12. חישוב יעיל

עד עתה התעניינו בשאלה "אילו פונקציות ניתנות לחישוב?". אולם, עצם העובדה שפונקציה ניתנת לחישוב, הינה כמעט חסרת ערך אם זמן החישוב ארוך מכדי שניתן יהיה לבצעו באופן ממשי. החשיבות של מושג **החישוב היעיל**, אינה נופלת מחשיבותו של מושג החישוב עצמו.

מהו חישוב יעיל? המדד הראשוני ליעילות של חישוב הוא "משך הזמן" הדרוש לביצועו, או באופן מדויק, מספר הצעדים בו. מדד זה הינו כמותי מטבעו. איפיון חישובים יעילים יתבסס לכן על איפיון של מספר צעדי החישוב. מספר "קטן" של צעדים יאפיין חישוב יעיל, ואילו מספר "גדול" יאפיין חישוב שאינו יעיל. אולם מהם המושגים של "קטן" ו"גדול" במקרה זה?
הגסות שנהגנו עד עתה, בזיהוי של "קטן" עם סופי ו"גדול" עם אינסופי, לא תוכל לשמש אותנו יותר (גסות זו, היא הבסיס לאפיון של חישוב עוצר כחישוב שמספר צעדיו סופי, כנגד חישוב שאינו מסתיים שאינו אלא "חישוב אינסופי").
כאשר מגדירים "מספר קטן של צעדים" אסור שמספר זה יהיה קטן מדי, אחרת הגדרת חישוב יעיל תאפשר רק חישובים כמעט טריביאליים. בפרט, מספר הצעדים צריך לאפשר לפחות את קריאת הקלט, אחרת לא נוכל לבצע חישובים התלויים בקלט כולו (למשל, ספירת מספר ה 1-ים בקלט). מכאן עולה, כי מספר הצעדים של חישוב יעיל אינו יכול להיות ערך אבסולוטי, אלא חייב להיות פונקציה של אורך הקלט. לפיכך, איפיון חישובים יעילים הוא איפיון של פונקציות המתארות את "זמן החישוב" כפונקציה של אורך הקלט.

הגדרה 12.1: **מכונת טיורינג** M, **עוצרת על קלט** x **ב**–i **צעדים** אם ורק אם מתקיים $q_0 x \overset{i}{\underset{M}{\vdash}} \alpha p \beta$, כאשר הקונפיגורציה $\alpha p \beta$ היא קונפיגורציה סופית.

הגדרה 12.2: **סיבוכיות הזמן** של מכונת טיורינג M, היא פונקציה $t_M : \{0,1\}^* \to N$, המוגדרת על אותם קלטים x, עליהם M עוצרת, כך ש- $t_M(x)$, היא מספר הצעדים של המכונה M על הקלט x.

קונוונציה: בתרגילים (ובהגדרות) הבאים, תוצגנה פונקציות ממחרוזות לטבעיים ו/או מטבעיים לטבעיים, ותדרשנה מכונות טיורינג לחישוב פונקציות אלו. בכל מקרה, אלא אם כן יאמר במפורש אחרת, יש להניח כי הטבעיים ניתנים (כקלט וצריכים להיות מוחזרים כפלט) ביצוג הבינארי הסטנדרטי.

תרגיל 12.1: הוכח כי לכל מכונת טיורינג M, סיבוכיות הזמן של M (כלומר הפונקציה t_M) ניתנת לחישוב. מה סיבוכיות הזמן של המכונה שהצעת לחישוב הפונקציה t_M?

הגדרה 12.3: **חסם מלעיל על סיבוכיות** (בקיצור, חסם סיבוכיות) של מכונת טיורינג M, היא פונקציה,

$T_M : N \to N$, המוגדרת על כל $n \in N$ ומקיימת לכל x עליו M עוצרת, $t_M(x) \le T_M(|x|)$ (זכור, $|x|$ הוא האורך של המחרוזת x).

$T_M(n)$ הינו חסם עליון על סיבוכיות הזמן של המכונה M לגבי קלטים באורך n. שים לב, כי אין דרישה שלכל $x \in \{0,1\}^n$, יתקיים $t_M(x) = T_M(|x|)$ ואפילו אין דרישה שקיים x אחד כזה. כלומר, $T_M(|x|)$ יכול להיות גדול ממש מ- $t_M(x)$ בכל מקום. כל מה שנדרש הוא שלא יהיה קטן ממנו. שים לב, שחסם הסיבוכיות, $T_M(n)$, חוסם את מספר הצעדים של M על הקלט הגרוע ביותר מאורך n (כלומר, שריצת M עליו היא הארוכה ביותר). זהו מדד של מקרה גרוע ביותר (worst case).

תרגיל 12.2: הראה כי אם M עוצרת על כל קלט, אז קיימת פונקציה T_M ניתנת לחישוב המהווה חסם מלעיל על סיבוכיות M (הדגש את חשיבות התנאי). תן חסם סיבוכיות למכונה המחשבת את הפונקציה T_M. שים לב כי במקרים מסויימים ניתן לחשב חסם סיבוכיות באופן יעיל יותר. תן דוגמאות למקרים כאלו.

הערה למתקדמים: ניתן גם להגדיר **חסם מלרע** על סיבוכיות של מכונת טיורינג, אך פונקציה זו מעניינת פחות במסגרת קורס זה. **חסם מלרע על סיבוכיות** של מכונת טיורינג, M, הוא פונקציה, $L_M : N \to N$, המקיימת, לכל x עליו M עוצרת, $L_M(|x|) \le t_M(x)$.

תרגיל למתקדמים: תהי M, מכונת טיורינג העוצרת תמיד. הראה כי קיימת מכונת טיורינג, M', השקולה למכונה M, כך ש- M' הינה בעלת חסם מלעיל על סיבוכיות השווה לחסם מלרע על הסיבוכיות (רמז: המכונה M', תרוץ מספר צעדים זהה על קלטים מאותו אורך).

אם נחזור לדיון האינטואיטיבי, הרי שהסכמנו שחישובים הניתנים לביצוע ע"י מכונת טיורינג בעלת חסם סיבוכיות לינארי (כלומר, $T_M(n) = O(n)$) הם ודאי חישובים יעילים. שים לב, שחישובים בעלי חסם סיבוכיות תת-לינארי, אינם יכולים להיות תלויים בכל הקלט (כי אינם מספיקים לקרוא את כולו) וודאי שאינם מעניינים. האם כדאי לזהות את החישובים היעילים עם חישובים הניתנים לביצוע בסיבוכיות לינארית? התשובה לכך שלילית ממספר סיבות:

(1) מספר גדול של אלגוריתמים שנהוג וסביר לזהותם כיעילים, אינם בעלי חסם סיבוכיות לינארי באורך הקלט (אלא פולינומיאלי בו). למשל, הסיבוכיות של מיון בכל מודל חישוב סביר הינה לפחות $n \log n$, הסיבוכיות של אלגוריתם למציאת זרימה ברשת בעלת n צמתים ו- m קשתות ($m \le \binom{n}{2}$) הינה (פחות או יותר) $O(n^3)$.

(2) מושג הסיבוכיות הלינארית רגיש מדי לפרטי מודל החישוב. זמן לינארי במכונת טיורינג בעלת סרט יחיד, אינו זהה לזמן לינארי במכונת טיורינג בעלת שני סרטים. בפרט, השפה $\{ww : w \in \{0,1\}^*\}$, אינה ניתנת לזיהוי בזמן לינארי על גבי מכונת טיורינג בעלת סרט יחיד (תרגיל למתקדמים מאד: הוכח טענה זו!) בעוד שהיא

ניתנת לזיהוי בזמן לינארי ע"י מכונת טיורינג בעלת שני סרטים.

לנוכח התנגדויות (1) ו- (2), עולה באופן טבעי הצעה לזהות את החישובים היעילים עם חישובים הניתנים לביצוע בסיבוכיות פולינומיאלית.

הגדרה 12.4: למכונת טיורינג, M, יש **סיבוכיות פולינומיאלית** (לפעמים נאמר כי M מכונה פולינומיאלית) אם קיים פולינום, $P(n)$, המהווה חסם מלעיל על סיבוכיות הזמן של M.

להצעה זו ישנם שלושה יתרונות חשובים על ההצעה הקודמת (של סיבוכיות לינארית):

(1) כמעט כל האלגוריתמים, שנהוג וסביר לזהותם כיעילים, הינם פולינומיאלים, כלומר, מתאימים למכונות טיורינג בעלות חסם סיבוכיות פולינומיאלי (על יוצאים מהכלל נרחיב את הדיבור בהמשך).

(2) מושג הסיבוכיות הפולינומיאלית אינו רגיש לפרטי מודל החישוב. בפרט, זמן פולינומיאלי במכונת טיורינג בעלת סרט יחיד, זהה לזמן פולינומיאלי במכונת טיורינג בעלת שני סרטים, או לזמן פולינומיאלי של כל מודל סביר וכללי של חישוב (ודא כי כל הוכחות של שקילות בין המודלים הסבירים של חישוב, מפרקים 4 ו- 5, משמרים את מושג הסיבוכיות הפולינומיאלית). טענה זו ידועה בשם **התזה של אדמונס** (אנלוגית לתזה של צ'רץ'): "בכל מודל סביר וכללי של חישוב, הפונקציות הניתנות לחישוב בזמן פולינומיאלי, הינן אלו הניתנות לחישוב בזמן פולינומיאלי ע"י מכונת טיורינג".

(3) תכונה חשובה של חישובים בעלי סיבוכיות חישוב פולינומיאלית, היא הסגירות שלהם להרכבות של אלגוריתמים. בפרט (נראה בהמשך) קיום רדוקציה פולינומיאלית של בעיה א' לבעיה ב' והיות בעיה ב' בעלת סיבוכיות פולינומיאלית, גורר קיום מכונת טיורינג פולינומיאלית הפותרת את בעיה א'. במילים אחרות, קיום אלגוריתם יעיל (פולינומיאלי) המשתמש בסברוטינה שזמן החישוב שלה אינו נחשב (כלומר, לא נלקח בחשבון בחישוב סיבוכיות הזמן) גורר קיום אלגוריתם יעיל (פולינומיאלי) אם הסברוטינה ניתנת למימוש באופן יעיל (זמן פולינומיאלי).

הגדרה 12.5: נסמן ב- **POLY** את קבוצת הפונקציות הניתנות לחישוב בסיבוכיות זמן פולינומיאלית ע"י מכונות טיורינג.

התזה של אדמונס (ניסוח מפורש): בכל מודל סביר וכללי של חישוב ולכל הגדרה סבירה של סיבוכיות זמן במודל זה, הפונקציות הניתנות לחישוב בזמן פולינומיאלי, הן בדיוק הפונקציות בקבוצה **POLY**.

תרגיל 12.3: הוכח את הטענה לגבי המודל של גישה ישירה והמודל של אוטומט תאים אינסופי. הגדר סיבוכיות זמן במודלים אלו (שים לב, כי הגדרה לא סבירה של סיבוכיות זמן לא תקיים את התזה של אדמונס).

לפני שנמשיך בפיתוח התיאוריה של חישובים פולינומיאלים, נתייחס לשתי התנגדויות אינטליגנטיות לזיהוי של חישובים יעילים עם חישובים פולינומיאלים. התנגדויות אלו הפוכות בגישתן:

(1) <u>חישובים פולינומיאלים מכילים הרבה מדי:</u> חישוב באורך של n^{1000} צעדים, על קלט באורך n, אינו ראוי להחשב כחישוב יעיל. הבעיה עם התנגדות זו, שהיא אינה מציעה אלטרנטיבה סבירה אחרת. זיהוי חישובים יעילים עם חישובים בסיבוכיות זמן לינארית, כבר נפסל. הצעה לזהות חישובים יעילים עם חישובים בזמן ריבועי ($O(n^2)$ צעדים, על קלט באורך n) הינה שרירותית וסובלת מתלות במודל הספציפי של החישוב (כמו זו של סיבוכיות זמן לינארית) וכן אינה סגורה תחת רדוקציות ושאר הרכבות אלגוריתמיות. כלומר, אלגוריתם יעיל (זמן ריבועי) אשר משתמש בסברוטינה שזמן החישוב שלה אינו נחשב, אינו גורר קיום אלגוריתם יעיל (זמן ריבועי) כאשר הסברוטינה ניתנת למימוש באופן יעיל (זמן ריבועי). אם נחזור להתנגדות עצמה, הרי שלהגנתנו נוכל לאמר כי לאותן בעיות טבעיות, הנמצאות במחלקה **POLY**, קיימים אלגוריתמים יעילים בפועל. יתר על כן, הניסיון מראה כי לאחר שנמצא אלגוריתם פולינומיאלי לבעיה טבעית, לא עובר זמן רב עד שנמצא לה אלגוריתם יעיל ממש (כלומר אלגוריתם שסיבוכיות הזמן שלו פולינומיאלית ודרגת הפולינום, החוסם את זמן הריצה שלו, היא "נמוכה").

(2) <u>חישובים פולינומיאלים מכילים פחות מדי:</u> ישנו מספר (אמנם קטן) של אלגוריתמים יעילים אשר אין להם חסם סיבוכיות פולינומיאלי, ובכל זאת בפועל הם רצים טוב. הסיבה לתופעה זו, היא שלאלגוריתמים אלו יש למעשה סיבוכיות זמן פולינומיאלית על אותם קלטים שמעניינים (כלומר מופיעים) בפועל, למרות שקיימים קלטים (אשר אינם מופיעים כמעט בפועל) שעליהם האלגוריתם רץ זמן רב מדי. התשובה הנאותה להתנגדות זו, היא פיתוח תאוריה של **סיבוכיות זמן ממוצעת**. נתייחס בקצרה לתיאוריה זו, לתוצעותיה ולמשמעותן בפרק 17.

<u>תרגיל 12.4</u> (האצה לינארית linear speed-up): הראה כי לכל מכונת טיורינג M, בעלת מספר $k>1$ של סרטים ולכל קבוע $c>1$, קיימת מכונת טיורינג M', שקולה (ובעלת k סרטים) כך שלכל קלט x מתקיים:

$$t_{M'}(x) \le \frac{t_M(x)}{c} + O(|x| + |f_M(x)|)$$

כאשר f_M הינה הפונקציה המחושבת ע"י M. לגבי מכונת טיורינג בעלת סרט יחיד, זמן הריצה של המכונה M' השקולה (בעלת סרט יחיד) יקיים:

$$t_{M'}(x) \le \frac{t_M(x)}{c} + O(|x|^2 + |f_M(x)|^2).$$

<u>הדרכה:</u> המכונה M', תעבוד עם א"ב עבודה אשר יקודד $6c$ תוים של א"ב העבודה של המכונה M. קידוד זה יאפשר ל- M' לבצע סימולציה של $6c$ צעדים של המכונה M ב-6 צעדים. במהלך $6c$ צעדים כאלו, המכונה M זזה לכל היותר $6c$ תאים ממיקומה המקורי. כל אות בא"ב המכונה M', יהוה דבוקה של $6c$ אותיות בא"ב המכונה M. לפיכך, סימולציה של $6c$ צעדים של M, יכולה להעשות ע"י קריאת התוכן של התא עליו נמצא הראש הקורא ושני

התאים הסמוכים לו, ושינוי ערכם בהתאם. תוספת הסיבוכיות נובעת מן התרגום ההתחלתי והסופי מא"ב המכונה M, לא"ב המכונה M' ולהיפך.

<u>הערה:</u> כל עוד לא התעניינו בחישוב יעיל, לא היתה חשיבות לקונוונציה מתחילת הפרק בדבר היצוג הבינארי של המספרים הטבעיים. אולם, ברגע שאנו מתעניינים בחישוב יעיל ישנה חשיבות לשאלות: "האם הקלט נתון ביצוג בינארי או אונרי?" וכן "באיזה יצוג יש להחזיר את הפלט?". בפרט, כתיבת פלט ביצוג אונרי דורשת זמן רב יותר מאשר כתיבתו ביצוג בינארי. כמו כן אופן הייצוג של הקלט עלול לשנות את החסם על זמן הריצה שהוא פונקציה של אורך הקלט. לדוגמא, זמן הריצה של האלגוריתם הטריביאלי לפרוק מספרים לגורמים ראשוניים (ע"י נסיון חלוקה בכל המספרים הקטנים מהקלט) הינו פולינומיאלי אם המספר נתון ביצוג אונרי, אולם הוא אקספוננציאלי אם המספר נתון ביצוג בינארי.