני (Kleene) של  $L$  מוגדר  $L^*$  =  $\bigcup_{k=0}^{\infty} L^k$

# הגדרת המושג "מחשב"

אנחנו רוצים הגדרה שתהיה כוללת – כדי שתכלול את כל המחשבים, בעבר ובעתיד – ושתהיה פשוטה, כדי שההוכחות שלנו יהיו פשוטות.

## מה זה מחשב?

מחשב (או כלי חישובי) הוא לוגיקה מסויימת שמקבלת קלטים, מבצעת עליהם סדרה של פעולות, ומחזירה פלט. אנחנו רוצים לוגיקה סופית, שתתאים לכל הקלטים האפשריים.

צריך גם להיות זיכרון, שמאפשר ללוגיקה לשמור מידע. צריך גם רכיב שיודע לקרוא את הקלט, ורכיב שמאפשר אינטרקציה עם הזיכרון.

כדי לפשט את המודל, הקלט יהיה בתוך הזיכרון. הזיכרון יהיה מערך חד מימדי אינסופי (בכיוון ימין. יש לו קצה שמאלי) שנקרא "סרט". הקלט יהיה בקצה השמאלי של הסרט. שאר הסרט ריק (נסמן עם התו '\_').

יש ראש שקורא/כותב לסרט. הראש יכול ללכת שמאלה או ימינה, צעד אחד בכל פעם:

## פעולות על הסרט

1. תזוזה ימינה (R)
2. תזוזה שמאלה (L)
3. כתיבה של תו במשבצת הנוכחית.

למחשב צריכים להיות מצבים, שכל מצב מגיב שונה לקלט. ה"לוגיקה" (או "מוח" או "בקר") היא אוסף סופי של פקודות מהצורה: אם נמצאים במצב "q", ורואים בסרט את "x", אזי לבצע פעולה y ולעבור למצב p.

אם אנחנו רוצים תשובה של כן או לא, אז אנחנו צריכים את המצבים:

- $q_0$  – מצב התחלתי
- $q_{accept}$  – מצב "כן"
- $q_{reject}$  – מצב "לא".

## דוגמה

$$\Sigma = \{a, b\}, L = \left\{ w \in \{a, b\}^* \mid \begin{array}{l} w \text{ את מכיל} \\ \text{כתת מחרוזת } bb \end{array} \right\}$$

רוצים לדעת אם הקלט הוא מילה בשפה. הקלט הוא

|   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|
| a | b | a | a | _ | _ | _ |
|---|---|---|---|---|---|---|

| מצב   | תו בסרט | פעולה  | מצב חדש      |
|-------|---------|--------|--------------|
| $q_0$ | a       | R      | $q_0$        |
| $q_0$ | b       | R      | $q_1$        |
| $q_0$ | _       | b(סתם) | $q_{select}$ |
| $q_1$ | a       | R      | $q_0$        |
| $q_1$ | b       | L(סתם) | $q_{accept}$ |
| $q_1$ | _       | L(סתם) | $q_{reject}$ |

## הגדרה

מכונת טיורינג (במודל T) הינה שביעיה  $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{ac}, q_{rej})$  כאשר:

- $Q$  – קבוצת מצבים סופית
- $\Sigma$  – א"ב קלט (הקלטים הינם מחרוזות מעל  $\Sigma$ ).  $\_ \notin \Sigma$
- $\Gamma$  – א"ב סרט.  $\Sigma \subset \Gamma$
- $q_0 \in Q$  – מצב התחלתי
- $q_{ac} \in Q$  – מצב מאשר (לפעמים נקרא מצב מקבל)
- $q_{rej} \in Q$  – מצב דוחה
- $\delta$  – פונקציית מעברים:  $\delta: (Q - \{q_{ac}, q_{rej}\}) \times \Gamma \rightarrow Q \times (\Gamma \cup \{R, L\})$  – הפונקציה מקבלת מצב ואות, ומחזירה מצב (שאליו עוברים) ואות (שאותה כותבים) או  $L/R$  (הזזת הראש על הסרט). הפונקציה לא פועלת על מצב האישור או מצב הדחיה – כי אז מסתיימת התוכנית.

## דוגמה

נכתוב תוכנית למכונת טיורינג שמזיזה את הקלט משבצת אחת ימינה

| מצב   | תו בסרט | מצב חדש  | פעולה |
|-------|---------|----------|-------|
| $q_0$ | $a$     | $r_a$    | –     |
| $q_0$ | $b$     | $r_b$    | –     |
| $q_0$ | –       | $q_{ac}$ | –     |
| $r_a$ | –       | $w_a$    | $R$   |
| $r_b$ | –       | $w_b$    | $R$   |
| $w_a$ | $b$     | $r_b$    | $a$   |
| $w_a$ | $a$     | $r_a$    | $a$   |
| $w_b$ | $a$     | $r_a$    | $b$   |
| $w_b$ | $b$     | $r_b$    | $b$   |
| $w_a$ | –       | $q_{ac}$ | $a$   |
| $w_b$ | –       | $q_{ac}$ | $b$   |
| $r_a$ | $a$     | $w_a$    | $R$   |
| $r_a$ | $b$     | $w_a$    | $R$   |
| $r_b$ | $a$     | $w_b$    | $R$   |
| $r_b$ | $b$     | $w_b$    | $R$   |

## הגדרה

תהי  $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{rej})$  מ"ט. קונפיגורציה של  $M$  הינה רביעיה  
 $(q, l, x, r) \in Q \times \Gamma^* \times \Gamma \times \Gamma^*$

- $q$  – המצב
- $l$  – תוכן הסרט משמאל לראש
- $x$  – תוכן הסרט במיקום הראש
- $r$  – תוכן הסרט מימין לראש, ולאחריו אין סוף "

נכתוב את הקונפיגורציה  $(q, l, x, r)$  גם כמחרוזת בודדת:  $lqxr \in (\Gamma \cup Q)^*$ . לדוגמה  $(w_a, ab, a, a)$   
אפשר לכתוב בתור  $abw_aaa$