בס"ד

אופרה אהרון'

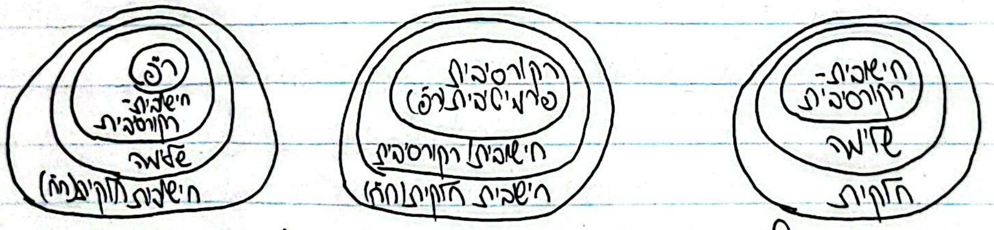
חישוביות - סיכום - חשוב לזכור!

* תוכנית בשפת S לעולם מחשבת פונקציה! לא כל פונק' ניתנת לחישוב ע"י תוכנית בשפת S.
* פלט של תוכנית - הערך של משתנה Y כאשר התהליך מסתיים. לא תמיד התהליך מסתיים.
* תוכנית שלא עוצרת לא תמיד מחזירה ערך, אבל כן מחשבת פונקציה.
* פונקציה - נוסחה התואמת מתמטית המתאימה לווקטור של כניסות פלט. ישות מתמטית.
* תוכנית - "שם מעשי" לנוסחה התואמת לחישוב. ישות תחבירית.
- פונקציה זו לא תוכנית. לכל פונק' יש אינסוף תוכניות המחשבות אותה.
- כל תוכנית מחשבת פונק' אחת בלבד.

• פונקציה מתאימה ערך y לכל ערך x שעליו היא מוגדרת
• תוכנית מחזירה ערך y לכל ערך x שעליו היא עוצרת

- קלט שעליו התוכנית לא עוצרת, הוא קלט שעליו הפונק' המתאימה לא מוגדרת.
- תוכנית שעוצרת תמיד מחזירה פלט (הפלט הוא מה שנמצא ב y).
- ↑ מסמל אי הגדרה בפונקציה, בתוכנית הכוונה לאי עצירה.

* פונקציה חלקית - כל פונקציה, כיוון שלכל נוסחה התואמת (פונקציה) בעולם.
* פונקציה חלקית ממש - פונקציה שקיים קלט אחד לפחות שעליו היא לא מוגדרת.
* פונקציה שלמה - פונק' שתחום הגדרתה הוא N. לכל ערך בקלט מתאים ערך בפלט. לא קיים ערך שעבורו הפונקציה לא מוגדרת (מוגדרת לכל x).
* פונקציה חישובית חלקית (ח"ח) - פונק' שקיימת תוכנית בשפת S המחשבת אותה, לא בהכרח עוצרת.
* פונקציה חישובית (ממש)/רקורסיבית - פונק' שקיימת תוכנית בשפת S המחשבת אותה, הפונק' תמיד עוצרת ולכל קלט מחזירה פלט.
- פונקציה חישובית היא בהכרח שלמה! אבל פונק' שלמה לא בהכרח חישובית.
* פונקציה שאינה חישובית בכלל (אפילו לא חישובית חלקית (ח"ח)) - פונק' שלא קיימת תוכנית בשפת S שיכולה לחשב אותה.



* הפונקציה $f(x)=1$, הפרדיקט האופייני של הקבוצה N (פונק' חלקית, שלמה, ח"ח, חישובית, פרדיקט אופייני חשוב!).
* הפונקציה $f(x)=0$, הפרדיקט האופייני של הקבוצה הריקה {}
* הפונקציה המתבדרת (הריקה) $h(x)=\uparrow$, פונק' שאינה מוגדרת לשום קלט, ח"ח.

בס"ד

חישוביות - סיכום - חשוב לזכור!

* **פרדיקט** - פונקציה שלמה שהטווח שלה הוא מהקבוצה {0,1}
- לכל פרדיקט מתאימה תת קבוצה - אוסף הקלטים שערך הפרדיקט עבורם הוא 1
ולהפך, לכל תת קבוצה מתאים פרדיקט - ערכו 1 עבור איברי תת הקבוצה ו0 אחרת.

* **פרדיקט אופייני** - כינוי לכל פרדיקט בעולם. מצד הגדרתו הוא מאפיין קבוצה שבה יכללו אך ורק הערכים שהפרדיקט מחזיר עבורם 1.

* **פרדיקט אופייני חישובי** - פרדיקט שקיימת תוכנית בשפת S המחשבת אותו ממש.
תחום ההגדרה שלו הוא כל N, טווח הקבוצה {1,0}, בתוכנית לחישוב הפרדיקט תהיה התוכנית תעצור לכל x. אם x בקבוצה התוכנית תחזיר 1 אחרת 0.
- בפרדיקט רגיל לא בהכרח קיימת תוכנית המחשבת אותו.
- תוכנית בשפת S אחרי IF חייב להופיע פרדיקט **חישובי**!

◦ **התוכנית הריקה** - אין בה שורות קוד { }, מבצעת 0 צעדים. **עוצרת** מיד עם תחילת החישוב, מחזירה ערך 0! נחשבת כעוצרת לכל קלט.

* **תחום** (Dom) - עבור אילו ערכים הפונק' מוגדרת.
כלומר, אלו ערכי הx (בשמאל) שעבורם מותאם ערך מספרי (מימין).

* **טווח** (Range) - עבור אלו ערכים הפונק' מתאימה. קבוצת כל ערכי הע' האפשריים שהפונק' מתאימה לעבור x כלשהו, כלומר כל האיברים שנמצאים בחלק הימני של הצמד.

* **פונקציות רקורסיביות פרימיטיביות** - כדי להראות שפונק' היא ר"פ: נראה שניתן לבנות אותה תוך שימוש בהרכבה/רקורסיה פרימיטיבית של פונק' ר"פ בלבד.

הפונקציות הבסיסיות (ר"פ):

◦ **פונקציית האפס**: $Z(\,)=0$ $\quad Z:N^0\to N$

◦ **פונקציית העוקב**: $S(x)=x+1$ $\quad S:N\to N$

◦ **פונקציות הטלה**: $\pi_i^n(x_1,\dots,x_n)=x_i \quad n\geq 1,\ \forall 1\leq i\leq n$ $\quad \pi_i^n:N^n\to N$

- **פונקציה שימושית להחזרת קבוע**: $C_i^n(x_1,\dots,x_n)=i$

* **פונקציה רקורסיבית פרימיטיבית** (ר"פ) - **הגדרה**: פונקציה המתקבלת מהפונקציות הר"פ הבסיסיות, ע"י **מס' סופי של הרכבות ורקורסיות**.
- כל פונקציה רקורסיבית פרימיטיבית היא חישובית/רקורסיבית.

* **הגדרה רקורסיבית מספקת 2 דברים**:
1. מס' ערכים בסיסיים מפורשים של הפונק'.
2. כלל המרה איך לקבל את הערך הבא באמצעות הערכים שכבר הוגדרו.

בס"ד

חישוביות - חשוב לזכור!

* פונק' שלמה איננה בהכרח חישובית.
הסבר: פונק' שלמה - קשורה לתחום - מתאימה לכל קלט פלט. חישובית - קשור לתוכנית שמחשבת אותה.
* פונק' חישובית היא בהכרח שלמה (שלמה חישובית)
* כל פונק' ר"פ היא חישובית (חישובית ר"פ)
* כל פונק' חישובית היא פונק' רקורסיבית (חישובית - מילה נרדפת לרקורסיבית).
* הפונק' $h(x)=\uparrow$ (המתבדרת) היא חישובית חלקית (קיימת תוכנית בשפת S שמחשבת אותה).
* הפונק' $g(x)=\frac{1}{x}$ איננה פונק' חישובית חלקית (אין תוכנית שמחשבת אותה).
* הפונק' $f(x)=\begin{cases}1 & x=3\\ \uparrow & else\end{cases}$ היא לא חישובית כלל - לא נכון! היא חישובית חלקית.
* הפרדיקט האופייני של הקבוצה הריקה הוא $f(x)=1$ - לא נכון! הוא $f(x)=0$.
* הפרדיקט האופייני של הקבוצה N הוא $f(x)=1$.
* אם $(x|2)$ "מחלק" אז x הוא מספר זוגי - לא נכון! אם $(2|x)$ "2 מחלק את x" אז x הוא מספר זוגי.
* התוכנית הריקה מחשבת כעוצרת לכל קלט, מחזירה תמיד את הערך 0 שהוא הפרדיקט האופייני של הקבוצה הריקה.
* כל תוכנית בשפת S מחשבת פונק', אבל רק תוכניות שעוצרות מחזירות ערך.
* התוכנית [A] goto A איננה מחזירה ערך, אבל כן מחשבת פונק' ח"ח (מחשבת את הפונק' המתבדרת).
* מס' הפרמטרים שמקבלת פונק' מורכבת נקבע לפי סכום מס' הפרמטרים שמקבלות הפונק' החיצוניות ביותר בהרכבה - לא נכון!! הפונק' הפנימיות ולא החיצוניות.
* פונק' החיסור בערך מוחלט ניתנת לייצוג תוך שימוש בהרכבה של פונק' ר"פ בלבד.
* בהינתן הפונק' הקבועה $f(x)=3$ הנוסחה הרקורסיבית המתאימה בצורה:
$$\begin{cases}f(0)=3\\ f(t+1)=\pi^2_2(t,f(t))=f(t)\end{cases}$$
(כמובן שניתן לייצג את כל הפונק' הקבועות - בשינוי ערכי העצירה).
* בהינתן הפונק' $f(x,y)=\min_{z\le x\cdot y}((x|z)\wedge(y|z))$ (הכ"מ: הראשון ש-x מחלק אותו וגם y מחלק אותו. המחלק בצד שמאל.)
הפונק' מחזירה את הכפולה המשותפת המינימלית של x ו-y
הערה: יש לשים לב לסדר בפונק' הזאת: $((z|x)\wedge(z|y))$ אז הפונק' הייתה מחזירה את המחלק המינימלי. (בקלט המקרים <0,<0,2>> ולא <0,<0,0>>)
* תוכנית שקידודה זוגי פותחת בפקודה $Z\leftarrow Z$ - לא נכון, פותחת בפקודה $Y\leftarrow Y$.
* קיימת תוכנית שקידודה זוגי ויש בה הוראת מעבר $V\leftarrow V$ - לא נכון.
* תוכנית שקידודה אי-זוגי פותחת בפקודה המתויגת N $Y\leftarrow Y$
* אני זוכרת שפונק' הפיבונאצ'י ר"פ, מכיוון שיש לה יותר ממשתנה עזר אחד, והיא נשענת על מס' ערכים קודמים. לא נכון! אומנם מכיוון שלפיבונאצ'י יש יותר ממשתנה עזר אחד וגם היא נשענת על יותר משני ערכים קודמים, לכן לא יכולנו להתאים אותה לתבנית של הרקורסיה הפרימיטיבית, אבל השתמשנו בדרך אחרת כדי להוכיח זאת.

חישוביות - חשוב לזכור!

* מס׳ פקודות של תוכנית תמיד מס׳ סופי. מס׳ הצעדים של תוכנית יכול להיות מס׳ אינסופי.
לבדוק אם הוראה מופיעה בתוכנית - בר ביצוע. לבדוק אם התוכנית מבצעת את ההוראה
זה כבר סיפור אחר - לא בהכרח בר בדיקה.

* השלשה $\langle 0,\langle 3,1\rangle\rangle$ מצפינה את ההוראה $\overset{\langle 1,\langle 3,0\rangle\rangle}{[A]\ \text{if } Y\neq 0 \text{ GOTO } A}$ - לא נכון (כי $a=0$ ויש לנו תוית).

* המס׳ 20 מצפין את הזוג $\langle 0,10\rangle$, וגם מספר גדל של הסדרה $[2,1]$ - לא נכון
אומנם $\langle 0,10\rangle = 2^0(2\cdot 10+1)-1 = 20$ אבל $2^2\cdot 3^1=12$ ולא 20.

* בהינתן x המהווה קידוד של תוכנית - האם הפונק׳ f שלהלן היא הפרדיקט האופייני
המתאים לקבוצת התוכניות שאורכן גדול מ-17.

$$\overset{\text{קידוד של תכנית}}{f(x)} = \begin{cases} 1 & lt(x-1) > 17 \\ 0 & \text{else} \end{cases}$$

- לא נכון. כך נכון f: $lt(x+1)$.

* פרדיקט האופייני לקבוצת התוכניות שההוראה השנייה בהן קידודה הוא 20:

$$f(x) \begin{cases} 1 & (x+1)_2 = 20 \\ 0 & \text{else} \end{cases}$$

* בהינתן x המהווה גדל של סדרת מספרים, כלומר $x = [a_1, a_2, \dots, a_n]$
נדרש לכתוב פונק׳ ר"פ המחזירה את האיבר המקסימלי מבין איברי הסדרה המוצפנת
ע"י x תוך שימוש במינימיזציה חסומה. $f(x) = \min_{z\le x}((x)_1 \le z \wedge (x)_2 \le z \wedge \dots \wedge (x)_n \le z)$

* נתון הפרדיקט הר"פ שלהלן: $\overset{\text{קידוד תוכניות}}{P(x)} = \sim\overset{x\text{ אינו זוגי}}{(2|x)} \wedge \underbrace{\overset{\text{גדל}}{(x+1)_2} = 0}_{\text{ההוראה השנייה בתוכנית היא }Y\leftarrow Y}$

בהינתן x המהווה קידוד של תוכנית, בעבור אילו תוכניות יתקיים $p(x)=1$:
תוכניות שקידודן אי זוגי, וההוראה השנייה בהן היא $Y\leftarrow Y$.

חישוביות - חשוב לזכור!

* תוכנית שעוצרת לכל קלט גוררת מס' אינסופי של קלטים - שקר נכון!
* תוכנית שעוצרת על מס' אינסופי של קלטים עוצרת לכל קלט - לא נכון!
* תוכנית שעוצרת לכל קלט גוררת על קלט הריק לעצור שלה - נכון!!
* תוכנית השייכת ל-K (עוצרת לקלט המתאר לעצמה) שייכת ל-TOT (עוצרת לכל קלט) - לא נכון!
* קבוצת התוכניות שעוצרות על קלט אחד בלבד היא קבוצה סופית - לא נכון! הסבר: כאשר מדובר בתוכניות תמיד אפשר להוסיף שורות שלא ידבר התוכניות למספר שעושות את אותו הדבר.
* פונק' חישובית - פונק' שקיימת תוכנית בשפת שיכולה לחשב אותה - לעצור ולהחזיר 1/0.
* קבוצה חישובית כריעה - כל קבוצה שאפשר לבדוק על כל אחד מהאיברים האם הוא שייך לקבוצה או לא שייך לקבוצה, קיים אלגוריתם שעושה את זה - עוצר ומחזיר תשובה.
  שואלים: האם קיים אלגוריתם שמסוגל לבדוק שייכות לקבוצה הזו?
* - תחום הגדרה: אלו כל הx-ים בשאלה שאותם להם ערך מספרי מימין
  - טווח: אלו כל הערכים בימין

		שם הפונק'	תחום A	טווח B	
$h(x) = \uparrow$	ח"ח	h	ϕ	ϕ	לה הטווח והתחום ריק
$halt(x,x)$	לא חישובית גם, ואפילו לא ח"ח	פונק' שלמה $halt$	N	$\{0,1\}$	שאין חח קבוצת ת"ה
$f_1(x) = 3$	חישובית רקורסיבית, ואפילו ת"ה	f_1	N	$\{3\}$	ובגלל ההגדרה תה
$f_2(x)=\begin{cases}\uparrow & if \quad x=3\\ 0 & else\end{cases}$	ח"ח	f_2	$N-\{3\}$	0	לא חישובית וזה מה

* כאשר שואלים על קבוצת תוכניות שמחשבת פונק' כלשהי היא בשום מקרה לא במ"ש מכיון שכדי ש הפונק' שווה היא מחשבת היא פונק' שע"מ הבדיקה המתבצעת היא בדיקה אינסופית. כאשר שואלים על קבוצת התוכניות המחשבת פונק' כשדבר לא במ"ש!
* חישוב פונק' פירושו לרשום באופן מלא נוסחת התשובה ונוסחת הביטוי המלא שלה לכל x.
* יש שואלים על קבוצת התוכניות המחשבות פונק' חישובית חלקית - אז הקבוצה הזו תהיה שווה ל-N - כל המספרים הטבעיים, כי כל מס' טבעי מצפין תוכנית, כל תוכנית מחשבת פונק' חישובית חלקית. לכן הקבוצה הזו היא חישובית כריעה.
  מדובר רק בפונק' חישובית חלקית כללית ולא דוגמא ספציפית כי אז מדובר במו' עצירה.
* שאלה: קבוצת התוכניות שקידודן x מ"ג: היא קבוצה במ"ש - נכון היא קבוצה חישובית ולכן גם במ"ש.
* תוכנית שעוצרת לכל קלט בהכרח מחשבת פונק' חישובית. (TOT מחשבת פונק' חישוביות)
* קבוצת התוכניות שעוצרות לכל קלט (הקבוצה TOT) היא לא במ"ש כי הבדיקה אינסופית.

בס"ד

חישוביות - חשוב לזכור!

* תוכניות שמחזירות לכל x את הקבוע 7, מחשבות את הפונק' $f(x)=7$. $B=\{n \mid \Phi(x,n)=7\}$
* תוכניות שלא עוצרות על הקלט 7 $B=\{n \mid \Phi(7,n)\uparrow\}$
* תוכניות שעוצרות על 7 קלטים בדיוק. $B=\{n \mid 7=|W_n|\}$
* תוכניות שעוצרות על הקלט 7 (בין היתר), כלומר, 7 שייך לקבוצת הקלטים שעליהם היא עוצרת $B=\{n \mid 7\in W_n\}$
* קבוצת התוכניות העוצרות רק על הקלט 7 $B=\{n \mid \{7\}=W_n\}$
* הפרדיקט המגדיר את האוסף אומר x השייך לאוסף W_5 $W_5=\{\overset{\text{קלט}}{x} \mid \exists z\, stp(x,5,z)\}$

תזכורת: W_n - קבוצת הקלטים עליהם עוצרת התוכנית n.

* פרדיקט אופייני - פרדיקט המחזיר 1 על איברי הקבוצה
* לכל קבוצה קיים פרדיקט אופייני!!
* חיתוך של 2 קבוצות זרות הוא הקבוצה הריקה ϕ שהיא חישובית והפרדיקט האופייני שלה הוא $f(x)=0$, והפונק' היא: $n(x)=0$
* טענה: החיתוך של שתי קבוצות שאינן במ"ן לא יהיה קבוצה חישובית - לא נכון! יש 2 הקבוצות הן זרות אז החיתוך שלהן הוא $\{\}$ שהיא חישובית.
* יש קבוצה שהיא במ"ן לא חישובית => שהמשלימה לא במ"ן.
* יש קבוצה במ"ן וגם שהמשלימה במ"ן => הקבוצה חישובית.
* טענה: לכל קבוצה במ"ן קיים פרדיקט אופייני חישובי המגדיר אותה - לא נכון! יש הקבוצה היא במ"ן לא חישובית אז לא קיים לה פרדיקט חישובי אלא רק פרדיקט.
* דוגמא לקבוצה חישובית: $Even=\{x \mid (2|x)\}$

$$f(x)=\begin{cases}1 & \text{if } (2|x)\\ 0 & \text{else}\end{cases}$$ פרדיקט אופייני חישובי!

תוכנית המחשבת:
```
T(x): if (2|x) GoTo B
      GoTo E
[B]: Y ← 1
```

מק"ש: $x\in A \Rightarrow T(x)=1$, $x\notin A \Rightarrow T(x)=0$

הפרדיקט והתוכנית המחשבת אותו תמיד ומחזירה 1 על איברי הקבוצה, ו0 על אלו שלא שייכים לקבוצה.

* משפט: A רקורסיבית (חישובית) $\Leftrightarrow$ A ו $\bar{A}$ במ"ן.
* משפט: אם A ו B הן במ"ן, אז גם $A\cup B$ ו $A\cap B$ הן במ"ן
* הקבוצה במ"ן לא סגורה למשלים! (אילו היא הייתה סגורה למשלים אז כל קבוצות במ"ן היו חישוביות).
* מחלקת הקבוצות הרקורסיביות (חישוביות) סגורה למשלים, איחוד וחיתוך.
* מחלקת הקבוצות הנ"ל (במ"ן) סגורה לאיחוד וחיתוך, אך לא למשלים.
* התנאים השקולים להגדרת קבוצה לא ריקה B שקולים: B במ"ן $\equiv$ B היא טווח של פונק' חישובית $\equiv$ B היא טווח של פונק' רקורסיבית חלקית

חישוביות - המשך סיכום

- הצפנת מספרים ותוכניות - המטרה: לתת "קוד" מספרי חח"ע לכל מיני סוגי אובייקטים, כדי שנוכל לדבר על קבוצות של מספרים טבעיים מהקשרים שונים.

1. קידוד/צימוד - פונק' לקידוד זוג מספרים טבעיים: $\langle x,y\rangle = 2^x(2y+1)-1$

הפעולות: נסמן $z=\langle x,y\rangle$ ונגדיר $r(z)=y$, $l(z)=x$.

2. הצפנת גדל - שיטה המאפשרת להצפין סדרה של n ערכים

דוגמא: (בהינתן 20, איך מגיעים לסדרה? ע"י חלוקה במספרים הראשוניים לפי הסדר). $[2,0,1] = 2^2\cdot 3^0\cdot 5^1 = 20$

פונקציות חשובות לעבודה עם מס' גדל:

פעולות על גדל:

חילוץ של מעריך החזקה של המס' הראשוני ה-i (האיבר ה-i בסדרה) בדוגמא שלנו: $*\ (x)_i=(20)_2=0$

פונק' אורך של סדרה. מתאימה את אורך הסדרה שהמס' הוא מס' גדל שלה. בדוגמא שלנו: $*\ Lt(x)=\overset{\text{אורך הסדרה}}{Lt(20)}=3$

- נגדיר את מספר גדל של הסדרה הריקה כ1. (הערה: אין סדרה שמס' גדל שלה הוא 0)

3. הצפנת תוכניות - $\langle \overset{\text{תווית שעוברת}}{a}, \langle \overset{\text{מס' הפקודה}}{b}, \overset{\text{המשתנה הנוכחי}}{c}\rangle\rangle$

- איך מוציאים תוויות: אם אין תווית a=0, אחרת: (ממספרים אותן) $A_1 B_1 C_1 D_1 E_1 A_2 B_2 C_2 D_2 E_2 \ldots$
- איך מוציאים פקודות:
  - אם הפקודה היא הפקודה הריקה ($V\leftarrow V$) אז $b=0$
  - אם הפקודה היא תוספת ($V\leftarrow V+1$) אז $b=1$
  - אם הפקודה היא חיסור ($V\leftarrow V-1$) אז $b=2$
  - אם הפקודה היא קפיצה (קפיצה מותנית) ($IF\ V\neq 0\ GOTO\ L$) אז $b=$ מס' התווית $+2$
- איך מוציאים משתנים: (רק כמו מוציאים אותם) $Y X_1 Z_1 X_2 Z_2 X_3 Z_3 \ldots$

- בקידוד תוכנית מורידים 1 כדי שיהיה לנו 0 שמבטא את התוכנית הריקה.

מספר הפקודות

$$\underbrace{\underbrace{2^{\square}\cdot 3^{\square}\cdot 5^{\square}\cdot 7^{\square}\ldots}_{\text{גדל (של תוכנית)}}-1}_{\text{קידוד (של התוכנית)}}$$

! תוכנית שקידודה זוגי ⇐ המס' גדל שלה אי זוגי ⇐ מעריך החזקה של 2 הוא 0 ⇐ קידוד הפקודה הראשונה הוא 0 $\langle 0,\langle 0,0\rangle\rangle$ ⇐ הפקודה הראשונה היא פקודה ריקה $Y\leftarrow Y$.

- לכל פונק' חשיבות חלקית קיימת תוכנית שקידודה זוגי, מכיון שאם קיימת לפונק' הנ"ל תוכנית שקידודה אי"ז (אי זוגי) שמחשבת אותה אזי בקלות ניתן להוסיף פקודה ריקה $Y\leftarrow Y$ שקידודה $\langle 0,\langle 0,0\rangle\rangle$ 0 שמייצגת גדל זוגי. מכיון שכדי שהמס' גדל יהיה אי זוגי פירושו שהחזקה של 2 היא 0.

! חשוב לזכור! 1-גדל = תוכנית ⇐ 1+קידוד = גדל ⇐ יש צורך להשתמש בLt על קידוד: $Lt(x+1)$.

! הפונק' $(x)_i$ ו-$Lt(x)$ הן רק שעובדות על גדל. יש צורך להמיר על קידוד/תוכנית נוסיף 1 לx.

חישוביות - המשך סיכום

* תבנית לפונק' שמקבלת פרמטר אחד:

$h(0)=k$

$h(t+1)=g(t,h(t))$

* תבנית לר"פ לפי משתנים:

$h(x,0)=f(x)$

$h(x,t+1)=g(t,h(x,t),x)$

- עושה שימוש בתבניות גם אם בכלל אין צורך ברקורסיה. (הפונק' "הקודם" אין צורך ברקורסיה).
- כאשר יש חוסר התאמות במס' הפרמטרים בד"כ נעזר בהטלה זו.

* הפונק' המורכבת - מקבלת את אותו מס' הפרמטרים של הפונק' הפנימית בדיוק בהרכבה ובאותו הסדר.

- פונק' המינוס השלם - הפונק' הזו היא שלמה, חישובית-רקורסיבית, רקורסיבית פרימיטיבית.
- פונק' החיסור בערך מוחלט ר"פ (מוכיחים ע"י הרכבה), פונק' העצרת ר"פ (מוכיחים ע"י רקורסיה).

* הדרך להפוך הגדרה רגילה של פונק' להגדרה רקורסיבית:

נראה כיצד להביע את $f(n+1)$ באמצעות $f(n)$

- נבדוק מה ההפרש בין $n+1$ לn:

$f(n+1)-f(n)=♡$

$f(n+1)=f(n)+♡$

ואז נגדיר פונק' שיש בה תנאי עצירה ושימוש ב $n+1$ בנוסחת נסיגה.

∘ מכיון שכל פונק' חישובית היא לא בהכרח ר"פ לכן כדי להוכיח ר"פ לא ניתן להסתפק בכך שהפונק' היא חישובית, אבל כל פונק' ר"פ היא גם חישובית, לכן:

- אם אנחנו נדרשים להוכיח שפונק' כלשהי היא חישובית, יש לנו 2 אפשרויות:

1. להוכיח שהיא חישובית (נראה שקיימת תוכנית בשפת S שתמיד עוצרת לכל קלט x מחזירה פלט תקף).
2. להוכיח שהיא רקורסיבית פרימיטיבית (בכך שניתן לייצג אותה באמצעות הרכבה/רקורסיה פרימיטיבית של פונק' ר"פ בלבד).

∘ כאשר אנחנו מתבקשים להוכיח שפונק' כלשהי היא חישובית/רקורסיבית) - נראה תוכנית בשפת S שתמיד עוצרת לכל ערך x ומחזירה פלט y כנדרש. מה עושים?

1. מציגים אי שוויון.
2. בוחרים לציין תוכנית את המאקרו שבו y גדול מ...
3. מסדרים את אי השוויון באמצעות העברת אגפים ופעולות אריתמטיות כך שיהיה מורכב רק מפעולות חשבון.
4. כותבים תוכנית בשפת S שעוצרת לכל קלט ומחזירה במשתנה הפלט את הערך הנשאר שמקיים את אי השוויון המתוקן.
5. מוכיחים שהתוכנית תמיד עוצרת ומחזירה את הערך הנדרש.

חישוביות - תרגול סיכום

* מינימליזציה חסומה - פונק' תהיה ר"פ אם היא ניתנת לייצוג ע"י מינימליזציה חסומה, שבה גם החסם וגם הפרדיקט הן פונקציות ר"פ.

$$f(x) = \min_{t \le \underset{\text{חסם}}{y}} P(x,t)$$

t הראשון שמקיים את הפרדיקט - הכי קטן!

* הפרדיקט ר"פ
* החיפוש תחום (בפונק' ר"פ)
* אם תחום: בתחום החיפוש הזה אין ערך שמקיים את הפרדיקט - מחזירים 0 (או החסם, תלוי בהגדרה).

- בשאלות שבהן ניתן לנסח במפורש את החסם זה אומר שהפונק' ניתנת לייצוג באמצעות חסם (נוסיף במפורש את החסם) והפונק' תהיה ר"פ.
- בשאלות שבהן צריך לחשב משהו שצריך לעבור בו על כל הערכים, הרבה שאין חסימה לא ניתן לחשב סופית. אלא חייב שיהיה לנו חסם כדי שיחסום את תהליך החיפוש.
- פונק' שיש בהן גידול כלפי מעלה/מטה כדאי לנסות לפתור באמצעות מינימליזציה חסומה.
- כאשר צריך לכתוב חסם הדוק מותר להגדיל אותו! אבל אסור להקטין אותו!
- חשוב לשים לב שבחסם עצמו כותבים ביטוי ר"פ!
- כשמגדירים חסם הערך שמוחזר מהמינימליזציה חייב להיות מוגדר! צריך להחזיר את הערך הראשון שמקיים את הפרדיקט כאשר תחום החיפוש חסום.
- כדי לזכור: כאשר מינימליזציה עושה שימוש ברקורסיה אין צורך להוכיח שהרקורסיה ר"פ.

* מקסימיזציה חסומה - הערך המקסימלי הכי גדול שמקיים את הפרדיקט כאשר החיפוש תחום.
- הערך הגדול ביותר שמקיים את הפרדיקט יהיה הערך שמרחקו בינו לבין החסם הוא מינימלי.
- מה מקסימיזציה חסומה? אם פונק' המחזירה מס', z כלשהו, שמרחקו של מהחסם הוא מינימלי: בגלל שהוא חסומה, אם אין קיים יוחזר החסם.

כלומר, מקסימיזציה חסומה מוגדרת ע"י מינימליזציה חסומה:

$$g(x) = \underset{\text{חסם}}{h(x)} \dot{-} \underbrace{\min_{z \le h(x)+1} R(x, h(x) \dot{-} z)}_{\text{המרחק}}$$

בס"ד

חישוביות - תרגול רקורסיה

* תבנית הקלאסית של רקורסיה פרימיטיבית (פונק' המקבלת פרמטר אחד).

$f(0) = k$ — התבנית של הרקורסיה הפרימיטיבית כזו בעניין מתכ"ש!

$f(t+1) = g(t, f(t))$ — 1 תנאי עצירה אחד. משתמשת רק בערך קודם אחד.

* סדרת פיבונאצ'י:

יש לנו 2 בעיות!

* 2 תנאי עצירה (שני ערכי בסיס)
* משתמשת ב-2 ערכים קודמים

$f(0) = 0$

$f(1) = 1$

$f(n+1) = f(n) + f(n-1)$

- ניתן להוכיח שהיא ר"פ, ואולם לא ניתן שימוש בתבנית של הרקורסיה הפרימיטיבית הקלאסית.

דרך הפתרון:

1. נגדיר פונק' עזר שתשתמש בזוג/קידוד ותשמור לנו בכל שלב 2 ערכים. פונק' העזר תהיה ר"פ קלאסית.
2. נבטא את f בעזרת הרכבה של פונ' ר"פ.

נגדיר פונק' עזר בשם g, נגדיר אותה בצורה מפורשת:

נגדיר ראשית:

$g(0) = \langle 0, 1 \rangle = 2^0(2\cdot 1+1)-1 = 2$

$g(t+1) = \langle r(g(t)), r(g(t)) + l(g(t)) \rangle$ $\quad plus(x,y) = x+y, \quad q(x,y) = \langle x,y \rangle$

בצורה קלאסית:

$g(t+1) = q(r(\pi^2_2(t, g(t))), plus[r(\pi^2_2(t,g(t))), l(\pi^2_2(t,g(t)))])$

- g היא ר"פ קלאסית, היא מקיימת תנאי עצירה אחד, משתמשת רק בערך קודם אחד ומתאימה לתבנית הקלאסית ע"כ אותה כהרכבה והרקורסיה פרימיטיבית של פונק' ר"פ בלבד! כעת נבטא את f תוך שימוש ב-g

$f(n) = l(g(n))$

f היא ר"פ מכיוון שהיא נוצרת ע"י הרכבה של פונק' ר"פ בלבד.

- נסכם את מאפייני פיבונאצ'י. הם כדלקמן:

$f(0)$	$f(1)$	$f(2)$	$f(3)$	$f(4)$	$f(5)$
0	1	1	2	3	5

$\underbrace{0 \; 1}_{g(0)} \quad \underbrace{1 \; 1}_{g(1)} \quad \underbrace{1 \; 2}_{g(2)} \quad \underbrace{2 \; 3}_{g(3)} \quad \underbrace{3 \; 5}_{g(4)}$

1. 2 תנאי עצירה (2 ערכי בסיס)
2. משתמשת ביותר מערך קודם אחד.
3. משתמשת במס' קבוע של ערכים קודמים שאינו משתנה. (שני ערכים לאורך כל הדרך).

חישוביות - רקורסיה

* כאשר פונק' מוגדרת ע"י רקורסיה ויש 3 דרכים להוכיח שהיא ר"פ.
1. תלויה רק בערך קודם אחד → תבנית רגילה של רקורסיה פרימיטיבית.
2. תלויה ביותר מערך קודם אחד, אבל מס' קבוע של ערכים קודמים (2 או 3)
→ הוכחה באמצעות פונק' עזר שעושה שימוש בקידוד (כמו פיבונאצ'י).
3. תלויה בכל אוסף ערכים קודמים → הוכחה באמצעות פונק' עזר שעושה שימוש בצבר.

- דוגמה לפונק' שנשענת על ערך קודם אחד (למרות שנראה בהתחלה שהיא צריכה אוסף ערכים).

$f(0)=2$

$f(1)=5$

$f(n+2)=\sum_{j=0}^{n+1}(j+1)^2\cdot(f(j))^3$

$n=0:\ f(0+2)=f(2)=\sum_{j=0}^{1}(j+1)^2(f(j))^3=\overbrace{\underbrace{1^2\cdot f(0)^3}_{j=0}+\underbrace{2^2 f(1)^3}_{j=1}}^{\Delta}=1f(0)^3+4f(1)^3$

$n=1:\ f(1+2)=f(3)=\sum_{j=0}^{2}(j+1)^2(f(j))^3=\underbrace{\underbrace{1f(0)^3}_{j=0}+\underbrace{4f(1)^3}_{j=1}}_{\Delta}+\underbrace{9f(2)^3}_{j=2}$

למרות ש $f(n+2)$ נשענת על n ערכים קודמים, בגלל שאין שינוי במקדמים והם נשארים אותו הדבר, יוצא שכדי לחשב צריך לזכור רק את הקודמת כולל הכפלה. ואפשר לומר שניתן להגדיר את הפונק' הזו ע"י רקורסיה פרימיטיבית שנשענת על ערך קודם אחד. ולכן היא מתאימה לסוג הראשון.

- צריך להמיר את הנוסחה לנוסחה הנשענת על ערך קודם אחד.

דוגמה: $f(n+1)=f(n)+♡$, לכן נרצה להביא את ♡ עד שיהיה תלוי רק בערך קודם יחיד.

$$f(n+1)=\sum_{j=0}^{n}(j+1)^2(f(j))^3=\underbrace{\sum_{j=0}^{n-1}(j+1)^2\cdot f(j)^3}_{f(n)}+\underbrace{(n+1)^2\cdot f(n)}_{♡}=\boxed{\underbrace{f(n)+(n+1)^2\cdot f(n)^3}_{f(n+1)}}$$

נוסחה רקורסיבית שקולה לנוסחה המקורית שנשענת על ערך קודם אחד.

- כדי לקבל את $f(1)$ בנוסחה החדשה צריך להציב $n=0$ ונקבל 10 ולא 5. ולכן לא תואם לנוסחה המקורית.

$f(0)=2$

$$f(n+1)=\begin{cases}5 & n=0\\ 1\cdot 2^3+2^2\cdot 5^3=508 & n=1\\ f(n)+(n+1)^2\cdot(f(n))^3 & n\geq 2\end{cases}$$

שינוי תנאי עצירה יחיד והוספה למקרים. בגלל שהשינוי המתאים לפונק' החדשה צריך לבטל את הערך הראשון שהוא תואם בנוסחאות.

חישוביות - רקורסיה - המשך

הגדרה של פונק' רקורסיבית המשתמשת ביותר מערך קודם:

$f(0) = 2$

$$f(n+1) = \sum_{j=0}^{n} (\underset{\text{המשתנה}}{n} - j + 1)^2 \cdot f(j)^3$$

g היא פונק' עזר שמספקת את ה f ים

$n=0: f(1) = \underline{1} \cdot f(0)^3$

$n=1: f(2) = \underline{4} \cdot f(0)^3 + \underline{1} \cdot f(1)^3$

$n=2: f(3) = \underline{9} \cdot f(0)^3 + \underline{4} \cdot f(1)^3 + \underline{1} \cdot f(2)^3$

נגדיר פונק' עזר רקורסיבית פרימיטיבית קלאסית!

הפונק' הזו תהיה מוגדרת על ערך קודם בלבד והאיבר שישמש בקבל שישמור לי בכל פעם n ערכים: את כל ה f ים מ $f(0)$ עד ה f הנוכחית

תהי g פונק' העזר:

$g(0) = [2] = 2^2 = 4$

$$g(n+1) = [(g(n))_1, \dots, (g(n))_{n+1}, \sum_{j=0}^{n} (n-j+1)^2 ((g(n))_{j+1})^3]$$

דוגמא:

$$n=6: g(6+1) = g(7) = [\underbrace{\underset{i=1}{f(0)}, \underset{i=2}{f(1)}, \underset{i=3}{f(2)}, \underset{i=4}{f(3)}, \underset{i=5}{f(4)}, \underset{i=6}{f(5)}, \underset{i=7}{f(6)}}_{\text{7 ערכים קודמים מהרשימה הקודמת}}, \underset{i=8}{f(7)}]$$

מתייחס לכל המקומות הראשונים את כל פונק' ה f הקודמות לחישוב $f(7)$ ובמקום ה 8 מחשבת מתוך הקודם שבאותו את הערך החדש $f(7)$ וכך $g(8)$ של הרשימה הבאה תוכל להשתמש רק בה!

ערך חדש המחושב ע"י הפעולה הנוכחית תוך שימוש בכל הערכים הקודמים מהרשימה

$g(8)$ תכיל 8 ערכים קודמים $f(0)$-$f(7)$ ותקבל את כולם מהשורה של $g(7)$

פונק' העזר g היא רפ' , שכן היא מוגדרת תוך שימוש ברקורסיה פרימיטיבית קלאסית והרכבות של פונק' רקורסיביות פרימיטיביות.

$g(n+1)$ אשר שימש רק ב $g(n)$.

כעת נוכיח ש f המקורית היא רפ' , נבטא את f תוך הרכבה של פונק' רפ' בלבד.

$f(n) = (g(n))_{n+1}$

$f(7)$ נמצא במקום האחרון של הרשימה המוחזרת ע"י $g(7)$ שהוא מקום השמיני: $f(7) = (g(7))_8$

f היא רפ' לפי שמוגדרת תוך שימוש בהרכבות של פונק' רפ' בלבד. $g(n)$, $(x)_i$

חישוביות - לכסון להוכחת לא במ״ר

* כלי להוכחת לא במ״ר לא תמיד מתאים.

הוא מתאים למקרה של תוכניות שמחשבות דווקא פונקציות שלמות!

ולכן, יש לדבר בתוכניות שמחשבות פונק׳ לא שלמות נצטרך להוכיח לא במ״ר בדרך אחרת.

- הוכחה של הלכסון זהה תמיד, פרט לפונק׳ h שאותה נצטרך להגדיר.

לדוגמה: * A = {תוכניות המחשבות פרדיקט חישובי} $\longrightarrow h(x) = 1 \dot{-} \overbrace{\Phi(x, g(x))}^{0/1}$

* $C = \{n \mid n \in TOT \wedge Range(\Phi_n) \subseteq \{7,8\}\} \longrightarrow h(x) = 15 \dot{-} \overbrace{\Phi(x, g(x))}^{7/8}$

- הערה: כדי לדבר על הפונק׳ שאותה מחשבת התוכנית p, אפשר לכתוב כך: $\Psi_p(x)$ או כך: Φ_p.

- נוכיח בעזרת לכסון כי $\bar{K}$ איננה במ״ר

תהי A קבוצה בת מניה כזו: $A = \{W_0, W_1, W_2, W_3 \dots\}$

כמובן שהיא בת מניה כי לכל W_i מתאים מס׳ טבעי שהוא קידוד התוכנית (i).

הקבוצה A היא קבוצה בת מניה וכל אחד מאבריה הוא קבוצה במ״ר

נניח בשלילה ש $\bar{K}$ היא קבוצה במ״ר.

כלומר, יש כך ש $\bar{K}$ בהכרח איבר מתוך הקבוצה A וחייבת להשתייך אליה.

$\Downarrow$

כלומר, יש תוכנית i המקבלת אותה ובהכרח $\bar{K} = W_i$

(הבהרה: קבוצה במ״ר היא תחום ההגדרה של פונק׳ חח כלשהי, בהכרח קיימת תוכנית המחשבת את הפונק׳ הזו $\rightarrow$ עוברת על כל האיברים שבקבוצה).

כעת, ניקח לדוגמה את $i=3$, נניח ש $\#(p)=3$ כאשר p היא התוכנית שמקבלת את $\bar{K}$.

כלומר מתקיים $\bar{K} = W_3$. נבדוק את הזוג $(3,3)$ באלכסון.

נבדוק מה קורה בזוג $(3,3)$ Φ (בדיקה באלכסון).

ישנן 2 אפשרויות או ש: $\Phi(3,3)\downarrow$ או ש: $\Phi(3,3)\uparrow$ נבחן את שתיהן:

אפשרות ב׳ | אפשרות א׳

$\Phi(3,3)\uparrow \Rightarrow 3 \notin W_3 \Rightarrow 3 \notin \bar{K} \Rightarrow \Phi(3,3)\downarrow$

$\Phi(3,3)\downarrow \Rightarrow 3 \in W_3 \overset{(**)}{\Rightarrow} 3 \in \bar{K} \Rightarrow \Phi(3,3)\uparrow$

(**) אבל $\bar{K} = W_3$

סתירה!

לכל i שנבחר תמיד $\bar{K} \neq W_i$ בזוג של i שנבחר יתקבל החישוב ההפוך והגענו לסתירה.

מסקנה: הקבוצה A מאגדת בתוכה, את כל האיברים שבמ״ר, כלומר הקבוצה $\bar{K}$ לא שייכת לקבוצה.

$\bar{K}$ איננה איבר בקבוצה A, מאחר ואיברי הקבוצה הן קבוצות במ״ר בעצמן הרי ש $\bar{K}$ איננה קבוצה במ״ר.

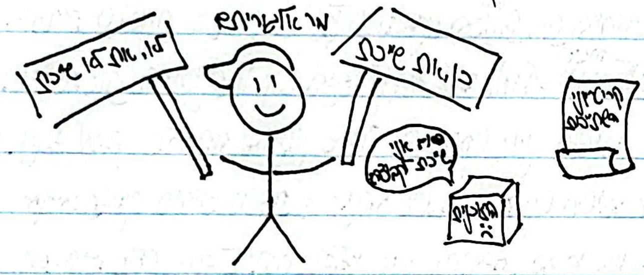
חישוביות - בעיית העצירה

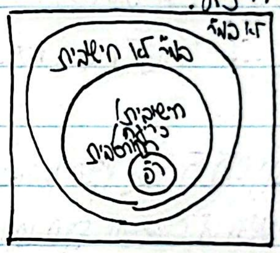
- כלל: בדיקה חייבת להיות תהליך סופי שמסתיים.
- דברים שמחשב לא יכול לעשות:

1. לבדוק אי עצירה של תוכנית. 2. לבצע בדיקה אינסופית.

* קבוצה חישובית / כריעה / רקורסיבית - קיים אלגוריתם שמסוגל לבדוק גם שייכות וגם אי שייכות. הפרדיקט האופייני שלה הוא פונק' חישובית (כריעה = ניתנת להכרעה).
* במ"ר לא חישובית - קבוצה שקיים אלגוריתם שמסוגל לבדוק שייכות אליה, אבל לא קיים אלגוריתם שמסוגל לבדוק אי שייכות. (במ"ר / נע"ר = בר מניה רקורסיבית).
* לא במ"ר - קבוצה שלא קיים אלגוריתם שיכול לבדוק שייכות אליה.

היררכיה:





- FIN (תוכניות שעוצרות רק על מס' סופי של קלטים) + INF (תוכניות שעוצרות על אינסוף קלטים) - קבוצות משלימות ששתיהן לא במ"ר.
- TOT (תוכניות שעוצרות לכל קלט) ו-$\overline{TOT}$ קבוצות משלימות לא במ"ר.
- EMPTY (תוכניות שלא עוצרות לשום קלט) ו-$\overline{EMPTY}$ (תוכניות שקיים קלט אחד לפחות שעליו הן עוצרות) שתיהן קבוצות משלימות, EMPTY - לא במ"ר, $\overline{EMPTY}$ - במ"ר לא חישובית.

! יש קבוצה שהיא במ"ר לא חישובית, אז המשלימה בהכרח לא במ"ר.

! יש קבוצה לא במ"ר, שהמשלימה יכולה להיות במ"ר לא חישובית או לא במ"ר אבל היא בהכרח אינה חישובית.

! יש קבוצה חישובית-כריעה שהמשלימה גם חישובית כריעה!

- נשים ♥ - כדי לבדוק שתוכנית עוצרת על מס' סופי של קלטים - התהליך הבדיקה הוא אינסופי!
- כאשר בדיקה של חלק מאברי הקבוצה לא אפשרית - כל הבדיקה כולה לא אפשרית!
- כלל אצבע - כאשר כתוב "התוכניות שלכל x..." זו תמיד קבוצה לא במ"ר, אלא אם כן מדובר בתכונה שמתקיימת בכל התוכניות או בשום תוכנית ואז היא חישובית!

דוג': תוכניות שלכל x מחשבות פונק' חח - נתון שנכון תמיד - חישובית-כריעה.

- תזכורת: פרדיקט אופייני - פונק' שלמה שטווח הערכים שלה הם 0,1, הפרדיקט האופייני מתאים 1 לכל האיברים שבקבוצה ומתאים 0 לאלו שאינם בקבוצה.

חישוביות

* קבוצה היא חישובית - כריעה - רקורסיבית אם״ם הפרדיקט האופייני שלה הוא חישובי רקורסיבי.
* קבוצה היא רה״ אם״ם הפרדיקט האופייני שלה הוא רה״.
* זכרון הוא שיטה להוכיח שפרדיקט כלשהו אינו חישובי רקורסיבי.
* מוכיחים פורמלית שקבוצה חישובית ע״י הצגת האלגוריתם שמכריע אותה (האלגוריתם המכריע עוצר תמיד - רק שלכל איבר בקבוצה הוא עוצר ומחזיר 1, ולכל איבר שאינו בקבוצה עוצר ומחזיר 0.
וכן הוא מתאר תוכנית המחשבת את הפרדיקט האופייני של הקבוצה.

- איך נראה זכרון להוכחת אי חישוביות?

1. מניחים בשלילה שהפרדיקט חישובי
2. יוצרים תוכנית בשפת S בעזרת תבנית המכילה את הפרדיקט הנ״ל:
3. מריצים את התוכנית על קלט המתאר את הקוד שלה ומקבלים (סתירה) ...
4. נובע מכך שההנחה שהתחלנו שגויה והפרדיקט אינו חישובי.

```
[A] if P(x) GOTO A
 ...
 ...
 ...  (שורות קוד בהמשך לצורך)
```

הערה: תוכנית חוקית בשפת S שעושה שימוש בפרדיקט חישובי ואמורה להתנהג באופן תקין עבור כל קלט נתון. אם קיים קלט אחד לפחות שעבורו היא מתנהגת באופן משונה, הרי שמשמעות הדבר היא שהפרדיקט לא חישובי. על קלט כלשהו חייבת להיות לתוכנית התנהגות אחת בדיוק. לא יתכן שאם עוצרת עליו הרי שנתקעת עליו וגם נתקעת עליו הרי שעוצרת. וכך נוצרת הסתירה. פרדיקט אופייני שאינו חישובי גורר שהקבוצה שאותה הוא מאפיין אינה חישובית, וכן קבוצה שאינה חישובית, גם הפרדיקט האופייני שלה לא חישובי.

- השוני הוא ניסוחו התאורטי שעומד בפני עצמו, קיימות לתוכנית בה קיים ישות שמסבך אותה ולכן קיימת פונק׳ שלמה שאינה חישובית - לדוגמה הפרדיקט HALT - פונק׳ שלמה וגם לא חישובית.

* לקבוצת חישוביות - כריעה קיים אלגוריתם המכריע אותה - עוצר ומחזיר 1 על איברי הקבוצה ועוצר ומחזיר 0 על אלו שאינם בקבוצה (כלומר, עוצר תמיד ונבדל בערך השונים שמחזיר).
* יש קבוצה A היא במ״ן אם קיים עבורה אלגוריתם P המקבל אותה המקיים: $P(x)\downarrow \iff x \in A$
האלגוריתם שמקבל את קבוצת הבמ״ן עוצר על איברי הקבוצה ורק עליהם
(נכון גם על במ״ן חישוביות וגם על במ״ן לא חישוביות).
* לכן המסקנה כל קבוצה חישובית כריעה היא גם במ״ן. ולכן אם קיים לקבוצה אלגוריתם שמכריע אותה אז גם קיים לה אלגוריתם שמקבל אותה.
* משפט: פונק׳ היא חישובית חלקית אם״ם היא מתקבלת מפונק׳ הבסיס ע״י מס׳ סופי של הרכבות, רקורסיות פרימיטיביות ומינימיזציות.

- כל קבוצה חישובית כריעה היא גם במ״ן.

! מס׳ השורות בתוכנית תמיד סופי, מס׳ הצעדים הוא אינסופי.

חישוביות - בעיית העצירה

* כל תוכנית שצריך לבדוק בה אי עצירה אינה ניתנת לחישוב, אפילו אם מדובר על בדיקה שמתבצעת רק על חלק מאיברי הקבוצה.
* בדיקה אינסופית אינה ניתנת לחישוב. כדי לבדוק שתוכנית עוצרת על מס׳ סופי של קלטים - כלו׳ שהיא לא עוצרת על שאר הקלטים - מצריך הבדיקה אינסופית.
* תוכניות שלא עוצרות על הקלט x - אי עצירה = לא במ״.
* תוכניות שעוצרות על הקלט x - (כמו חיבור) מדובר בבדיקת עצירה בלבד - במ״ לא חישובית.
* תוכניות שעוצרות רק על הקלט x - כלומר, לא עוצרות על שאר הקלטים. אי עצירה = לא במ״.
* תוכניות שעוצרות על x קלטים בדיוק / הקבוצה FIN (תוכנית שעוצרת על מס׳ סופי של קלטים) - מדובר באי עצירה על שאר הקלטים, מצריך הבדיקה אינסופית = לא במ״.
* תוכניות שקיים קלט שעליו הן עוצרות ($\overline{EMPTY}$) = במ״ לא חישובית.
* תוכניות שעוצרות על קלט הגדול מקידוד שלהן (x) - (כמו חשבון) = במ״ לא חישובית.
* תוכניות שעוצרות לכל קלט (TOT) / תוכניות שעוצרות על אינסוף קלטים (INF) - מצריך הבדיקה אינסופית = לא במ״.
* תוכניות שקידודן זוגי (EVEN) / אי זוגי (ODD) - הבדיקה נעשית על הקידוד של התוכנית ולא על התוכנית עצמה (אין בדיקת עצירה וכו׳) = חישובית - כריעה.
* תוכניות שלכל x מחשבות את הפונק׳ f(x) - בדיקה אינסופית = לא במ״.
* הקבוצה המשלימה לזו - תוכניות שקיים קלט שעבורו הן לא מחשבות את הפונק׳ f(x) - לא במ״.
* בכל פעם שכתוב "תוכניות שלכל x..." זו תהיה קבוצה לא במ״, אלא אם כן מדובר בתכונה שמתקיימת בכל תוכנית או באף תוכנית ואז הן חישוביות.
* תוכניות שמחשבות פונ׳ שתחום ההגדרה שלהן הוא הקבוצה {x} - אלו תוכניות שעוצרות רק על הקלט x - כלו׳ על שאר הקלטים הן לא עוצרות - לא במ״.
* אם קבוצה במ״ לא חישובית משלימתה לא במ״!
* אם קבוצה במ״ וגם משלימתה במ״ הקבוצה חישובית!

חישוביות- פונקציות ופרדיקטים חשובים.

* **הגדרת הפרדיקט $HALT(x_1,x_2)$:**

$HALT(x_1,x_2)=1$ אם"ם התוכנית שמספרה x_2 עוצרת על הקלט x_1.

☑ פרדיקט ☑ פונ' שלמה

☒ לא חישובית, ועולם לא תהיה.

$$halt(x_1,x_2) = \begin{cases} 1 & \text{אם התוכנית } x_2 \text{ עוצרת על הקלט } x_1 \\ 0 & \text{אם התוכנית } x_2 \text{ לא עוצרת על הקלט } x_1 \end{cases}$$

- הפרדיקט $HALT(x_1,x_2)$ אינו ניתן לחישוב!

* **הגדרת התוכנית האוניברסלית $\Phi^{(n)}(x_1 \dots x_n, x_{n+1})$:**

התוכנית האוניברסלית מריצה את התוכנית x_2 על הקלט x_1 $\qquad \Phi(x_1,x_2)$

משפט האוניברסליות: לכל n הפונ' $\Phi^{(n)}(x_1 \dots x_n, x_{n+1})$ היא פונ' רקורסיבית חלקית.

- התוכנית האוניברסלית לא בהכרח עוצרת כי היא תלויה (בקלט) בתוכנית שאותה היא מריצה.

! אסור לשים את התוכנית האוניברסלית אחרי IF כי היא חח (לא בהכרח עוצרת).

* **הגדרת הפרדיקט $STP(x,y,z)$:**

האם התוכנית x ברצתה על הקלט y עוצרת תוך z צעדים. $\qquad stp(\overset{\text{קלט}}{y}, \overset{\text{תוכנית}}{x}, \overset{\text{צעד}}{z}) = 0/1$

- אפשר לבדוק שתוכנית לא עוצרת, תוך מס' צעדים סופי.

- צעד = מעבר של פס' פקודה קוד בסיסית = רגע.

- ברגע שיש רגע עצירה מאותו רגע והלאה יש העצירה והיא מתוך t הצעדים עד לכל שאר הרגעים מאותו הרגע והלאה 1.

- stp הוא פרדיקט ר"פ.

* **הגדרת הפונ' $Snap(x_1 \dots x_n, y, t)$:**

פונ' ר"פ $\qquad Snap(x_1 \dots \overset{\text{קלט}}{x_n}, \overset{\text{תוכנית}}{y}, \overset{\text{צעד}}{t}) = \langle v, m \rangle$

גודל של מצבע התוכנית, מס' ההוראה הבאה לביצוע

$m = 2^{y} \cdot 3^{x_1} 5^{z_1} 7^{x_2} \dots$ והמחרוזת שאומרת התפקוד.

- Snap מחשב את מס' ההוראה הבאה לביצוע ודוחף במשבון גם תקצובת.

- הפונ' מקפיאה את התוכנית ברגע t ומחזירה את הערך כל המשתנים של התוכנית.

* **הגדרת הפונק' W_n :**

$W_n =$ קבוצת הקלטים שעליהם עוצרת התוכנית n

חשיביות - קבוצות במ"ע (נמ"ר)

* תבנית לאלגוריתם שמכריע קבוצה חישובית-כריעה:

האלגוריתם של הקבוצה A מקיים:

$T(x)=1 \Leftarrow x\in A$, $T(x)=0 \Leftarrow x\notin A$

```
           פרדיקט חשיבי
T(x): if [____________] goto B
      GOTO E
[B] Y <- 1
```

* תבנית לאלגוריתם המקבל קבוצה במ"ע:

האלגוריתם P מוכיח שקבוצה היא במ"ע אם

שמקבל אותה ומקיים: $P(x)\downarrow \Leftrightarrow x\in A$

```
               פרדיקט חשיבי
P(x): [A] if [____________] goto E
          Z <- Z+1
          GOTO A
```

- אנחנו לא יודעים אם התכנית תעצור עבור x שלא ב-A או לא, אנחנו כן יודעים שאם היא שייכת אז P ידע לעצור עליה.
- אפיונים של קבוצות במ"ע (נמ"ר)

1. יש קבוצה B היא במ"ע אם קיים אלגו' T המקיים $T(x)\downarrow \Leftrightarrow x\in B$
אלגוריתם זה מקבל את איברי הקבוצה B (עוצר על האיברים בקבוצה ורק עליהם)
2. יש קבוצה B היא במ"ע אם קיימת פונ' חשיבית חלקית f, כך שB היא תחום ההגדרה של f
$B=\{x \mid f(x)\downarrow\}$, ניסוח אחר: $Dom(f)=B$ (תחום ההגדרה של הפונ' f)
הסבר: האלגוריתם T עוצר רק על איברי B ומחשב פונקציה ח"ח שמוגדרת רק על איברי B.
3. יש קבוצה B היא במ"ע אם קיים n כך ש: $B=W_n$
(n זה קידוד התוכנית שמקבלת אותה. קבוצת הקודים שקודם עוצרת אלו בדיוק איברי הקבוצה).
4. יש B היא טווח של פונ' רקורסיבית $\Rightarrow$ B נמ"ר
- משפט: הטווח של פונק' רקורסיבית הוא קבוצה במ"ע.
5. יש B היא טווח של פונק' ח"ח f $\Rightarrow$ אז B היא במ"ע.
- משפט: הטווח של פונק' רקורסיבית חלקית היא קבוצה במ"ע.

* B נמ"ר (במ"ע) $\Leftrightarrow$ היא תחום הגדרה

כלומר, B במ"ע אם קיימת תכנית P המקבלת אותה.

6. במ"ע $\Rightarrow$ טווח של פונק' רקורסיבית
- משפט: כל קבוצה במ"ע לא ריקה היא טווח של פונק' רקורסיבית.
- מסקנה: כל קבוצה במ"ע (כולל ϕ) היא טווח של פונק' רקורסיבית חלקית.

* המסקנה המלאה עבור קבוצה B לא ריקה שקולים: * B במ"ע (היא תחום של פונק' רקורסיבית חלקית)

* B היא טווח של פונק' רקורסיבית , * B היא טווח של פונק' רקורסיבית חלקית.

חישוביות - ב"מ

- תנאי חשוב של אלגוריתם ב"מ חייב לכלול:
1. את שורת הנתונים
2. את האלגוריתם
3. את הוכחת הנכונות או כוונה של שורת הנתונים
(הכוונה שהאלגוריתם אכן עוצר על איברי הקבוצה ורק עליהם).

* קבוצה ב"מ ניתנת לייצוג כך: $\{x \mid \exists t R(x,t)\}$ וש $R(x,t)$ הוא פרדיקט חישובי כלשהו.
קבוצה ב"מ זו קבוצת x-ים כך שקיים t שיחד איתם מתקיים ועצמו פרדיקט חישובי.
- אלגוריתם ב"מ מוצא את ה-t הראשון שמקיים את הפרדיקט.

* קבוצות שונות נתונות לפעמים עם הגדרת התוכניות לפעמים עם הגדרת הפונקציה אותה הן מחשבות:

- כל הקבוצות A הללו הן אותה קבוצה בדיוק! — ב"מ לא חישובית

* $A = \{$ תוכניות שעוצרות על 3 $\}$
* $A = \{x \mid \Phi(3,x)\downarrow\}$
* $A = \{\exists t \; stp(3,x,t)\}$
* $A = \{x \mid 3 \in W_x\}$
* $A = \{x \mid$ תוכניות אשר מספר 3 נמצא בתחום הפונקציה שאותה הן מחשבות $\}$
* $A = \{x \mid 3 \in Dom(\Phi_x)\}$ → 3 נמצא בתחום ההגדרה של הפונקציה אותה משבת
* $A = \{x \mid \psi_x^{(3)}\downarrow\}$ ← הפונקציה אותה משבת התוכנית x מוגדרת על 3

- תוכנית עוצרת על 3
⇕
הפונקציה שאותה משבת מוגדרת על 3
- כל הקלטים שעליהם התוכנית עוצרת הם הקלטים שעליהם הפונקציה אותה משבת מוגדרת.
- ↓ המוצמד לתוכנית = עצירה
↓ המוצמד לפונקציה = הגדרה.

* הקבוצה $\overline{Empty}$ (ב"מ לא חישובית) ניתנת להגדרה כך:

* $\overline{Empty} = \{n \mid \exists x_0 \, \Phi(x_0, n)\downarrow\} \equiv \{n \mid \exists x_0 \in Dom(\Phi_n)\}$

* הגדרות שקולות לקבוצה Empty

* $Empty = \{n \mid \forall x \, \Phi(x,n)\uparrow\}$
* $Empty = \{n \mid W_n = \phi\}$
* $Empty = \{n \mid |W_n| = 0\}$
* $Empty = \{Dom(\Phi_n) = \phi \wedge Range(\Phi_n) = \phi\}$
* $Empty = \{n \mid \psi_n^{(x)} = h(x) = \uparrow\}$

חישוביות - רדוקציות

הגדרה: נאמר שקבוצה A ניתנת לרדוקציה לקבוצה B, אם קיימת פונק' רקורסיבית (שלמה ניתנת לחישוב) $f: \mathbb{N} \to \mathbb{N}$ כך שלכל מס' טבעי n:

$$n \in A \Leftrightarrow f(n) \in B$$

תזכורת:

- K (תוכניות שעוצרות על עצמן) - במ"ר לא חישובית
- $\bar{K}$ (תוכניות שלא עוצרות על עצמן) - לא במ"ר
- TOT (תוכניות שעוצרות לכל קלט) - לא במ"ר
- $\overline{TOT}$ (תוכניות שקיים קלט עליו הן לא עוצרות) - לא במ"ר
- Empty - (תוכניות שלא עוצרות לשום קלט) - לא במ"ר
- $\overline{Empty}$ - (תוכניות שקיים קלט עליו הן עוצרות) - במ"ר לא חישובית.
- FIN (תוכניות שעוצרות על מס' סופי של קלטים) - לא במ"ר.
- INF (תוכניות שעוצרות על מס' אינסופי של קלטים) - לא במ"ר.

* עקרון גריסת הקלט - עקרון המסייע לבטל את התלות בקלט, להפוך את כל קלט' הx למס' אחיד m ובכך לשמור את כל הקלטים על m כדי להשאיר את ההתנהגות של כל הקלטים הללו להתנהגות של אותו m.

כשאנו תוצר את הסימון $\Phi(m,m)$ המשמעות שבתחילה תרצה $\Phi(x,x)$ בשלב ראשון מקדים בתוכנה השורה $m \Leftarrow x$ שמשמעה: הקלט x נדרס ע"י הקלט m. כלומר כל קלט' הx הפכו להיות המס' m וכעת מריצה התוכנית m על הקלט m מאין יותר תלות בx הקלט. ומה שיקרה כעת תלוי רק בm: (ואפשר להוסיף הגדלות לתנאים מסקנות).

ואם $\Phi(m,m)\downarrow$ אזי לכל קלט שיגיע התוכנה עוצרת וכן להפך אם $\Phi(m,m)\uparrow$ אז לא תמיד עוצרת.

1. אם $A \le B$ וB רקורסיבית/חישובית/כריעה אז גם A רקורסיבית/חישובית/כריעה.

מסקנה: אם $A \le B$ וA לא חישובית אז גם B לא חישובית.

2. אם $A \le B$ וB במ"ר אז גם A במ"ר

מסקנה: אם $A \le B$ וA לא במ"ר אז גם B לא במ"ר.

3. אם $A \le B$ אז גם $\bar{A} \le \bar{B}$

- הרדוקציה $A \le B$ פירושה שקיימת פונק' f חישובית המקיימת: $x \in A \Leftrightarrow f(x) \in B$
  1. $x \in A \Rightarrow f(x) \in B$
  2. $x \notin A \Rightarrow f(x) \notin B$

- אם "חדשה" $\le_m$ "מוכרת" - אז כל מה שהקבוצה המוכרת לא, ישליך גם על הקבוצה החדשה.
- אם "מוכרת" $\le_m$ "חדשה" - אזי כל "הכן" של הקבוצה המוכרת ישליך על החדשה (קבוצה חישובית נחשבת גם "כן").

* קבוצה תקרא m שלמה אם: 1) B היא במ"ר, 2) לכל קבוצה במ"ר C, מתקיים: $C \le_m B$

חישוביות - רדוקציות

* נכתוב לנו ארכיון של רדוקציות שעשינו ופונקציות הרדוקציה שהשתמשנו:

- $K \le TOT$
- $\bar{K} \le \overline{TOT}$
- $\bar{K} \le Empty$
- $K \le \overline{Empty}$
- $\bar{K} \le FIN$
- $K \le INF$
- $\bar{K} \le R$

פו' הרדוקציה:

$f(m) = \#\{z \leftarrow \Phi(m,m)\}$

פו' הרדוקציה חישובית בגלל משפט הפרמטר וכן בגלל היא מוגדרת תכניות שרצוי פעולה רק.

- $\bar{K} \le TOT$
- $K \le \overline{TOT}$
- $\bar{K} \le INF$
- $K \le FIN$
- $\bar{K} \le A$
- $K \le \bar{A}$

פו' הרדוקציה:

$f(m) = \#\{[A]\ \text{if } stp(m,m,y)\ \text{GoTo } A\}$

- A קבוצת קידוד תכניות אשר עוצרות עם אינפוט של קלט

- R - קבוצת קידוד התכניות בשפת S אשר עם קבלת החישוב שלהן לכל קלט הוא לפחות x^3

* תיאור A: $A = \{n = \langle a, \langle b, c \rangle\rangle \mid$ התכניות הרצות על הקלט b מעוגלת המשתנים, הפלט c בשלב מסוים של התכנית$\}$

$f(m) = \langle p, \langle 1, q \rangle\rangle$ $\quad p = \#\{z_1 \leftarrow \Phi(m,m) \;\; \vdots \;\; Y \leftarrow 3\}$ (איפוס משתני התכנית) $\quad q = [3,0,0,0\ldots]$

- $K \le A$
- $\bar{K} \le \bar{A}$

* קבוצה לא ב"מ

$B = \{$ q התוכניות שעוצרות על הקלט $\}$

$f(m) = \#\{\text{if } x=q\ \text{Goto } E;\ z \leftarrow \Phi(m,m)\}$

- $\bar{K} \le B$
- $K \le \bar{B}$

* קבוצה לא ב"מ

$C = \{$ קבוצת התכניות המחשבות פו' השונה מהפו' $f(x)=x^2$ עבור לכל היותר מס' סופי של קלטים $\}$

$g(m) = \#\{[A]\ \text{if } stp(m,m,X)\ \text{Goto } A;\ Y \leftarrow X^2\}$

- $\bar{K} \le C$
- $K \le \bar{C}$

*הערכות: 0 קלטים הוא מס' סופי!

* קבוצה לא ב"מ

$D = \{$ קבוצת התכניות שעוצרות על כל הקלטים הזוגיים ולא עוצרות על האי זוגיים $\}$

$f(m) = \#\{[A]\ \text{if } \overline{EVEN(x)}\ \text{GOTO } A;\ z \leftarrow \lfloor \frac{x}{2} \rfloor;\ z_2 \leftarrow \Phi(z_1, m)\}$

- $TOT \le D$
- $\overline{TOT} \le \bar{D}$

* לא ב"מ

$E = \{$ כל התכניות שמחזירות x^2 והפלט הראשון של התכנית גדול מ... לא עוצרות x $\}$

$f(m) = \#\{z \leftarrow \Phi(m,m);\ Y \leftarrow x+1\}$

- $\bar{K} \le E$
- $K \le \bar{E}$

* לא ב"מ

$F = \{\langle n, m \rangle \mid 0 < |W_n| < |W_m|\}$

$n = \#\{\text{if } x=0\ \text{GoTo } E;\ z \leftarrow \Phi(p,p)\}$ $\quad m = \#\{Y \leftarrow 1\}$ $\quad f(p) = \langle n, m \rangle$

- $\bar{K} \le F$
- $K \le \bar{F}$

כיוון מכאן שבו התכנית m תעצור בכל מקרה על אינסוף קלטים ולכן $|W_m| = \aleph_0$
ודאי התוכנית n תעצור גם

חישוביות - רדוקציות

* המשך ורכין הרדוקציות שהשתמשנו בהם:

* כמו להוכיח שהקבוצה הבאה אינה כך שהתוכנית a מ-b בריצה על הקלט c מסתיימת בתוצאה $A=\{\langle a,\langle b,c\rangle\rangle \mid c\}$

$f(m)=\#\langle 200 < p, lt(p-1)\rangle$ $\quad p=\#\begin{cases} z \leftarrow \Phi(m,m) \\ y \leftarrow 1 \end{cases}$

$* K \le A$

* קבוצת התוכניות המחשבות פונקציה קבועה - לא במ"מ

$CONST=\{n,m \in N \mid \forall x\ \Phi(x,n)=m\}$

$f(m)=\#\begin{cases} z \leftarrow \Phi(x,m) \\ y \leftarrow 3 \end{cases}$ $\quad * TOT \le CONST$

! נשתמש ברדוקציה מ-TOT בכל פעם שנרצה קבוצה של תוכניות המחשבות פונקציה שלמה

כדי שתהיה מוגדרת לכל x, ואז פשוט נשווה בין ערך מוחזר מ"ש ומחזיר.

רדוקציה מ-TOT אפשרי $* CONST \le TOT$

$f(m)=\#\begin{cases} z_1 \leftarrow \Phi(x,m) \\ z_2 \leftarrow \Phi(7,m) \\ [A]\ \text{if } z_1 \ne z_2 \text{ GoTo } A \end{cases}$

* תהי A קבוצת כל הזוגות $\langle n,m\rangle$ כך ש-n,m הן תוכניות ולכל x מתקיים:

$\Psi_n(x) \ne \Psi_m(x)$ (הערך המוחזר מהתוכנית n על הקלט אינו שווה לערך המוחזר מהתוכנית m על הקלט הזה)

$f(m)=\langle m,n\rangle$ $\quad m=\#\{z \leftarrow \Phi(x,p)$ $\quad n=\#\{[A]\ \text{goto } A$ לא עוצרת לעולם

התוכנית m תלויה בp $\quad * \overline{TOT} \le \bar{A}$ $\quad * TOT \le A$

* תהי B קבוצת כל הזוגות $\langle m,n\rangle$ כך ש-$|\bar{W}_m| < |\bar{W}_n|$, B אינה במ"מ

$f(\ ) = \langle n,m\rangle$ $\quad * \bar{K} \le B$

התוכנית n לא עוצרת רק על 0 $\quad |\bar{W}_n|=1$ ולכן $\quad n=\#\{[A]\ \text{if } x=0 \text{ GoTo } A$ $\quad m=\#\{z \leftarrow (p,p)$ תלויה בp

* תהי C קבוצת כל הזוגות $\langle m,n\rangle$ כך ש-$W_m = W_n$ (n עוצרת על אותם הקלטים שעליהם m לא עוצרת)

$f(p)=\langle p,q\rangle$ $\quad q=\#\{[A]\ \text{GoTo } A$ $\quad * Tot \le C$

* תהי A קבוצת קידודי התוכניות אשר עוצרות על מס' אינסופי של קלטים. נוכיח שאינה במ"מ

$f(m)=\#\{[A]\ \text{if } stp(m,m,x) \text{ GoTo } A$ $\quad * \bar{K} \le A$

* תהי B קבוצת קידודי תוכניות שעוצרות רק על מספר ראשוני. נוכיח שאינה במ"מ

$f(m)=\#\begin{cases} \text{if } prime(x) \text{ GoTo } E \\ z \leftarrow \Phi(m,m) \end{cases}$ $\quad * \bar{K} \le B$

$* K \le \bar{B}$

* תהי A קבוצת קידודי הזוגות $\langle n,m\rangle$ כך ש-$1 < |\bar{W}_n| < |\bar{W}_m|$. נוכיח שאינה במ"מ

$f(p)=\langle n,m\rangle$

$n=\#\begin{cases} \text{if } x>3 \text{ GoTo } E \\ [A]\ \text{GoTo } A \end{cases}$ עוצרת על 0,1,2 $\quad m=\#\{z \leftarrow \Phi(p,p)$

$* \bar{K} \le A$ $\quad * K \le \bar{A}$

חישוביות - רדוקציות ורייס

דוג' מתוך מבחן!!! * תהי הקבוצה הבאה $A=\{n\in\mathbb{N} \mid \psi_m^{(1)}(x) = f(x)=x^2\}$ צריך להוכיח כי $TOT\equiv A$ A לא במ"ש

$h(m) = \# \begin{cases} z \leftarrow \Phi(x,m) \\ [A]\ \text{if } z \neq x\cdot x \text{ Goto } A \end{cases}$ $*A\le TOT$

$g(m) = \# \begin{cases} z \leftarrow \Phi(x,m) \\ y \leftarrow x^2 \end{cases}$ $*TOT\le A$

- לזכור בקוד משתנה התנאי כי הוא יכול להמשיך להחזיר שלילות!

* A קבוצת קידודי הזוגות $\langle\langle a,b\rangle,\langle c,d\rangle\rangle$ כך שהתוכנית מס' a בריצה על הקלט b יש הוא מבצעת את שורה c בריצה, מוגש ועד הוא בריצה תבצע את שורה d בריצה שלה.

- כדי להשתמש בקבוצה הזאת צריך לבדוק שהיא מבצעת את שורה c מבצע את d אך לא מבצע D ואז בדיקה אינסופית. - לא ניתן לבדיקה - לא במ"ש! נעשה רדוקציה $*K\le B$

$f(m) = \langle\langle 17, q\rangle, \langle lt(q+1)\dot{-}2, lt(q+1)\rangle\rangle$

$g = \# \begin{cases} z \leftarrow \Phi(m,m) \\ y \leftarrow y & \text{// שורה } c \\ [A]\ \text{Goto } A \\ y \leftarrow y & \text{// שורה } d \end{cases}$

יש התוכנית המקבלת קלט הסבר קידוד שלה אם לא עוצר
התוכנית תגיע לשורה c לעולם, ⇐ חוקי ולכן שורש לא
ויש התוכנית כן עוצרת על קלט הסבר קידוד שלה
אז הוא תבצע את שורה c ותבצע בלולאה אינסופית לא שורה d.

* נבחין בין המהלכים של הקבוצות הבאות ונוכיח:

- A = קבוצת התוכניות שלכל קלט x התוכנית מבצעת עד x^3 צעדים -
הקבוצה הזאת לא במ"ש כי מדובר בבדיקה לכל x - בדיקה אינסופית.
- B = קבוצת התוכניות שקיים קלט x שעבורו התוכנית מבצעת עד x^3 צעדים:
ניתן לבדוק כמות לקלט את התוכנית בשלבים ואלגוריתם קבלה המשלים הפרמטר
חישובי קלט, הבדיקה אי-סופית לא אפשרי - במ"ש לא חישובית:
- C = קבוצת התוכניות שעבור הקלט 5 מבצעות עד 5^3 צעדים -
מכיון שמדובר בקלט ספציפי, ניתן לכתוב אלגוריתם קבלה ולכן הקבוצה חישובית.

! * חשוב מאוד לזכור!

- במשפט של רייס כאשר מתקבלת על הקבוצה N חשוב לזכור!
N היא קבוצת אינדקסים! הבעיה המתאימה לה: {כל הפונ' החשיבות חלקיות} = F
ו-N מכילה את R_F כאשר F מכילה את כל הפונ' הח"ח ⇐ N קבוצת אינדקסים מאוד שגרתית.
אבל לא ניתן להוכיח אי חישוביות ע"פ רייס מכיון שזו כן מלאה!

- במשפט של רייס כאשר מתקבלת על קבוצה ריקה ϕ חשוב לזכור! *
ϕ מכילה את R_F כאשר F ריקה!
כלומר, ϕ היא R_F קבוצת אינדקסים כאשר F ריקה.
אבל אי אפשר להשתמש ברייס מכיון שזו ריקה!
כלומר היא כן קבוצת אינדקסים אבל לא ניתן להוכיח שאינה ריקה!

חישוביות - ר"ס

- לפני משפט ר"ס נגדיר את קבוצת אינדקסים:

קבוצת אינדקסים - קבוצת מספרים טבעיים אשר מקודדים תוכניות המחשבות פונקציה ח"ח זהה. כלומר, כל תוכנית השייכת לקבוצה תחשב בדיוק את אותה הפונקציה שמחשבות התוכניות בקבוצה. (אין פה הגבלות בכמות תוכ', יש פונק' יש יותר מפי אחת).

- ובהתאמה יהיו לנו תמיד שתי קבוצות:

Γ ו-R_Γ כאשר Γ היא קבוצה של פונקציות ח"ח (מתאימות מוש)

ובהתאמה R_Γ היא קבוצת אינדקסים - קבוצת קידודי תוכניות המחשבות כל אחת את הפונקציות/פונקציה שנמצאות בקבוצת הפונקציות Γ.

לא להתבלבל: Γ היא קבוצת פונקציות, R_Γ היא קבוצת תוכניות!

- יש קבוצת אינדקסים אינה ריקה, היא התכונה אינסופית. (קיימות אינסוף תוכניות המחשבות פו' זהה: ואפשר להוסיף לכל תוכנית המחשבת פו' להוסיף לה שורות שלא משפיעות - קוד מת ושגוי).

המשפט:

יש R_Γ היא קבוצת אינדקסים, ובהתאמה Γ המתאימה לה היא לא ריקה ולא מלאה (יש לפחות פו' אחת שנמצאת ב-Γ ולפחות פו' אחת שלא נמצאת ב-Γ) $\Rightarrow R_\Gamma$ איננה חשיבה.

הערה: במידה וכן המתאימה מוכלת הפו' המוגדרת $\uparrow$ = Λx וכו' R_Γ המתאימה היא לא כמו".

- משפט ר"ס הוא כלי המוכיח כי קבוצת תוכניות כלשהי איננה חישובית.

* דוגמה לקבוצה שאינה קבוצת אינדקסים:

תוכניות שמחזירות 1 מיני מחלק של המספר שלהן. נקרא לקבוצה A.

לא קבוצת אינדקסים מכיון שמחשבות הן לא פונקציונליות. נראה דוגמה לב' תוכניות המחשבות פו' זהה כך שאחת השתייך לקבוצה והשנייה לא.

$$\Psi_q^{(1)}(x) = \Psi_p^{(1)}(x) = f(x) = 1$$

$q \in A$ ואילו $p \notin A$ לכן הקבוצה A לא קבוצת אינדקסים.

$$p = \# \begin{cases} Z \leftarrow Z + 1 \\ Y \leftarrow Y + 1 \end{cases} \qquad q = \# \begin{cases} Y \leftarrow Y \\ Y \leftarrow Y + 1 \end{cases}$$

* דוגמה לכך שקבוצה מלאה תמיד. תמיד כשמדובר קבוצה המכסה את $\mathbb{N}$ $\mathbb{N}$ היא קבוצת אינדקסים. $\{$כל הפו' הח"ח$\} = \Gamma$, $\mathbb{N}$ מכיל R_Γ כאשר Γ מכילה את כל הפו' הח"ח. ואילו $\mathbb{N}$ כן משמרת שקילות ולכן היא מלאה ולכן לא ניתן להשתמש בר"ס כי לא ניתן להראות פונקציה שלא נמצאת בכל המחשבות אותה. כלומר שאיננה מלאה.

* דוגמה לכך שקבוצה ריקה תמיד. הקבוצה הריקה ϕ היא קבוצת אינדקסים ϕ מהווה קבוצת אינדקסים R_Γ כאשר Γ ריקה! היא כן משמרת שקילות ולכן לא ניתן להשתמש בר"ס כי Γ ריקה ולא נוכל להראות שאיננה ריקה.

בס"ד

חישוביות - מכונת טיורינג

מכונת טיורינג = תוכנית בשפת C = מחשב = אלגוריתם

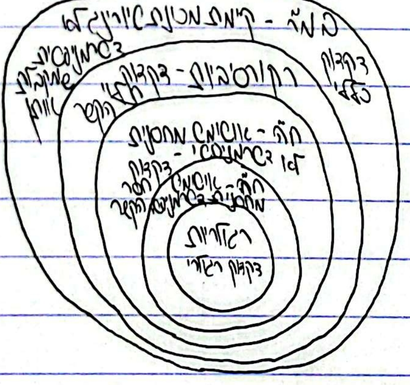
- מכונת טיורינג כמ"פ מחשבת פונקציה.
- במקום קבוצה -> שפה, במקום אלגוריתם -> מכונה.

* למכונת טיורינג יש מס' סופי של מצבים.
* יש סרט אינסופי שעליו המכונה קוראת ורק עליו היא יכולה לכתוב ולקרוא.
* מכונת טיורינג עוצרת כאשר נמצאת במצב q_i וקוראת את הסמל s_j ואין במכונה הוראה שמתחילה בזוג $s_j q_i$ $q_i s_j$
* הפלט הוא מה שכתוב על הסרט בסיום ריצת המכונה.
* מכונה שמכריעה שפה - עוצרת ומחזירה 1 על כל מילה שבשפה, עוצרת ומחזירה 0 על כל מילה שלא בשפה.
* מכונה שמקבלת שפה - עוצרת על המילים שבשפה ורק עליהן! (ואולי עוצרת על מילים שלא בשפה)

! אם קיימת מכונה מכריעה, אז קיימת מכונה מקבלת.

* במקרה של בניית מכונה שמכריעה שפה, אפשר גם להפסיק 0 או 1. גם להגדיר 2 מצבים שאחד הוא קבלה (accept) והשני הוא דחייה (reject).
* כל המילים שעליהן המכונה עוצרת = (המילים) שבשפה שאותה המכונה מקבלת = השפה המתקבלת ע"י המכונה.

ההיררכיה:



* בעיית העצירה - אין אלגוריתם שבהינתן לו מכונת טיורינג M ומילת קלט w, יכריע האם w מתקבלת ע"י המכונה M.
* מכונות לא דטרמיניסטיות נועדו רק לקבלת שפות ולא לחישוב פונקציות - השפה שהמכונה מקבלת הוא קבוצת המילים שיש להן חישוב מקבל. (חישוב שבו המכונה עוצרת לבסוף)

- מכונה לחישוב פונקציות תהיה לנו תמיד ע"י דטרמיניסטית.

◦ אם המכונה דטרמיניסטית אז קבלת מילה = עצירה.
◦ אם המכונה לא דטרמיניסטית אז קבלת מילה = קיום של מסלול אחד לפחות שבו המכונה עוצרת בעבור המילה הנתונה.

* שפה כמ"ה ⟺ קיימת מכונת טיורינג א"ד דטרמיניסטית שמקבלת את השפה.

טיורינג - המשך

* משפט: לכל שפה במ"ז B יש מכונת טיורינג של אנומרציה שמקבלת את B.
* משפט: כל שפה שמתקבלת ע"י מכונת טיורינג של אנומרטורית היא שפה במ"ז.

- המכונת טיורינג ע"י אנומרטור לא מוסיף כוח.
- יכולים לתת לה מכונה המחשבת את השאלות כאשר שהמ"ט עוצרת לכל מחרוזת קלט כולל הריקה, עוצרת ומדפיסה מחרוזת פלט.

ויכולים לתת לה מכונה המכריעה שפה והיא מתפקדת בדיוק כמו מ"ט מכריע. עוצרת ומדפיסה 1 עבור כל מחרוזת שבשפה, ועוצרת ומדפיסה 0 עבור כל מחרוזת שלא בשפה.

* מכונה שעוצרת לכל קלט, מחשבת פו' שמוגדרת לכל קלט.
* מכונה שמחשבת פו' חח"ע ועל - מילה ועוד לפחות שאינו מציג הריקה - אם ריק לא עוצרת, המספור אינסופי וחוזרת בחזקת המספרי של הסדר.

מכונה שקיים קלט שבו לא עוצרת, מחשבת פו' שקיים לפחות קלט אחד שעליו לא מוגדרת.

* מכונה שמכריעה שפה - עוצרת ומדפיסה 1 על מילה בשפה ועוצרת ומדפיסה 0 על מילה שלא בשפה.

חישוביות - מחלקות סיבוכיות P, NP

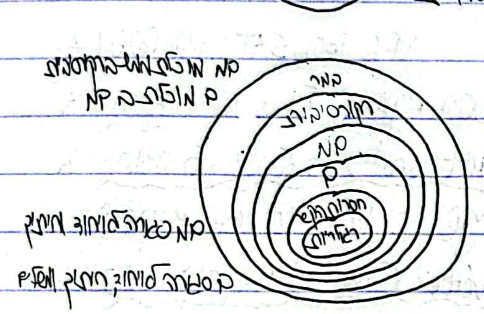
* הגדרה: נאמר כי שפה L מעל אלפבית נתון Σ שפתי ניתנת להכרעה בזמן פולינומיאלי, אם:
1. קיימת מכונת טיורינג M (דטרמיניסטית) שמקבלת את L
2. קיים פולינום $p(x)$ כך שמס' צעדי החישוב של M על קלט w ששייך ל-L אינו עולה על $p(|w|)$

במילים פשוטות: נאמר כי שפה שפתי ניתנת להכרעה בזמן פולינומיאלי אם קיימת מכונה דטרמיניסטית שמכריעה כל מילה שבשפה ורק אותם, וגם מס' צעדי החישוב על מילה מסוימת חסום בפולינום התלוי באורך מילת הקלט.

* קבלה פולינומיאלית ⇐ הכרעה פולינומיאלית.
* המחלקה P - מחלקת השפות הניתנות להכרעה בזמן פולינומיאלי נקראת P, זאת אומרת גם ניתנות להכרעה בזמן פולינומיאלי.

חישוביות P

- שימו לב: המחלקה P מוכלת ממש במחלקת השפות הרקורסיביות.
- קל להוכיח ששפה נתונה שייכת ל-P, אבל קשה להוכיח ששפה נתונה L אינה שייכת ל-P.



* SAT - שפת הפסוקים הספיקים בצורת CNF (השמה מספקת=T)
- אלגוריתם להכרעת שייכות ל-SAT הוא אקספוננציאלי $2^{\text{מס' משתנים}}$
* הגדרה: נאמר כי שפה L מעל אלפבית נתון Σ שפתי ניתנת לקבלה בזמן פולינומיאלי לא דטרמיניסטי, אם:
1. קיימת מכונת טיורינג לא דטרמיניסטית שמקבלת את L
2. קיים פולינום $p(x)$ כך שלכל קלט w ששייך ל-L יש חישוב מקבל (עץ חישוב) שמס' הצעדים בו אינו עולה על $p(|w|)$
* הגדרה: מחלקת השפות הניתנות לקבלה בזמן פולינומיאלי לא דטרמיניסטי נקראת NP.

במילים פשוטות: קיימת מכונת טיורינג א"ד שמקבלת את השפה בזמן החסום בפולינום התלוי באורך הקלט.

* משפט $SAT \in NP$

הוכחה: נבחר השמה ונבנה מכונת טיורינג שמקבלת את SAT בזמן פולינומיאלי שפועלת כך.
1. עבור קלט $x \in \Sigma^*$ המכונה בודקת אם x מייצג נוסחה בצורת CNF, הבדיקה דורשת $O(|x|)$ צעדים. אם התשובה שלילית המכונה נעצרת במצב דוחה. (דוגמה ל-SAT: $(A\vee B)\wedge(\bar{A}\vee C)\wedge(\bar{B}\vee \bar{C})$)
2. הבדיקה פועלת בשלב השני, המכונה נכנסת לשלב ניחוש, זה השלב הלא דטרמיניסטי.

יש לנחש השמה של פסוקית היא 1 או 0, המכונה סורקת את הנוסחה ומכל פסוקית שמופיע בה p_j היא מוחקת את הפסוקית (כי היא כבר סופקה), ומכל פסוקית שמופיע בה $\sim p_j$ אם יש לה משתנים בפסוקית המכונה מוחקת רק את $\sim p_j$, ואם אין פסוקים אחרים אז [illegible] שכל הנוסחה מקבלת ערך 0 והמכונה נעצרת במצב דוחה.

אם ערך השמה של p_j הוא 0, המכונה תבצע את הפעולה ההפוכה. וכך ממשיך, אם הנוסחה נמחקה - השמה T.

חישוביות - P, NP ורדוקציות

* ידוע ש NP מוכלת ממש ברקורסיביות, ידוע ש P מוכלת ב NP.
* המחלקה P סגורה לאיחוד, חיתוך ומשלים
* המחלקה NP סגורה לאיחוד וחיתוך
* רדוקציה פולינומיאלית - נאמר ששפה L ניתנת לרדוקציה פולינומיאלית לשפה Q אם קיימת פו' f רקורסיבית וניתנת לחישוב בזמן פולינומיאלי, כך שלכל מילה w מעל האלפבית של L מתקיים $f(w)\in Q \Leftrightarrow w\in L$, סימון: $L\le_p Q$
* משפט: $L\le_p Q$ ו $Q\in P$ אז L שייכת ל P.
* משפט: $L\le_p Q$ ו $Q\in NP$ אז L שייכת ל NP.
* שפות NPC - שפה L נקראת NP שלמה אם: 1) L שייכת ל NP 2) לכל שפה $Q\in NP$ מתקיים $Q\le_p L$

* משפט קוק: SAT היא NPC
* 3-SAT - שפת כל הנוסחאות הספיקות בצורת CNF, שבהן בכל פסוקית יש 3 ליטרלים.
* קבוצה בלתי תלויה (Independent set) - בגרף לא מכוון $G=(V,E)$ היא קבוצת קודקודים שאין אף צלע בין שניים מהם.
* כיסוי קודקודים (Vertex Cover) - קבוצת הקודקודים בגרף לא מכוון $G=(V,E)$ נקראת כיסוי קודקודים, אם לכל צלע (v,w) בגרף, v או w (או שניהם) שייכים לU
* תת גרף מלא (Clique) - קבוצת קודקודים U בגרף לא מכוון $G=(V,E)$ נקראת קליקה אם יש צלע בין כל שני קודקודים מקבוצה.
* כיסוי קבוצות (Set Cover) - בהינתן משפחה של קבוצות של תתי איברים $\{S_1,S_2,\dots,S_m\}$ ומספר k. השאלה היא האם יש k קבוצות במשפחה שהאיחוד שלהן שווה לאיחוד כל הקבוצות שבמשפחה.
* בעיות נוספות: מעגל המילטון, תת גרף איזומורפי, בעיית הסוכן הנוסע, בעיית החלוקה.

חישוביות - חשוב לזכור!

* הפונק' המחושבת ע"י התוכנית הריקה, מהו פרדיקט האופיין בעבור הקבוצה הריקה.
* קבוצה ללא איברים => חישובית-כריעה, הפרדיקט האופיין $f(x)=0$
* תוכנית ס המחושבת את הפונק' הריקה $h(x)$ (זו הפונק' המוגדרת $h(x)=\uparrow$) לחשבת כמקבלת את הקבוצה הריקה - כן, זו התוכנית: $P = \#\{[A]\ GOTO\ A\}$
* יש פרדיקט האופיין של הקבוצה הוא ר"ח, אז הקבוצה המוכרעת באמצעותו הוא ר"ח.
* קבוצה הוא חישובית-כריעה-רקורסיבית אם"ם הפרדיקט האופיין שלה הוא פונק' חישובית רקורסיבית.
* אוסף הכוונים של פונק' רקורסיביות הוא במ"ן לא חישובית.
* תחום ההגדרה של פונק' רקורסיבית הוא קבוצה חישובית כריעה. (פונק' חישובית = ע"מ = תחום הוא N = חישובי).
* תוכנית שעוצרת לכל קלט, לא עוצרת בעבור קבוצה סופית של קלטים.
  הסבר: תוכנית שעוצרת לכל קלט, לא עוצרת בעבור 0 קלטים וזו קבוצה סופית.
* קבוצת התוכניות שקידודן זוגי זו קבוצה חישובית כריעה וגם במ"ן (נובע מההגדרה), קבוצה אינסופית, הפרדיקט האופיין שלה הוא $(x|2)$, מכילה תוכניות שהקוד שלהן הוא אי זוגי, וכן מכילה תוכניות שהפקודה הראשונה שלהן הוא [illegible]. קבוצה זו היא הקבוצת המספרים הזוגיים.
  A קבוצת התוכניות שקידודן זוגי. זו הקבוצה Even - קבוצת המספרים הזוגיים. $(x|2)$.
* פונק' רקורסיבית זו פונק' שקיימת תוכנית בשפת S המחשבת אותה והיא עוצרת ומחזירה ערך בעבור כל קלט.
* $A=\{x \mid \forall z \text{ not } stp(x,x,z)\}$ זו הקבוצה $\bar{K}$. (קבוצת כל הקלטים שלא עוצרים בעבור קלט השווה לקידוד).
* הקבוצה N זו קבוצה שהפרדיקט האופיין שלה הוא פונק' קבועה (הפרדיקט הוא $f(x)=1$).
* הפונק' המחושבת ע"י תוכנית שעוצרת לכל קלט (TOT) הוא פונק' חישובית, וקבוצת התוכניות שעוצרות לכל קלט (TOT) הוא קבוצה לא במ"ן.
* יש פרדיקט האופיין של הקבוצה הוא ר"ח, אז הקבוצה המוכרעת באמצעותו הוא ר"ח.
* תחום ההגדרה של פונק' רקורסיבית הוא קבוצה במ"ן חישובית. (תחום ההגדרה הוא N)
* תוכנית שעוצרת לכל קלט, לא עוצרת בעבור קבוצה סופית של קלטים (לא עוצרת בעבור 0 קלטים).
* תהי A קבוצת התוכניות שקידודן אי זוגי, הקבוצה A לא מכילה תוכניות שמתחילות בהוראה הדמה $y \leftarrow y$, מכילה תוכניות שהקוד שלהן הוא זוגי, כי המס' הזוגיים שייכים לקבוצה, הקבוצה אינסופית, הפרדיקט האופיין של הקבוצה הוא $\sim(x|2)$, הקבוצה חישובית-כריעה.

חשוב לזכור!

* K איננה משלימה סגורה - K היא קבוצת תוכניות שעוצרות על קלט המהווה הקידוד שלהן
והקידוד שלה כזה שהתוכן משנה לו סמנטיקה, אולי נגיד ישירות מתאימים מספקים.
נבנה 2 תוכניות שקולות שאחת תשתייך לK והשניה לא.

$$p = \# \begin{cases} Y \leftarrow Y \\ [A]\ IF \sim(2|X)\ GOTO\ A \end{cases} \qquad q = \# \begin{cases} Z \leftarrow Z \\ [A]\ IF \sim(2|X)\ GOTO\ A \end{cases}$$

שתי התוכניות מחשבות פו' זהה - זוגיות EVEN, אולו תוכניות המחשבות זוגיות EVEN
והלא p שייכת לקבוצה וq לא שייכת ⇐ K לא משלימה סגורה ⇐ אינו ספציפי.

* תוכניות המחשבות את הפו' הקבועה $f(x)=7$ — לא בע"מ $A = \{n \mid \forall x\ \Phi(x,n) = 7\}$
הקבוצה לא בע"מ (תנאי שלילי, בדיקה לכל x, בדיקה אינסופית).

* תוכניות שעוצרות על קלט המהווה הקידוד שלהן ומחשבות את $f(x)=7$ בע"מ לא רקורסיבית $B = \{n \mid \Phi(n,n) = 7\}$
הגדרה שקולה! $B = \{n \mid \exists z_0\ stp(n,n,z_0) \wedge (r(snap(n,n,z_0)))_1 = 7\}$

* משלים התוכניות שעוצרות על עצמן לא מחזירות 7 - יכולות גם לא לעצור. לא בע"מ $C = \{n \mid \Phi(n,n) \neq 7\}$
משלים הבדיקה אינסופית, צריך לבדוק שהתוכניות עוצרות לכל קלט — לא בע"מ $D = \{C \cap TOT\}$

* תוכניות שעוצרות לכל x. לא בע"מ $TOT = \{n \mid \forall x\ \Phi(x,n)\downarrow\}$

$A = \{n = \langle a, \langle b, c \rangle \rangle \mid$ קבוצת כל השלשות שאומרות כך שהתוכנית a בריצתה על b, מגיעה לשלב שבו הערך של המשתנה הראשון הוא $c\}$
הריצה תמיד מגיעה בשלב מסוים שבו ערך המשתנה יהיה c
תוכניות אלו לא חייבות לעצור, העיקר שבריצה בשלב כלשהו ערך המשתנה יהיה c - בע"מ. לא רקורסיביות
x הוא מס' המקודד שלשה $l(x) = a$, $l(r(x)) = b$, $r(r(x)) = c$

$$T(x) = \begin{cases} [A]\ if\ (r(snap(l(r(x)), l(x), z)))_1 = r(r(x))\ Goto\ E \\ z \leftarrow z+1 \\ Goto\ A \end{cases}$$

* קבוצת התוכניות שקיים קלט (אחד לפחות) שעליו הן עוצרות - לא רקורסיבית. בע"מ $\overline{Empty} = \{n \mid \exists x_0\ \Phi(x_0,n)\downarrow\}$

$$T(x) = \begin{cases} [A]\ if\ stp(l(z), x, r(z))\ Goto\ E \\ z \leftarrow z+1 \\ Goto\ A \end{cases}$$

תוכנית
שבה היא עוצרת z מקודד גם קלט וגם צעד.

* תוכניות שמחשבות פו' 0 או 1 קבועה — לא בע"מ. לא בע"מ $A = \{n \mid |W_n| \leq 1\}$
תוכניות שעוצרות לפחות על 2 קלטים שונים. בע"מ לא רקורסיבית $B = \{n \mid |W_n| > 1\}$

$$T(x) = \begin{cases} [A]\ if\ stp((z)_1, x, (z)_3) \wedge stp((z)_2, x, (z)_3) \wedge z_1 \neq z_2\ Goto\ E \\ z \leftarrow z+1 \\ Goto\ A \end{cases}$$

תוכנית
שעוצרת כאשר
x שייך לקבוצה B

כשל המבנה של stp בודק עבור תוכנית x בצעד ניתן לבדוק כמה צעדים ... לא מבצעת
כך שבכל מקרה נעצור בזמן סופי כדי לבדוק את שניהם.

חישוביות - חשוב לצטט!

* תהי A קבוצת התוכניות שעוצרות על n קלטים עוקבים.
הקבוצה הנ"ל לא חישובית - ניתן להראות אלגוריתם המקבל אותה תוך שימוש במשתנה עזר

$$T(x) = \begin{cases} [A]\ \text{if}\ (stp(l(z),x,r(z)) \wedge stp(l(z)+1,x,r(z)) \wedge stp(l(z)+2,x,r(z)) \wedge stp(l(z)+3,x,r(z)))\ \text{Goto}\ E \\ z \leftarrow z+1 \\ \text{Goto}\ A \end{cases}$$

תוכנית

* $B=\{<x_1,x_2> \mid x_1+x_2 \in K\}$ הנ"ל לא חישובית

$B = \{x \mid \Phi(l(x)+r(x), l(x)+r(x))\downarrow\}$

$$T(x) = \begin{cases} [A]\ \text{if}\ stp(l(x)+r(x), l(x)+r(x), z)\ \text{Goto}\ E \\ z \leftarrow z+1 \\ \text{Goto}\ A \end{cases}$$

מקבלת

* הגדרת ר"ס: F - קבוצת פונקציות ח"ח
R_F - קבוצת התוכניות המחשבות את אחת הפונ' השייכות ל-F (כאשר מתקיים F, R_F מוגדרת
לפי משפט ר"ס: קבוצת האינדקסים שלה!)
* בהינתן F שהיא קבוצת אינדקסים ואם F לא ריקה ולא מלאה ⇐ R_F אינה חישובית לפי ר"ס.

A) הקבוצה Even: הפרדיקט הבוליאני הוא: (הפרדיקט הבוליאני הוא ר"פ) $EVEN=\{x \mid (2|x)\}$ ר"פ

$$g(x) = \begin{cases} \text{IF}\ (2|x)\ \text{Goto}\ B \\ \text{Goto}\ E \\ [B]\ Y \leftarrow 1 \end{cases}$$

התוכנית המחשבת:
$$g(x) = \begin{cases} 1 & \text{if}\ (2|x) \\ 0 & \text{else} \end{cases}$$

תוכנית שמחזירה ע"ג יכולה להיות כל פעם ח"ח!
הקבוצה A לא משמרת שקילות (אינה קבוצת אינדקסים)

B) קבוצת התוכניות המחשבות פונ' שקיימת לה תוכנית שמחשבת אותה ומקודדת ע"י.
$\mathbb{N}=B$ לכל הפונ' כאשר קיימת תוכנית שמקודדת ע"י. לכן B חישובית ואפילו ר"פ
F מלאה לכן אי אפשר להשתמש בר"ס.

C) קבוצת תוכניות המחשבות את הפרדיקט הבוליאני של Even

$C=\{n \mid \forall x\ \Phi(x,n) = g(x)\}$
$$g(x) = \begin{cases} 1 & \text{if}\ (2|x) \\ 0 & \text{else} \end{cases}$$

הקבוצה C היא R_F עבור F המכילה את הפונ' $g(x)$
הקבוצה הנ"ל לא ב"ה כי מדובר בבדיקה לכל x - אינסופית. (תוכנית המחשבת פונ' שלמה)
כדי להוכיח אי חישוביות לפי ר"ס יכולים להשתמש בזה.
$C=R_F$ והיא קבוצת אינדקסים - תכונה המשותפת לקבוצה היא פונקציונליות להגדרה
ואינה תלויה במימוש הספציפי.
∘ F לא ריקה שכן (g שייכת לה), ∘ F לא מלאה שכן ↑ (הפונ' שלא מוגדרת) לא שייכת לה.
לכן הקבוצה C לא חישובית לפי ר"ס.

נספח

חישוביות - חשוב לזכור!

* D = קבוצת התוכניות שלכל קלט מחשבות לפחות m^2 צעדים
הקבוצה היא לא בע״ת, ומדובר בבדיקה לכל קלט - בדיקה אינסופית.
הרעיון: רדוקציה K ל־$\bar{K}$ והיא f פו׳ הרדוקציה $f(m) = \#\{z \leftarrow \Phi(m,m)\}$
(הערה: גם היה מאחר בקלט יחיד זה היה בע״ת וניתן לחשוב באמצעות פונקצית stp כך: $\sim stp(z,x,z^2)$
זו גישה להוכחת אי חישוביות כי ר״ס לא יכולה להיות פה כיון ש-D לא משמרת שקילויות
כי מדובר בתכונה שאינה פונקציונאלית אלא תלויה מימוש.

* E = קבוצת התוכניות המחשבות פונקציות שקיימת לה תוכנית המחשבת אותה בזמן m^2 צעדים לכל x.
$E = N$ היא חישובית ואף רה״ מכיון שמדובר בכל התוכניות החח״, לכל תוכנית חח״ ניתן למצוא מקרה מוגדר לכך שמחשבת ביותר m^2 צעדי חישוב.
$N = R_1 = E$, הקבוצה היא קבוצת אינדקסים אולם היא קבוצה מלאה לכן ניתן להשתמש בר״ס.

* F = תוכניות המחשבות פו׳ שקיימת לה תוכנית המחשבת אותה ב-x^2 צעדים לכל היותר לכל x.
יש אוסף חסום שלכל פו׳ קיימת תוכנית שמחשבת אותה בפחות m^2 צעדים - הרי זה חסם עליון.
צריך לחשוב מה מס׳ הצעדים המינימליים הנחוצים לחישוב התוכנית.
למשל הפו׳ $f(x)=0$ מס׳ הצעדים הנחוצים לחישובה: התוכנית הריקה 3 מחשבת אותה ב-0 צעדים
לכל x, לכן לפו׳ $f(x)=0$ קיימת תוכנית המחשבת אותה ב-x^2 צעדים לכל היותר לכל x (כולל המקרה 0!)
לעומת זאת הפו׳ $f(x)=1$ התוכנית הקצרה ביותר המחשבת אותה היא $1 \leftarrow y$ $p = \#\{y \leftarrow 1\}$
כאשר נחוץ לפחות צעד 1, ומכאן שלפו׳ $f(x)=1$ לא קיימת תוכנית שיכולה לחשב אותה
תוך לכל היותר x^2 צעדים לכל x (כי עבור המקרה 0 לא מספיק לה היותר 0 צעדים אלא מספיק 1!)
ולכן $f(m) \notin r$, לעומת $f(x) \in r$
הקבוצה היא קבוצת אינדקסים והתכונה פונקציונאלית בלבד. קיבלנו: $R_1 = F$
כאשר f היא הפונקציה $0=f(x)$. $F = \{n \mid \forall x\, \Phi(x,n) = 0\}$
קבוצה זו אינה כח״ כיון שמחשבת פו׳ שונה. ומכאן אי חישוביות לפי ר״ס.

* G = 1 היא קבוצת פו׳ חח״ שקיימת לה תוכנית המחשבת אותה, כך שאם נמחק את כל
ה-G השורות שמשתמשות בפקודת goto, נקבל תוכנית המחשבת את הפרדיקט האופייני של
הקבוצה הריקה $\Rightarrow$ 1 מכילה את כל פו׳ החח״.
לכל פו׳ חח״ קיימת תוכנית כזו, כך ש-G מכילה את כל פו׳ החח״ והתוכנית המחשבת אותן
מן ה-G התוכניות $N = R_1$. לכל פו׳ חח״ ניתן לכתוב תוכנית שקולה שמתחילה בצ׳ פקודות goto
ואז אם נמחק את כל השורות האלו שמשתמשות ב-goto נקבל תוכנית ריקה.
והתוכנית הריקה מחשבת את $0=f(x)$ שהוא הפרדיקט האופייני של $\emptyset$.
הקבוצה היא קבוצת אינדקסים, אולם חישובית לכן מלאה לכן לא ניתן להשתמש בר״ס.

חישוביות - תשובה לתרגיל

* I = תהי f קבוצת כל הפונ' ה-n שקיימת לה תוכנית המחשבת אותה כך שמס' הצעדים של התוכנית הזו אינו מתחלק ב-7.

R_f הקבוצה של כל התוכניות המחשבות פונ' אלה. $f = \{\text{כל הפונ' ה-n}\}$; $R_n = N$

כי תמיד ניתן לסדר תוכנית כך שמס' הצעדים שלה לא יתחלק ב-7.

הקבוצה הזו הקבוצה טריוויאלית, ולכן f מלאה. לכן לא ניתן להשתמש ברייס.

התוכנית המחשבת את הפרדיקט המופיע שלה:

$$\begin{cases} \text{IF } (x+1)_n = 0 \text{ GOTO } B \\ \text{GOTO } E \\ [B]\ Y \leftarrow 1 \end{cases}$$

הקבוצה כמו השאלה

$$g(x) = \begin{cases} 1 & \text{if } (x+1)_n = 0 \\ 0 & \text{else} \end{cases}$$

- וכלל גם כן לפי הקבוצה הנ"ל C = קבוצת התוכניות המחשבות את הפרדיקט המופיע של קבוצת התוכניות שמספרן לא מתחלק ב-7 - הקבוצה הזו חישובית לא ריקה.

* J = קבוצת כל הפונ' ה-n שקיימת לה תוכנית המחשבת אותה כך שבתוכנית לא מופיעה הפקודה מהסוג $Y \leftarrow Y+1$

ישנן פונ' שעבורן כן נצטרך להשתמש ואחר פונ' כאלה לא קיימת תוכנית המחשבת אותה בלעדיה. דוגמה $f(x)=0 \in J$ ו- $f(x)=1 \notin J$ לא ריקה ולא מלאה.

R_J קבוצת כל קודי התוכניות המחשבות את הפונ' שב-J. הקבוצה לא טריוויאלית ⇐ לא חישובית לפי רייס.

* K = קבוצת התוכניות אשר מחזירות 1 על הקלט x אם התוכנית שמספרה $r(x)$ עוצרת על הקלט $l(x)$, ואחרת - הן מחזירות 0. התוכניות שעונות על תיאור זה הן כאלה:

$$f(x) = \begin{cases} 1 & \text{if } \Phi(l(x), r(x))\downarrow \\ 0 & \text{else} \end{cases}$$

מחשבות פונ' שאינה לפרדיקט halt

לא ניתנת לחישוב

ולכן אין תוכנית המחשבת פונ' זו.

הרי ש $K=\emptyset$ והקבוצה ריקה = חישובית, אפילו ה"ה שכן הפרדיקט המופיע שלה הוא $f(x)=0$ וכמובן שלא ניתן להשתמש ברייס כי f ריקה.

* L = קבוצת התוכניות שמחזירות 1 על הקלט x אם לתוכנית שמספרה $l(x)$ יש לפחות 3 שורות, והשורה השלישית של התוכנית מופיעה בתוכנית שמספרה $r(x)$, ואחרת - הן מחזירות 0.

התוכניות שעונות על תיאור זה הן כאלה שמחשבות:

$$g(x) = \begin{cases} 1 & \text{if } Lt(r(x)+1) = 3 \wedge (r(x)+1)_3 = l(x) \\ 0 & \text{else} \end{cases}$$

ניתנת לחישוב

הפונ' g הנ"ל ח"ח, ולכן קיימת תוכנית המחשבת אותה.

R_L = אוסף כל התוכניות המחשבות את g שהן שייכות ל-L. $R_L = L$

מחשבות את g. ולכן $1 \notin f$, כמו כן L הוא קבוצה טריוויאלית

⇐ לא משנה רייס L לא חישובית (הסבר מפורט בהמשך)

חשיבות - חשוב לזכור!

א) הוכח שקיימת סדרה אינסופית של קבוצות חשיבות שהאיחוד שלהן הוא קבוצה לא חשיבה.
יש לחתוך את איברי הקבוצה הזו נקבל את הקבוצה K. $A_i = i \wedge k$
כיון שיש i נמצא בK נקבל את i אחרת נקבל ϕ קבוצה ריקה. האיחוד שווה לK.
וכל איבר זה הוא i - כאשר מס' סופי של איבר בודד - חשיב.

ב) הוכח שקיימת סדרה אינסופית של קבוצות חשיבות שהחיתוך שלהן הוא קבוצה לא במ"ש.
דוגמה: N_i יהיה $N-k_i$ יחתוך כל החלקים יחד הוא זה בK. $N-k$
יש לעשות חיתוך של הקבוצה הזו נקבל את $\bar{K}$ שהוא לא במ"ש
ולעומת זאת כל איבר לכשעצמו הוא חשיבי כי זה כל N פחות איבר יחיד.

ג) הוכח שקיימת סדרה אינסופית של פונ' שונות זו מזו כך שכל אחת מהן אינה חשיבה
דוגמה: הפונ' $f(x) = \begin{cases} i & \text{if } x=0 \\ \uparrow & \text{else} \end{cases}$ הפונ' לא חשיבות - לא מוגדרת על כל הא"ים
וכל פעם נציב i אחר וזה עדיין יהיה לא חשיב.

* חשוב לזכור הקבוצה $|W|$ או רק דוגמה: $2 \le |W|$ יהיה במ"ש לא חשיבי כי זה מאפשר
לן קבוצה שמוגדרת על 2 קלטים (בין היתר) בשפה המוקד"ש מוגדר הצירוף השני
ולכן (דוגמה: $2 = |W|$ או $2 \le |W|$ יהיה לא במ"ש).

* חשוב לזכור!

	תחום	טווח
$h(x) = \uparrow$ חח	ϕ	ϕ
$f(x) = x^2$ חשיבות	N	פונ' מלאה $\{x \mid \exists t (t \cdot t = x)\}$ כל הריבועים השלמים רק
$g(x) = HALT(x,x)$ לא חשיבות כלל	N	$\{0,1\}$
$f_2(x) = 0$ רק	N	$\{0\}$
$f_3(x) = \begin{cases} \uparrow & \text{if } x=3 \\ 0 & \text{else} \end{cases}$ חח	$N - \{3\}$	$\{0\}$

* שאלה: הוכיחו או הפריכו: יהיו A מוכלת שווה לB ו-B רקורסיבית (שאינה N) אז A רקורסיבית.
$A \subseteq B$. לא נכון! נראה דוגמה נגדית:

תהי $B = EVEN$ שהוא כמובן רקורסיבית
ותהי A קבוצה לא במ"ש המוכלת בB
$A = \{x \mid \Phi(x,x)\uparrow \wedge (2 | x)\}$ קבוצת הקלטים שלא עוצרים על קלט עצמם
זוגיות שלהן וגם הן מספרים זוגיים - כמובן לא במ"ש מכיון שמוגדר הבדיקה אי עצירה.
הוכחה ע"י רדוקציה של A לבעיה רדוקציה $\bar{K} \le A$

$$f(m) = \#\begin{cases} y \leftarrow y \\ z \leftarrow \Phi(m,m) \end{cases}$$

יש תוכנית זו עוצרת על קלט המקורי שלה כלומר $m \in \bar{K}$ וגם האיבר שמתקבל $2 | f(m)$
ואם התוכנית עוצרת כשקיבלה $y \leftarrow y$ מתקבל $f(m) \in A$ ו-A לא במ"ש כמו $\bar{K}$. והפוך.

חישוביות - תשובה לצבור!

* נבחין בשלושת הקבוצות הבאות:

- A = תוכניות שמחשבות פו' חח"ע שבעבור כל תוכנית שמחשבת אותה ומס' 6 אינו מחלק של הקלט שלה. הקבוצה ר"ה - וכמובן גם חישובית (כי היא ריקה)
- B = תוכניות שמס' 6 אינו מחלק של הקלט שלהן - הקבוצה ר"ה.
- C = תוכניות שמחזירות 1 אם מס' 6 אינו מחלק של הקלט של התוכנית המקודדת ע"י x. אחרת - הן מחזירות 0. - לא כ"מ (בדיקה אינסופית).

מדובר בתוכניות המחשבות פו' חישובית/ר"ה והפו' שלמה ולכן הקבוצה לא כ"מ.

נסביר קבוצה קבוצה בהרחבה:

- הקבוצה B היא ר"ה - הפרדיקט המאפיין של הקבוצה:

$$f(x) = \begin{cases} 1 & \text{if } \sim 6 \mid (x+1) \\ 0 & \text{else} \end{cases}$$

תוכנית

*ניתן לכתוב גם: (2 ו-3 לא מחלקים -)

$(x+1)_1 = 0 \quad (x+1)_2 = 0$

מכיון שמדובר בתכונה לא סמנטיקורית, ולכן תנאית קיימת אולי היא מבקשת להשתמש בר"ס לא יכול לעשות כי הקבוצה הזו לא קבוצת אינדקסים.

נראה 2 תוכניות שונות המחשבות את הפו' $f(x)=1$ אבל אחת תשייך לקבוצה והשנייה לא.

$$q = \# \begin{cases} Y \leftarrow Y \\ Y \leftarrow Y \\ Y \leftarrow Y+1 \end{cases} \qquad p = \# \begin{cases} Z \leftarrow Z+1 \\ Y \leftarrow Y+1 \end{cases}$$

$1 = f(x) = \Psi_p(x) = \Psi_q(x)$ אבל $q \in B$ ואילו $p \notin B$

p הוא מחלק של הקלט, כי p יש 2 פקודות כמו $[2^{\square}3^{\square}\dots-1]$ ולכן 6 לא מחלק של הקלט.

- הקבוצה A מכילה מספר את כל הפונקציות החח"ע כלומר את כל N מכיון שאם נדרוש שהתוכנית בעצמה תקיים את התנאי הנ"ל ואז קוד של תוכנית פשוטה מספיק בקוד להוסיף 2 פקודות כמו מבחינה ולכן 6 לא יחלק את הקלט. ולכן היא ר"ה. ולכן לא צריך להשתמש בר"ס כיון ש-A מלאה בגלל שהיא מכילה את כל N. (כמובן שN היא כן קבוצת אינדקסים).
- הקבוצה C אלו תוכניות המחשבות את הפרדיקט המאפיין מהשאלה והם של הקבוצה B כמובן שהקבוצה לא כ"מ מכיון שמדובר בתוכניות המחשבות פו' ר"ה - אבל פו' שלמה ולכן מדובר בבדיקה לכל x - שמשמע בדיקה אינסופית - אינה כ"מ בדיקה = לא ניתן להוכיח ע"י חישוביות גם ר"ס.

C מחזירה R_A עבור x המכילה את הפרדיקט המאפיין של B (הוא ר"ה) לכן היא קבוצת אינדקסים. כאשר $f(x) \in$ ולכן $f \notin$ $\Leftarrow$ C לא חישובית גם ר"ס.

בס"ד

בחינות בחישוביות

* הפונקציה הריקה מחשבת את הפרדיקט האופייני של הקבוצה הריקה ≥ שאמור $f(x)=0$.
* הפונקציה $f(x)=1$ היא פרדיקט אופייני חישובי והוא פ' ר"פ.
* כל פ' ר"פ היא רקורסיבית, אך לא כל פ' רקורסיבית היא ר"פ.
* תוכנית בשפת S לעולם מחשבת פונקציה. (לכל פ' ניתנת לחישוב יש תוכנית בשפת S).
* לא כל פונקציה שמוגדרת ניתנת לחישוב ע"י תוכנית בשפת S (דוג': Halt - מוגדרת אך לא חישובית).
* המחלק המינימלי הקטן ביותר של שני מס' הוא תמיד 1, לעומת זאת המכפיל המשותף הקטן ביותר של שני מספרים הוא המס' הראשון שגם x וגם y הם מחלקים שלו. (הוא מתחלק בשניהם ללא שארית).
* ניתן למצוא את המחלק המקסימלי של שני מספרים ע"י שימוש במינימליזציה חסומה.
* ניתן למצוא את המס' המקסימלי מבין שני מספרים ע"י שימוש במינימליזציה חסומה.
* יש פרדיקט חישובי, והמינימליזציה לא חסומה, אז הפ' המוגדרת באמצעותו היא חישובית חלקית.
* פונקציית העוקב היא חח"ע אך לא על.
* פונקציית קידוד התוכניות היא חח"ע ועל.
* אם x הוא קידוד של תוכנית, וידוע ש-3 אינו מחלק של $(x+1)$, אז הפקודה הראשונה בתוכנית היא $Y \leftarrow Y$.
* תוכנית שקידודה אי-זוגי יכולה להסתיים בהוראה $Z \leftarrow Z$ (קידוד אי-זוגי ⇒ בגלל זוגי ⇐ מעריך החזקה של 2 שונה מ-0 ⇒ הוראה הראשונה מההוראות הבאה $Y \leftarrow Y$).
* התוכנית שקידודה x כוללת 3 פקודות בסיסיות בלבד אם"ם $lt(x+1)=3$ (lt פונקציית אורך!).
* לכל מס' טבעי קיימת תוכנית שקידודה זוגי המחשבת אותה (ניתן להוסיף פקודות כאלה).
* אלגוריתם שמקבל קבוצה עוצר ומחזיר 1 עם איבר השייך לקבוצה ועוצר ומחזיר 0 עם איבר שלא שייך לקבוצה.
* קבוצת התוכניות שמחשבות את הפ' $f(x)=3$ היא לא במ"ר (אף ש-$f(x)=3$ היא פ' חישובית, מכיוון שמדובר בקבוצה אינסופית).
* קבוצת התוכניות שעוצרות עם 3 קלטים בדיוק היא לא במ"ר (בדיקת אי עצירה עד האינסוף).
* N וגם הקבוצה הריקה הן קבוצות ר"פ, כיוון שהפרדיקטים האופייניים שלהם הם פ' ר"פ.
* Empty - זו קבוצת התוכניות אשר קיים קלט אחד לפחות שעליו הן עוצרות.
* $\bar{K}$ - היא קבוצה לא במ"ר (קבוצת תוכניות שלא עוצרות על קלט השווה לקידוד שלהן).
* אם קבוצה ר"פ, אז הפרדיקט האופייני שלה הוא פ' ר"פ.
* יש קבוצה שהיא רקורסיבית, אך קיים לה פרדיקט שמקבל אותה.
* אם קבוצה במ"ר אז קיים לה אלגוריתם שמקבל אותה.
* לכל קבוצה בעולם קיים פרדיקט אופייני שמקבל את איברי הקבוצה.

בחנץ חשובות

* תחום ההגדרה של פונ' המתבדרת h הוא קבוצה ר"ה (התחום והטווח הם ϕ)
* הטווח של הפרדיקט האופייני של הקבוצה הריקה הוא קבוצה ר"ה שאיננה ריקה (הקבוצה הריקה, התחום: N, הטווח: 0).
* תהי A קבוצת התוכניות שמחשבות את הפרדיקט האופייני של N, אפשר להוכיח ש-A לא כמ"ר ע"י שימוש באלכסון (הפרדיקט האופייני של N הוא 1: $f(x)=1$ שהיא פונקציה שעליה לכן ניתן להשתמש באלכסון).
* אם לא קיימת פו' רקורסיבית g כך ש $Range(g)=C$ אז הקבוצה C היא לא כמ"ר.
* תהי הקבוצה A שלהלן $A=\{n \mid Dom(\phi_n)=\{1\} \text{ and } Range(\phi_n)=\{1\}\}$
הקבוצה A לא כמ"ר כי רק המספר 1 (נשאר בתחום - המספר לא נמצא - בדיקה אינסופית).
* תהי B הקבוצה שלהלן $B=\{n \mid |Dom(\phi_n)|=18 \text{ and } Range(\phi_n)=\phi\}$
הקבוצה הזאת לא ריקה, תחום אלה המספרים שמצומצם שאותם את הקלט מיידי ומכיון שיש לנו X אובייקטים לא יכול להיות שהטווח ריק. לכן הקבוצה כמ"ר (ואפילו חישובית).
* התחום של פונ' הריקה הוא N והטווח הוא $\{0\}$. שניהם קבוצות ר"ה.

חישוביות - תרגול 6

הקורס עוסק במה מחשב לא יכול לחשב. יש דברים שלא קיים אלגוריתם שיכול לחשב אותם, לא כל תכונה תמיד ניתנת לבדיקה ובעיית הבדיקה ספציפית.
(אין מצפים לתשובה בסוף הבדיקה או לפחות לעצירה).
לא קיים אלגוריתם שמקבל קלט ותוכנית ובודק האם הוא לא עוצרת. (בעיית העצירה)
ניתן לבדוק עצירה ואם לא אז עצרה. נשתמש בה בכדי חישובים לשם כך:

1. $stp(3, 7, 19)$ - פרדיקט חישובי השואל במקרה זה האם התוכנית שקודדה 7, הקלט מס׳ התוכנית מס׳ הצעדים
מריצים על הקלט 3 עוצרת תוך 19 צעדים
יש תוכנית 7 על הקלט 3 עוצרת בצעד הקשור שווה 19 יוחזר 1, אחרת - יוחזר 0.
השוני מהותי: כיוון לבין לעצור שהבדיקה מתבצעת בצעד-רגע ספציפי.
ובכך מסמן את השאלה ומפרט אותה לבעיית הבדיקה.

$Snap(3, 7, 19) = <W, Z>$

מס׳ הקלט, מס׳ התוכנית, מס׳ הצעד, מס׳ הפקודה הבאה שתבוצע (המצב של המשתנים)

* תזכורות: משתני התוכנית מופיעים בסדר הבא:
$Y, X_1, Z_1, X_2, Z_2 \dots$

snap הוא סוג חישובים, שמקבל שמכיל עוצרת ומחזירה ערך ומאפשרת לבדוק את התוכנית, בריצתה על קלט מסוים ברגע נתון, משמע כמין "תמונה" - צילום מצב של ריצה ברגע ספציפי, ולא להגיע עד כל אחד מהמשתנים שלה, וכן מאפשרת לקבוע באיזו מספר הפקודה הבאה להגיע אותה בהרצה הולכת לבצע התוכנית.

* קבוצה חישובית - רקורסיבית - קיים אלגוריתם שמסוגל לבדוק שייכות ואי שייכות לקבוצה.
* קבוצה כמו"ע לא חישובית - קיים אלגוריתם שמסוגל לבדוק רק שייכות לקבוצה.
* קבוצה לא כמו"ע - לא קיים אלגוריתם שמסוגל לבדוק אפילו שייכות לקבוצה.

- בכלל: במקרה של פרדיקטים יש לנו רק 2 אפשרויות: או שהם חישוביים ממש (רקורסיביים) או שהם אינם חישוביים בכלל.

1. קבוצת תוכניות שעוצרות על הקלט 7 = כמ"ע לא חישובית.
2. קבוצת התוכניות שעוצרות רק על הקלט 7 - לא כמ"ע (כי מחשב לא עוצרות)
3. קבוצת התוכניות שעוצרות לכל קלט (TOT) = לא כמ"ע (בדיקה אינסופית).
4. קבוצת התוכניות שיש קלט שהן לא עוצרות (Empty) = כמ"ע לא חישובית
5. קבוצת התוכניות שקידודן זוגי = חישובית רגילה.
6. קבוצת התוכניות המחשבות את הפונ׳ $f(x)=x^2$ = לא כמ"ע (פו׳ שלמה - בדיקה לכל x)
7. קבוצת התוכניות כך ש $|W_n| = 15$ = לא כמ"ע
8. קבוצת זוגות $<n,m>$ כך ש $|W_m| = |W_n|$ = לא כמ"ע

חישוביות - תרגול 6

* קבוצת רביעיות <<<a,b,c>,10>> כך שהתוכנית a כשמריצה על הקלט b, תמונת החישוב בצעד ה-10 = c חישובית כרגיל.
* קבוצת שלשות <<a,b,c>> כך שהתוכנית a כשמריצה על הקלט b, תמונת החישוב בשלב כלשהו היא c = בר"מ לא חישובית.
* A - תוכניות שלא עוצרות על הקלט 3 = לא בר"מ (אי עצירה)
* B - תוכניות שעוצרות על הקלט 3 - בר"מ לא חישובית
* C - תוכניות שעוצרות רק על הקלט 3 - לא בר"מ (אי עצירה על השאר)
* D - תוכניות שעוצרות על 3 קלטים בדיוק (מוכלות בתוך FIN) - לא בר"מ (אי עצירה על השאר)
* TOT - תוכניות שעוצרות לכל קלט - לא בר"מ (בדיקה אינסופית)
* FIN - תוכניות שעוצרות על מס' סופי של קלטים - לא בר"מ (בדיקה אינסופית)
* Empty - תוכניות שלא עוצרות לשום קלט - לא בר"מ (אי עצירה, בדיקה אינסופית)
* $\overline{Empty}$ - תוכניות שקיים קלט עליו הן עוצרות - בר"מ לא חישוביות.
* K - תוכניות שעוצרות על קלט המהווה את הקידוד שלהן - בר"מ לא חישובית.
* $\bar{K}$ - תוכניות שלא עוצרות על קלט המהווה את הקידוד שלהן - לא בר"מ (אי עצירה)
* INF - תוכניות שעוצרות על אינסוף קלטים (מכילה את TOT) - לא בר"מ (בדיקה אינסופית)
* תוכניות שלכל x מחשבות פו' חח - חישובית כרגיל ([illegible])
* Even - תוכניות שקידודן זוגי - חישובית כרגיל.
* E - תוכניות שלכל x מחשבות את הפו' x=f(x) - לא בר"מ
* $\bar{E}$ - תוכניות שקיים קלט שעבורו הן מחשבות את הפו' $f(x)=x^2$ = לא בר"מ
* F - תוכניות המחשבות את הפו' שתחום ההגדרה שלה הוא {3} - לא בר"מ
* Q - תוכניות שעוצרות על 5 קלטים ומהקבוצה הנ"ל E - לא בר"מ
- מה לא בדיק? 1. אי עצירה של תוכנית. 2. משהו שמצריך בדיקה אינסופית.

∇ יש קבוצה חישובית-כרגיל, שמשלימה גם חישובית-כרגיל.
∇ יש קבוצה בר"מ לא חישובית, שמשלימה היא לא בר"מ
∇ יש קבוצה לא בר"מ, שמשלימה גם היא אינה חישובית (יכולה להיות לא בר"מ/בר"מ לא חישובית).

* קבוצה היא חישובית-כרגיל-רקורסיבית וגם המשלים שלה חישובית-רקורסיבי.
- אם צריך להוכיח שקבוצה לא חישובית, צריך להוכיח גם שהמשלים שלה לא חישובי.

מסקנות מתוך המבחנים

* פונקציה שלמה היא ממש לא בהכרח חישובית.
* פונקציה חישובית היא בהכרח שלמה.
* כל פונקציה חישובית היא רקורסיבית.
* ממש לא כל פונקציה חישובית היא ר"פ.
* הפונקציה $\uparrow$=ממש היא פונקציה חישובית חלקית.
* הפונקציה $g(x)=\frac{1}{x}$ היא לא פונקציה חישובית חלקית - היא כלל לא חישובית.
* הפונקציה $f(x)=\begin{cases}1 & x=3\\ \uparrow & else\end{cases}$ היא פונקציה חישובית חלקית.
* הפרדיקט האופייני של הקבוצה הריקה הוא $f(x)=0$
* הפרדיקט האופייני של הקבוצה $\mathbb{N}$ הוא $f(x)=1$
* יש $\sim(x|2)$ אם x הוא מס' אי זוגי - לא נכון, יש $\sim(2|x)$
* התוכנית הריקה אינה מחשבת כלום - לא נכון! מחזירה $y=0$ (הערך הדיפולטיבי)
* התוכנית הריקה מחשבת פונקציה שמחזירה פרדיקט אופייני עבור הקבוצה הריקה
* התוכנית A GOTO [A] אינה מחשבת דבר, אך כן מחשבת פונ' (מחשבת את $\uparrow$=ממש)
* מס' הפרמטרים שמופיעים כל מורכבת לעקב לא סגר מס' הפרמטרים שמופיעים בה בפונקציה המורכבת.
* פי המספר בקרב מחלק ניתנות ל"ג קיק שמוש מורכבת של פו' ר"פ.
* בהינתן הפו' $f(x)=0$, האם הנוסחה הרקורסיבית הבאה מתאימה להגדרה?

לא נכון, כי היה צריך לכתוב $\pi_2^2(t,f(t))$

$$\begin{cases} f(0)=0\\ f(t+1)=\pi_1^2(t,f(t))\end{cases}$$

* בהינתן הפונקציה $f(x,y)=\min_{z\le x\cdot y}((x|z)\wedge(y|z))$

הפו' מחשבת את המכפלה המשותפת המינימלית של x ו-y (המכפלה המשותפת הקטנה ביותר שזה המס' הראשון שגם x וגם y הם מחלקים שלו)

- יש הבדל חשוב: פו' שמחשבת את המחלק המינימלי של שני מספרים: $\min_{z\le x+y}((z|x)\wedge(z|y))$

הפו' תמיד תחזיר 1 כי הוא המחלק המינימלי תמיד.

* תוכניות שקיצורן זוגי פותחות בפקודה $Z\leftarrow Z$ - לא נכון, פותחות ב $Y\leftarrow X$.
* תוכניות שקיצורן אי זוגי פותחות בפקודה השנייה $Y\leftarrow Y$
* ניתן להוכיח שכל הפונקציות ר"פ למרות שיש לה יותר מפעמיים ... מס' פעמי קידוד.
* בהינתן x המתאר קידוד של תוכנית - הפונקציה f שלהלן היא הפרדיקט האופייני

המתאים לקבוצת התוכניות שאורכן גדול מ-17

$$f(x)=\begin{cases}1 & Lt(x+1)>17\\ 0 & else\end{cases}$$

סוף

מסקנות מתוך התרגולים

* כשנתון x ורוצים לבדוק שכל הספרות שונות, כלומר $x=[a_1,a_2...a_n]$, נבדוק לכל פעם רק את החזקה המינימלית מבין אברי הסדרה המקודדת ע"י x, נבדוק שגדול מהמינימום הכולל.

$$f(x) = \min_{z \le x} ((x)_1 \ge z \wedge (x)_2 \ge z \wedge (x)_3 \ge z \dots \wedge (x)_n \ge z)$$

* נכתוב את הפרדיקט הזה כך: $p(x) = \sim(2|(x+1)) \wedge (x+1)_3 = 0$
כשנתון x שמקודד סדרה של פונקציות, נבדוק אילו תוכניות יחזירו $p(x)=1$,
נבחר פונקציות שהגבול שלהן אי-זוגי, והפונקציה השלישית שלהן היא $Y \leftarrow Y$ (כלומר $1+$ פונקציה)
* תהי B קבוצת התוכניות המחשבות את הקבוע 7. ההגדרה הפורמלית תהיה כזו:
$B = \{n | 7 \in W_n\}$
* הפרדיקט המגדיר את אוסף הקלטים שעליהם התוכנית 5 עוצרת W_5: $W_5 = \{x | \exists z\, stp(x,5,z)\}$
* לכל קבוצה סופית קיים פרדיקט אופייני. הסבר: לכל קבוצה סופית יש פרדיקט אופייני!
* הקבוצה N היא קבוצה שהפרדיקט האופייני שלה הוא פ"ר קבוע. הסבר: הוא $f(x) = 1$.
* הפונקציה המחושבת ע"י תוכנית שעוצרת לכל קלט היא פ"ר חישובית, וקבוצת התוכניות שעוצרות לכל קלט היא לא כמ"ר. הסבר: זהו לא כמ"ר, היא משלימה של זהו התוכניות הנ"ל.
* תכנית שאינה מחשבת את הפונקציה הריקה, אזי הפרדיקט האופייני שלה הקבוצה הריקה.
* אם הפרדיקט האופייני של הקבוצה הוא פ"ר, הקבוצה המשלימה גם היא פ"ר.
* ההגדרה של פ"ר רקורסיבית היא קבוצה בת מנייה חישובית (כאשר ההגדרה היא אם הקבוצה שלמה).
* המשוואה של פ"ר רקורסיבית יכולה להיות כל קבוצה שהיא בהכרח חישובית.
* תוכנית שעוצרת לכל קלט, לא בהכרח מחזירה קבוצה סופית של קלטים (0 היא קבוצה סופית).
* תהי A קבוצת התוכניות שעוצרות אי-זוגי - הקבוצה A לא מכילה תוכניות שעוצרות בהכרח $Y \leftarrow X$, הקבוצה A חישובית-בריאה, הקבוצה A היא אינסופית,
הפרדיקט האופייני של הקבוצה הוא $\sim(2|x)$, כל התוכניות המחזירות אי-זוגי שייכות לקבוצה.
מכילה תוכניות שהגבול שלהן הוא זוגי
* הקבוצה המשלימה לקבוצה A היא $\bar{K}$: $A = \{x | \forall z \text{ not } stp(x,x,z))$