

תוכן העניינים

8	1 מכונות טיורינג
9	2 וריאציות של מכונות טיורינג
10	3 התזה של צ'רצ'-טיורינג
14	4 אי-כריעות
18	5 סיבוכיות זמן
21	6 נוסחאות נוספות

1 מכונות טיורינג

הגדרה 1: מכונת טיורינג

מכונת טיורינג (מ"ט) היא שבעיה .

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

$\Sigma \neq \emptyset$ א"ב קלט סופי

$\Sigma \subseteq \Gamma$, $_\in \Gamma$ א"ב סרט סופי

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

מצב הinitialי q_0

מצב מקבל acc

מצב דוחה rej

הגדרה 2: קונפיגורציה

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

קונפיגורציה של M הינה מחרוזת

$$uq\sigma v , \quad u, v \in \Gamma^*, \quad \sigma \in \Gamma, \quad q \in Q .$$

משמעות:

q מצב המכונה,

σ הסימן במקומות הראש

u תוכן הסרט משמאלי לראש,

v תוכן הסרט מימין לראש.

הגדרה 3: גירה

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

נסמן

$$c_1 \vdash_M c_2$$

(במילים, c_1 גורר את c_2) אם כמפורטים ב- c_1 עוברים ל- c_2 בצעד אחד.

נסמן

$$c_1 \vdash_M^* c_2$$

אם ניתן לעبور מ- c_1 ל- c_2 ב- 0 או יותר צעדים.

הגדרה 4: קבלה ודחיה של מחרוזות

תהי $M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג, ו- $w \in \Sigma^*$ מחרוזת.

נאמר כי:

- M מקבלת את w אם $q_0 w \vdash_M^* u \text{ acc } \sigma v$

- M דוחה את w אם $q_0 w \vdash_M^* u \text{ rej } \sigma v$
כאשר $\Gamma \in u, v \in \Gamma^*, \sigma \in \Gamma$.

הגדרה 5: הכרעה של שפה

תהי $(Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג, ו- $L \subseteq \Sigma^*$ שפה.
נאמר כי M מכריעה את L אם לכל $w \in \Sigma^*$ מתקיים

- $M \Leftarrow w \in L$ מקבלת את w .
- $M \Leftarrow w \notin L$ דוחה את w .

הגדרה 6: קבלה של שפה

תהי $(Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$ מכונת טיורינג, ו- $L \subseteq \Sigma^*$ שפה.
נאמר כי M מקבלת את L אם לכל $w \in \Sigma^*$ מתקיים

- אם $w \in L$ מקבלת את w .
 - אם $w \notin L$ לא מקבלת את w .
- במקרה כזה נכתב ש- $L(M) = L$.

הגדרה 7: חישוב פונקציות

תהי $f : \Sigma_1^* \rightarrow \Sigma_2^*$ מכונת טיורינג ותהי $M = (Q, \Sigma, \Gamma, \delta, q_0, \text{acc}, \text{rej})$:
נאמר כי M מחשבת את f אם:

- $\Sigma = \Sigma_1, \Sigma_2 \subseteq \Gamma$
- לכל $w \in \Sigma_1^*$ מתקיים $q_0 w \vdash_M^* \text{acc } f(w)$

2 וריאציות של מכונות טיורינג

הגדרה 8: מודל חישוב

מודל חישובי = אוסף של מכונות שעבורם מוגדרים המושגים של הכרעה וקבלת שפות.

הגדרה 9: מודלים שקולים חישובית

יהיו A, B מודלים חישוביים. נאמר כי A ו- B שקולים אם לכל שפה L :

- קיימת מכונה במודל A שמכריעה את L אם ו רק קיימת מכונה כזו במודל B .
- קיימת מכונה במודל A שמקבלת את L אם ו רק קיימת מכונה כזו במודל B .

הגדרה 10: מכונות שקולות חישובית

שתי מכונות הן שקולות חישובית אם הן מקבלות ודווחות בדיקת אותן המילים.

משפט 1: מכונות טיורינג עם סרט ימינה בלבד

מודול מ"ט טם סרט אינסופי לכיוון אחד בלבד (מודול O) שקול למודל אינסופי בשני הכוונים (מודול T).
כלומר, לכל שפה L :

- יש מ"ט מודול O שמקבלת את L אם ויחי יש מ"ט במודול T שמקבלת את L .
- יש מ"ט מודול O שמכריעה את L אם ויחי יש מ"ט במודול T שמכריעה את L .

משפט 2: מכונות טיורינג מרובת סרטים

במכונת טיורינג מרובה סרטים:

- יתכנו מספר סטריטים.

מספר הסרטיטים סופי וקבוע מראש בזמן בניית המ"ט, ואיןו תלוי בקלט או במהלך החישוב.

- לכל סרט יש ראש נפרד.

הפעולות (תנועה וכתיבה) בכל סרט נעשית בנפרד.

בפרט, הראשים יכולים לאוז בכיוונים שונים בסרטיטים שונים.

- ישנו בקר מרכזי יחיד, שקובע את הפעולות בכל אחד מהסרטיטים, על סמך המידע שמתתקבל מכל הסרטיטים.

לכן, תוכן סרט אחד יכול להשפיע על הפעולות בשאר הסרטיטים.

- בתחלת החישוב, הקלט נמצא בסרט הראשון ושאר הסרטיטים ריקים.

משפט 3:

לכל k , המודול של מ"ט עם k סטריטים שקול חישובי למודול של מ"ט עם סרט אחד.

משפט 4:

קיבלה ודחיה של מחרוזות:

עבור מ"ט לא דטרמיניסטיבית N ומחרוזת w :

- N מקבלת את w אם קיים חישוב של N על w שmagiu במצב מקבל.
- N דוחה את w אם כל החישובים של N על w עצרים במצב דוחה.

הכרעה וקיבלה של שפות:

עבור מ"ט לא דטרמיניסטיבית N ושפה L :

- N מכריעה את L אם N מקבלת אצ' כל המילים ב- L ודוחה את כל המילים שאין ב- L .
- N מקבלת את L אם N מקבלת אצ' כל המילים ב- L ולא מקבלת את כל המילים שאין ב- L .

משפט 5:

לכל מ"ט לא דטרמיניסטיבית קיימת מ"ט דטרמיניסטיבית שcola.

3 התזה של צ'רצ'-טיורינג

שמות נרדפים לשפות כריעות ושפות קבילות

Acceptable languages	שפות קבילות	Decideable languages	שפות כריעות
recognizable languages	שפות ניתנות לייחוי	Recursive languages	שפות רקורסיביות
Semi-decidable languages	שפות כריעות למחצה		
Partially-decidable languages			
Recursively enumerable languages	שפות הניתנות למנייה רקורסיביות		

משפט 7: סגירות שפות קבילות

- איחוד
- חיתוך
- שרשור
- סגור קלין

משפט 6: סגירות שפות כריעות

השפות הכריעות סגורות תחת:

- איחוד
- חיתוך
- משלימים
- שרשור
- סגור קלין

משפט 8: היחס בין הכרעה לקבלה

אם שפה הינה כרעה אז היא קבילה.
 אם שפה ומשלימים שלה קבילות אז היא כרעה.

**הגדרה 11: שפת סימפל
משתנים**

- טבאים: ..., j, i, k.
 - מקבלים כערך מספר טבעי.
 - מערכים: ..., C[], B[], A[] בכל תא ערך מתוך א"ב Γ אין סופיים.
 - אתחול: הקלט נמצא בתאים הראשונים של [A].
- כל שאר המשתנים מאותחלים ל-0.

פעולות

- השמה בקבוע:

$$i=3, B[i]="#"$$

- השמה בין משתנים:

$$i=k, A[k]=B[i]$$

- פעולות חשבונ:

$$x = y + z, x = y - z, x = y \cdot z$$

תנאים

$B[i] == A[j]$ •

(מערכיים).

$x >= y$ •

(משתנים טבאיים).

כל משתנה מופיע רק פעם אחת בכל פעולה או תנאי.

זרימה

• סדרה פקודות ממספרות.

• `goto`: מותנה ולא מותנה.

• עצירה 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)
```

הגדרה 12: קבלת ודוחיה של מחוזות בשפה SIMPLE

עבור קלט

w ותוכנית

P בשפת SIMPLE. נאמר 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 13: הכרעה וקבלת של שפה

עבור שפה L ותוכנית P בשפת SIMPLE. נאמר כי

- P מכירעה את L אם היא מקבלת את המילים שב- L ודוחה את אלה שלא ב- L.
- P מקבלת את L אם היא מקבלת את כל ורקי המילים ב- L.

משפט 9:

המודלים של מכונת טיורינג ותוכניות SIMPLE שקולים.

משפט 10: מ"ט ותוכניות מחשב

מ"ט חזקה לפחות כמו תוכנית משחוב.

כל תוכנית מחשב ניתנת למימוש במ"ט.

לכן, כל שפה שהינה כריעה ע"י מחשב היא גם כריעה ע"י מ"ט.

וכמו כן, שפה שהינה קבילה ע"י מחשב היא גם קבילה ע"י מ"ט.

הגדרה 14: דקדוקים כלליים

בדקdock כללי, מצד שמאל של כלל יצירה יכולה להופיע מחרוזת (לא ריקה) כלשהי.
פורמלית, כלל יצירה בדקדוק כללי הוא מהצורה

$$\gamma \rightarrow u$$

כאשר $(V \cup \Sigma)^*$, $\gamma \in (V \cup \Sigma)^+$.

משפט 11:

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

מודל חישובי	דקdock	משפחת שפות
מכונת טיורינג	כללי	קבילות
אוטומט מחסנית	חסר הקשר	חסירות הקשר
אוטומט סופי	רגולרי	רגולריות

משפט 12:

כל שפה חסרת הקשר אינה כריעה.

משפט 13: התזה של צ'רצ' טיורינג

התזה של צ'רצ' טיורינג מודל מ"ט מגלה את המושג האבסטראקטי של "אלגוריתם".

כלומר, כל אלגוריתם ניתן לתיאור כתהיליך מכניים שבו:

- התהיליך מתבצע סדרה של צעדים.
- כל צעד מצריך כמהות סופית של "עובדת".

ניתן גם לתיאור כמ"ט.

בפרט, אין מודל מכנייסטי / אוטומטי יותר ממ"ט.

4 אי-כריעות

הגדרה 15: השפה ATM

$$ATM = \{(P, w) \mid P(w) = 1\} .$$

השפה ATM כוללת את כל הזוגות של מחרוזות w , P כך ש:

- P היא קוד (תקין) של תוכנית.

• w מחרוזת.

- מתקיים שם מרכיבים את התוכנית P על הקלט w אז התוכנית עוצרת עם ערך חזרה 1.

הגדרה חלופית:

$$ATM = \{\langle M, w \rangle \mid w \text{ מכונת טיורינג שמקבלת את } M\}$$

השפה A_{TM} כוללת את כל הזוגות של מחרוזות $\langle M, w \rangle$ של כל מכונת טיורינג M וכל קלט w כך ש- M מקבלת את w .

סיכום 1: התוכנה U

התוכנה U היא תוכנה שמקבלת כקלט זוג מחרוזות w , P ופועלת כך:

- U מחזירה את ערך החזרה שהתקבל מהריצה של P על w .

- מರיצה את התוכנה P על קלט w (במקרה שבו P אינה תוכנית מחשב תקינה אז U מחזירה ערך 0).

נשים לב שגם P לא עוצרת על w אז גם U לא עוצרת על הזוג w, P .

התוכנה U פועלת באופן דומה לאופן שבה מערכת הפעלה מפעילה תוכנות אחרות.

התוכנה U נקראת גם "תוכנה אוניברסלית" (או, בעולם מכונות טיורינג: "מכונת טיורינג אוניברסלית") כיון שהיא תוכנה אחת שمدמה כל תוכנה אחרת.

U היא תוכנית שמקבלת את ATM . כמובן:

$$L(U) = ATM .$$

מסקנה: ATM קבילה.

שיטת ההוכחה:

אם שפה הינה קבילה אבל לא כריעה, אז המשלימים שלה בהכרח אינה קבילה. לכן, בשביל להוכיח שפה אינה קבילה, די להוכיח שהשפה לא כריעה, והמשלימים שלה כן קבילה.

הגדרה 16: השפה HALT

$$HALT = \{(P, w) \mid P(w) \downarrow\} .$$

השפה HALT כוללת את כל הזוגות של מחרוזות w , P כך ש:

- P היא קוד (תקין) של תוכנית.

• w מחרוזת.

- מתקיים שם מרכיבים את התוכנית P על הקלט w אז התוכנית עוצרת (הסימן \downarrow מסמן עצירה).)

בUPIIT העצירה קבילה אבל לא כריעה.

כיוון שכך, המשלים שלו לא כריעה ולא קבילה.

הגדרה חלופית:

$$HALT_{TM} = \{\langle M, w \rangle \mid w \text{ מוגדרת על } M\}$$

השפה $HALT_{TM}$ כוללת את כל הזוגות של מחרוזות $\langle M, w \rangle$ של כל מכונת טיורינג M וכל קלט w כך ש- M עוצרת על w .

הגדרה 17: השפה E

$$E = \{P \mid L(P) = \emptyset\}$$

השפה E כוללת את כל המחרוזות P כך ש-

- P היא קוד (תקין) של תוכנית.
- השפה של P ריקה.

כלומר, לכל קלט w , הריצה של P על w לא מחזירה 1.

הגדרה חלופית:

$$E_{TM} = \{\langle M \rangle \mid L(M) = \emptyset\}$$

השפה E_{TM} כוללת את כל מחרוזות $\langle M \rangle$ של כל מכונת טיורינג M כך ש- M לא מקבלת אף מילה. במילים אחרות, השפה של M ריקה: $L(M) = \emptyset$.

הגדרה 18: השפה EQ

$$EQ = \{(P_1, P_2) \mid L(P_1) = L(P_2)\}.$$

השפה EQ כוללת את כל זוגות המחרוזות P_1, P_2 כך ש:

- P_1, P_2 הינם קודים (תריניים) של תוכניות.
- השפות של P_1, P_2 זהות.

כלומר, P_1, P_2 מקבלות בדיקת אוטן המילים.

הגדרה חלופית:

$$EQ_{TM} = \{\langle M_1, M_2 \rangle \mid L(M_1) = L(M_2)\}$$

השפה EQ_{TM} כוללת את כל זוגות של מכונות טיורינג $\langle M_1, M_2 \rangle$ שמקבלות בדיקת אוטן המילים. במילים אחרות, השפות של M_1 ו- M_2 זהות: $L(M_1) = L(M_2)$.

כריעה	קבילה	
✓	✗	ATM
✗	✗	\overline{ATM}
✓	✗	$HALT$
✗	✗	\overline{HALT}
✗	✗	E
✓	✗	\overline{E}
✗	✗	EQ
✗	✗	\overline{EQ}

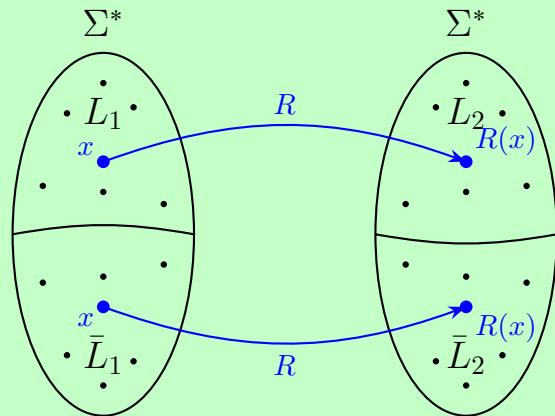
הגדרה 19: הרדוקציה

רדוקציית התאמה $L_2 \subseteq \Omega_2$ מקבוצה $L_1 \subseteq \Omega_1$ (many to one reduction) לקבוצה הינה **פונקציה**

$$R : \Omega_1 \rightarrow \Omega_2$$

כך שלכל $x \in \Omega_1$ מתקיים:

$$x \in L_1 \Leftrightarrow R(x) \in L_2 .$$



סיכום: רימית רדוקציה התאמה ניתנת לחישוב מ- $L_1 \leq_m L_2$.

משפט 14: משפט הרדווקציה

טענה:
אם:

- L_2 קבילה
- $L_1 \leq L_2$
- או L_1 קבילה.

- L_2 כריעה
- $L_1 \leq L_2$
- או L_1 כריעה.

מסקנה:
אם:

- L_1 לא קבילה
- $L_1 \leq L_2$
- או L_2 לא קבילה.

- L_1 לא כריעה
- $L_1 \leq L_2$
- או L_2 לא כריעה.

מתכוון להוכחה ששפה L_2 לא קבילה:

1. בחר שפה L_1 לא קבילה.
2. מצא רדווקציית התאמה ניתנת לחישוב מה- L_1 ל- L_2 .

מתכוון להוכחה ששפה L_2 לא כריעה:

1. בחר שפה L_1 לא כריעה.
2. מצא רדווקציית התאמה ניתנת לחישוב מה- L_1 ל- L_2 .

משפט 15: תכונות של רדווקציות

A	\leq_m	B
כריעת	\Leftarrow	כריעת
לא כריעת	\Rightarrow	לא כריעת

A	\leq_m	B
קבילת	\Leftarrow	קבילת
לא קבילה	\Rightarrow	לא קבילה

משפט 16: לכל שפה קיימת רדווקציה ל- A_{TM}
מכל שפה כריעת A קיימת רדווקציה חישובית ל- A_{TM} .

כלומר

$$A \leq_m A_{TM} .$$

משפט 17: רדוקציה משפות בריעות

מכל שפה כריעה קיימת רדוקציה חסיבה לכל שפה אחרת שאינה \emptyset או Σ^* .

הגדרה 20:

$$NOTREG = \{P \mid L(P) \text{ לא רגולרית}\}.$$

השפה NOT-REG כוללת את כל המחרוזות P כך ש:

- P אינה קוד (תקין) של תוכנית.
- השפה של P לא רגולרית.

הגדרה חלופית:

$$NOTREG_{TM} = \{\langle M \rangle \mid L(M) \text{ לא רגולרית}\}.$$

השפה $NOTREG_{TM}$ כוללת את כל המחרוזות $\langle M \rangle$ של מ"ט M כך שהשפה של M לא רגולרית.

משפט 18: השפה $NOT - REG$ אינה קבילה.

השפה $NOT - REG$ אינה קבילה.

5 סיבוכיות זמן

הגדרה 21: זמן הריצה

זמן הריצה של מכונת טיורינג M על קלט w הוא מספר צעדי החישוב ש- M מבצעת על w .

הגדרה 22: סיבוכיות זמן ריצה

תהי M מ"ט דטרמיניסטי אשר עוצרת על כל קלט. **הזמן הריצה או הסיבוכיות זמן של M** היא פונקציה $\mathbb{N} \rightarrow \mathbb{N}$: f , כאשר $f(n)$ המספר צעדי חישוב המקסימלי ש- M מבצעת על קלט w של אורך n .

אם $f(n)$ זמן הריצה של M , אומרים כי M רץ בזמן $f(n)$ וש- M היא $(f(n))$ זמן מכונת טיורינג

הגדרה 23: מחלוקת של סיבוכיות זמן

המחלקת הסיבוכיות זמן מסומנת $((t(n)) \text{ TIME})$ ומוגדרת להיות אוסף של כל השפות אשר ניתן להכרעה על ידי מכונת טיורינג בזמן $((t(n))O)$.

משפט 19:

הגדרת זמן הריצה שנתנו היא תליה במודל של מכונת הטיורינג שאיתו אנחנו עוסדים.

משפט 20:

תהי $t : \mathbb{N} \rightarrow \mathbb{R}^+$ פונקציה (asm 타입) מ"ט.

$$t(n) \geq n$$

אז לכל מכונת טיורינג $O(t(n))$ רב-סרטי קיימת מ"ט $O(t^2(n))$ עם סרט אחד.

הגדרה 24: זמן הריצה של מ"ט לא דטרמיניסטיבית

תהי N מכונת טיורינג לא דטרמיניסטיבית.

זמן הריצה של N מוגדרת להיות הפונקציה $\mathbb{N} \rightarrow \mathbb{N}$: $f(n)$ הוא המספר הצעדים המקסימלי אשר N מתוק כל הענפים של החישוב שלו על קלט של אורך n .

משפט 21:

תהי $t(n)$ פונקציה המקיים $n \geq t(n)$. כל מ"ט $O(t(n))$ לא דטרמיניסטיבית N סרט אחד, שකולה למוכנת טיורינג $2^{O(t(n))}$ דטרמיניסטיבית סרט אחד.

הגדרה 25: מכונת טיורינג פולינומית

מכונת טיורינג M תיקרא **פולינומית או עיליה** אם קיים $\mathbb{N} \in c$ כך ש- M פועלת בסיבוכיות זמן ריצה $O(n^c)$.

הגדרה 26: המחלקה P

המחלקה P היא אוסף השפות שקיימות מכונת טיורינג פולינומיאלית M המכדרעה אותן. ככלומר:

$$P = \bigcup_k \text{TIME}(n^k).$$

הגדרה 27: אלגוריתם אimotoות

אלגוריתם אimotoות של שפה A הוא אלגוריתם V כך ש-:

$$A = \{w \mid V \text{ מקבל } \langle w, c \rangle \text{ על פי}\}$$

במילים, **אלגוריתם אimotoות** הוא אלגוריתם V אשר מאמת כי הקלט w שייך לשפה A על פי התנאי c , שנקרא **אישור** (certification).

אנחנו מגדירים את זמן הריצה של V על פי האורך של w . לכן **אלגוריתם אimotoות זמן-פולינומיאלי** רץ בזמן פולינומיאלי $O(n^k)$ כאשר n האורך של w .

הגדרה 28: מחלקה הסיבוכיות NP

- המחלקה NP היא מחלקה השפות שניתן לאמתן באמצעות אלגוריתם זמן-פולינומיאלי.

הגדרה חלופית למחלקה NP הינה:

- המחלקה NP היא מחלקה השפות שניתן להכרעה באמצעות מ"ט אי-דטרמיניסטיבית זמן-פולינומיאלית.

למטה במשפט 22.

משפט 22: אם $A \in NP$ ניתן לאimotoות ע"י N_{TM} שפה A כלשהי שיכת למחלקה NP אם ורק אם A ניתנת להכרעה על ידי מכונות טיורינג אי-דטרמיניסטיבית זמן-פולינומיאלית.

הגדרה 29: פונקציה הניתנת לחישוב זמן-פולינומיאלי פונקציה $\Sigma^* \rightarrow \Sigma^*$: ניתנת לחישוב זמן-פולינומיאלי אם קיימת מ"ט זמן-פולינומיאלית M , עבורה על הקלט w , M עוצרת עם $f(w)$ על הסרט שלה.

הגדרה 30: פונקציה שנייה לרזוקציה זמן-פולינומיאלית השפה A ניתנת לרזוקציה זמן-פולינומיאלית לשפה B , שנסמן $B \leqslant_P A$, אם קיימת פונקציה שתניתת לחישוב זמן-פולינומיאלית $\Sigma^* \rightarrow \Sigma^*$ כך שלכל

$$w \in A \Leftrightarrow f(w) \in B .$$

הfonקציה f נקראת **רזוקציה זמן-פולינומיאלית** של A ל- B .

משפט 23: אם $A \in P$ אז $B \in P$ ו- $A \leqslant_P B$.
אם $B \in P$ אז $A \leqslant_P B$.

משפט 24: SAT-3 ניתנת לרזוקציה זמן-פולינומיאלית ל- CLIQUE בהביית SAT-3 ניתנת לרזוקציה זמן-פולינומיאלית לביעית CLIQUE:
 $3SAT \leqslant_p CLIQUE$.

מסקנה 1: $CLIQUE \in P \Rightarrow 3SAT \in P$
לפי המשפט 23 וממשפט 24:

אם $3SAT \in P$ אז $CLIQUE \in P$

הגדרה 31: NP-שלומות שפה B היא NP-שלמה או שלמה ב- NP (NP-complete) אם היא מקיימת את שני התנאים הבאים:
(1) $B \in NP$ וגם
(2) $A \in NP$ $A \leqslant_p B$ עבור כל $A \leqslant_p B$.
במילים פשוטות: כל A ב- NP ניתנת לרזוקציה זמן-פולינומיאלית ל- B .

הגדרה 32: NP קשה אם שפה B מקיימת את תכונה (2) אולם לא בהכרח את תכונה (1) בהגדרה 31 אז אומרים כי B -קשה או קשה ב- NP (NP-hard).

משפט 25:אם $B \in P$ אז $P = NP$ - שלמה ו-**משפט 26: אסוציאטיביות של NP שלמות**
אם השני תנאים הבאים מתקיימים:

- (1) B היא שפה NP - שלמה.
- (2) קיימת $C \in NP$ עבורה $B \leq_p C$ אז C שפה NP - שלמה.

משפט 27: משפט קוק לויןהבעית SAT היא NP - שלמה.**משפט 28: 3-SAT הינו NP שלמה.**
אם $3-SAT$ הינו NP שלמה.

6 נושאות נוספות

הגדרה 33: הבעית הספיקות SAT

$$SAT = \{ \langle \phi \rangle \mid \phi \text{ היא נוסחה בוליאנית ספיקה} \}$$

במילים, בעית SAT שואלת את השאלה: בהינתן נוסחה שמכילה רק קשרים \wedge , \vee ו- \neg ("גמ", "או" ו"לא"), האם קיימות השמה של ערכי אמת למשתנים כך שהנוסחה ϕ תקבל ערך אמת? אם קיימת השמה כזו אז נאמר כי הנוסחה ϕ ספיקה.

הגדרה 34: הבעית 3-SAT

$$3SAT = \{ \langle \phi \rangle \mid \phi \text{ היא נוסחת בוליאנית בצורה 3CNF ספיקה.} \}$$

במילים, $3SAT$ היא הבעית SAT שמוגדר בהגדירה 33 במקרה הנוסחה שהיא בצורה 3CNF. דוגמה של נוסחת בצורה 3CNF היא:

$$(x_1 \vee \bar{x}_1 \vee \bar{x}_2) \wedge (x_3 \vee x_2 \vee x_4) \wedge (\bar{x}_1 \vee \bar{x}_3 \vee \bar{x}_4)$$

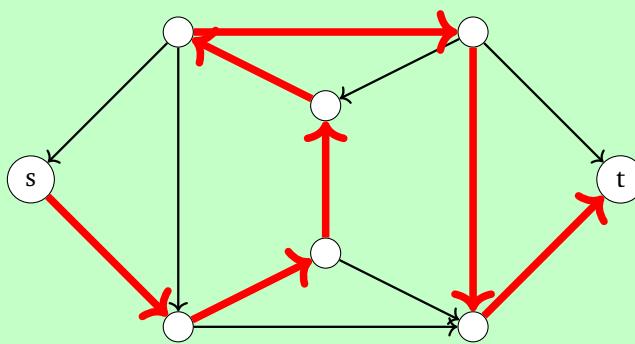
הגדרה 35: הבית PATHבහינתן גרף מכוון $G = (V, E)$.הבעית PATH שואלת את השאלה הבאה: בהינתן גרף מכוון $G = (V, E)$, וקודקודים s ו- t . האם הגרף G מסלול בין קודקוד s לבין קודקוד t .

$$PATH = \{ \langle G, s, t \rangle \mid t \text{ נמצא במסלול שמכיל מסלול מכוון מ- } s \text{ ל- } t \} .$$

הגדרה 36: מסלול המילטונינתון גרף מכובן $G = (V, E)$.מסלול המילטוני מקדקוד s לקדקוד t הוא מסלול אשר עובר דרך כל קדקוד בדיקות אחת.**הגדרה 37: הביעית מסלול המילטוני HAMPATH**בהינתן גרף מכובן $G = (V, E)$ וקדקודים s ו- t .הבעיית המסלול המילטוני שואלת את השאלה: האם קיים מסלול המילטוני מקדקוד s לקדקוד t ?

$$HAMPATH = \{\langle G, s, t \rangle \mid \text{גרף } G \text{ מכיל מסלול המילטוני מ- } s \text{ ל- } t\}$$

התרשימים למטה מראją דוגמה של מסלול המילטוני בגרף מכובן.

**הגדרה 38:**בහינתן שלמים x , y .הבעיה RELPRIME שואלת את השאלה: האם x , y זרים.

$$RELPRIME = \{\{x, y\} \in \mathbb{N} \mid \gcd(x, y) = 1\}$$

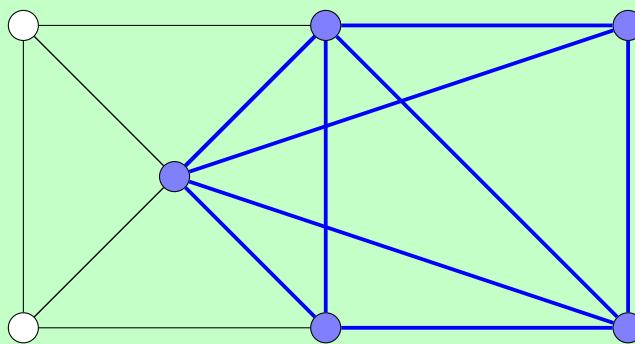
הגדרה 39: קליקה

נתון גרף לא מכובן.

- קליקה בgraf לא מכובן הוא תת-graf שבו כל זוג קדקודים קשורים על ידי קשת.

- k - קליקה היא קליקה שבו יש בדיקוק k קדקודים.

התרשימים למטה מראją דוגמה של 5- קליקה.



הגדרה 40: בעיית הקליקה

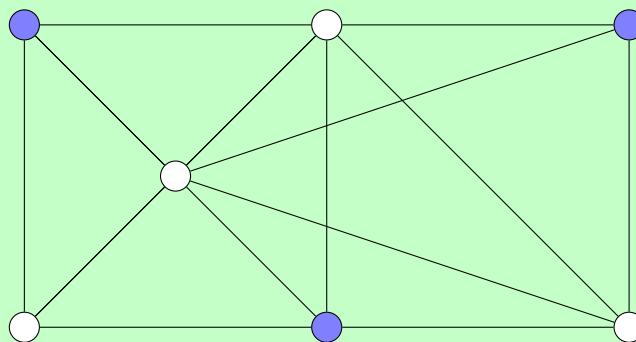
נתון גרף לא מכוון $G = (V, E)$. בעיית הליקה שואלת את השאלה:
אם הגרף G מכיל קליקה בגודל k .
בשפה פורמלית:

$$CLQ = \{\langle G, k \rangle \mid G \text{ גרף לא מכוון שמכיל קליקה בגודל } k \text{ לפחות.}\}$$

הגדרה 41: קבוצה בלתי תלויות

נתון גרף לא מכוון $G = (V, E)$. קבוצה בלתי תלויות ב- G היא תת-קבוצה של קדוקודים $S \subseteq V$ כך שלכל שני קדוקודים $S \in S$, u_1, u_2 מתקאים $(u_1, u_2) \notin E$.

התרשימים למטה מראה דוגמה של גרף לא מכוון G שמכיל קבוצה בלתי תלויות בגודל 3.

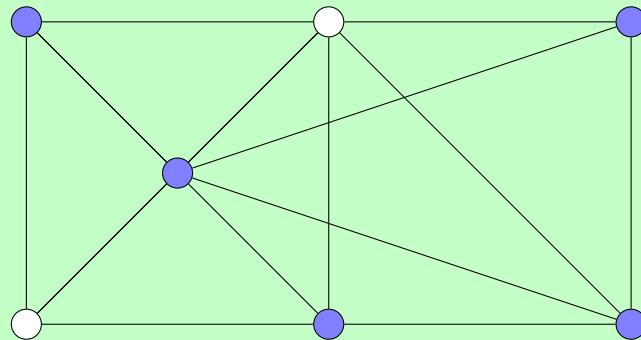
**הגדרה 42: בעיית קבוצה הבלתי תלויות (IS)**

בהינתן גרף לא מכוון $G = (V, E)$ ומספר טבעי k .
הבעיה IS שואלת את השאלה: האם קיימת קבוצה בלתי תלויות ב- G בגודל k לפחות.
בשפה פורמלית:

$$IS = \{\langle G, k \rangle \mid G \text{ מכיל קבוצה בלתי תלויות בגודל } k \text{ לפחות.}\}$$

הגדרה 43: כיסוי קדוקודים

בהינתן גרף לא מכוון $G = (V, E)$:
כיסוי קדוקודים ב- G הוא תת-קבוצה של קדוקודים $V \subseteq C$ כך שלכל צלע $(u_1, u_2) \in E$ מתקיים:
 $u_1 \in C$ או $u_2 \in C$.
הgraf למטה מכיל כיסוי קדוקודים בגודל 5.

**הגדרה 44: הבעית כיסוי קדקודים (VC)**

בහינתן גרף לא מקוון $G = (V, E)$ ומספר טבעי k .

הבעית כיסוי קדקודים שואלת את השאלה הבאה:

האם קיימים כיסוי בקדקודים ב- G בגודל k ?

בשפה פורמלית:

$$VC = \{\langle G, k \rangle \mid G \text{ מכיל כיסוי בקדקודים בגודל } k\}.$$

משפט 29: שפות NP-שלמות

(משפט קוק לוין) SAT -NP- שלמה.

3SAT -NP- שלמה.

HAMPATH -NP- שלמה.

CLIQUE -NP- שלמה.

INDEPENDENT-SET -NP- שלמה.

VERTEX-COVER -NP- שלמה.