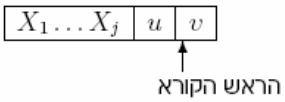
הוכחת הטענות מסוף הרצאה 8:

יהא $X_1 \dots X_j$ רישא חיוני של דקדוק G $LR(k)$, ותהא $T = \langle f, g \rangle$ הטבלה המתקבלת מ־ $V_k(X_1 \dots X_j)$. נניח ששארית הקלט כרגע היא v, ונסמן $u = \mathrm{FIRST}_k(v)$ (כלומר u הוא ה־ look-ahead של האלגוריתם).

אלגוריתם לבניית עץ הגזירה של מילה

נשמור במחסנית ניתוח את הרישא החיוני שנקרא עד־עתה ואת הטבלאות המתאימות לכל מצב. נניח שתוכן המחסנית הוא $T_0 X_1 T_1 \dots X_j T_j$ ו־ w – שאר הקלט. האלגוריתם מחזיק בזכרון את k האותיות הראשונות של שארית הקלט (או עד סוף הקלט אם יש בו פחות מ־ k אותיות), כלומר מילה u כך ש־ $|u| \le k$ ו־ $w = uv$.



נניח שבראש המחסנית נמצאת $T_j = \langle f, g \rangle$.

1. בצע פעולה לפי פונקציית הניתוח:
   - אם $f(u) = \mathrm{ACCEPT}$, קבל.
   - אם $f(u) = \mathrm{ERROR}$, דחה.
   - אם $f(u) = \mathrm{SHIFT}$, העבר את האות הראשונה מ־ u למחסנית ואם $v \neq \varepsilon$, העבר את האות הראשונה של v לסוף u.
   - אם $f(u) = \mathrm{REDUCE}_i$ כש־ i הוא כלל $A \to \alpha$, הוצא $2|\alpha|$ סימנים מהמחסנית ($|\alpha|$ סימנים של הדקדוק והטבלאות הצמודות לכל אחד), והכנס במקומם את A.
2. דחוף את $g(u) = T'$ לראש המחסנית.

טענת עזר: כאשר נמצאים במצב עם פריט $[A \to \beta_1.\beta_2, u]$ אז β_1 היא סיפא של תוכן המחסנית.
כלומר הפריט תקף עבור תוכן המחסנית.
נוכיח זאת באינדוקציה על האורך של β_1.

בסיס: הפריט הוא מהצורה: $[A \to .\beta, u]$ וכמובן ש ε היא סיפא של כל דבר.

צעד: $[A \to \alpha X.\beta, u]$

הגענו פריט הזה עם $GOTO(V_k(\alpha), X)$, כלומר הפריט הקודם היה $[A \to \alpha.X\beta, first_k(\beta u)]$
לפי הנחת האינדוקציה α כבר נמצא במחסנית ולכן לאחר הכנסת X בפעולת ה $GOTO$, הטענה מתקבלת.

משפט: אם $X_1 \dots X_n$ הוא תוכן המחסנית (מבחינת משתנים וטרמינלים) אחרי קריאת w אזי $X_1 \dots X_n \Rightarrow_R^* w$.
ההוכחה באינדוקציה על מספר צעדי האלגוריתם:
בסיס: עבור אפס צעדי גזירה נקבל ש $w = \varepsilon$ והמחסנית ריקה, כלומר הטענה מתקיימת.
נניח נכונות עבור מספר צעדי האלגוריתם קטן מ n ונוכיח עבור n.
אם הצעד האחרון הוא REDUCE או SHIFT הטענה גם מתקיימת:
עבור פעולת SHIFT, פשוט מכניסים עוד אות למחסנית וזוהי בדיוק האות שקוראים מהקלט, לכן הטענה מתקיימת.
עבור פעולת REDUCE, מחליפים את סיפא המחסנית במשהו שגוזר אותו, לכן הטענה מתקיימת.
איך יודעים שמה שמתחלף בתוך המחסנית מתחלף אכן על פי כלל גזירה חוקי? זה נובע מטענת העזר.

1. אם לא קיים $w \in L(G)$ כך ש־ $X_1 \dots X_j v \Rightarrow_R^* w$ אז האלגוריתם פולט ERROR.

2. אם $f(u) = \text{SHIFT}$ אז

$$S \Rightarrow_R^* X_1 \dots X_\ell A w \Rightarrow X_1 \dots X_j w_1 w = X_1 \dots X_j uv$$

כך ש־ $\ell < j$ ו־ $w_1 \neq \varepsilon$.

3. אם $f(u) = \text{REDUCE}_i$ והכלל ה־ i הוא $A \to \alpha$, אז α הוא סיפא של $X_1 \dots X_j$.

על פי טענת העזר, כאשר נמצאים במצב עם פריט $[A \to \beta_1.\beta_2, u]$ אז β_1 היא סיפא של תוכן המחסנית, כלומר, סיפא של $X_1 \dots X_j$.

אם $f(u) = REDUCE$ אז על פי הגדרת הטבלה, מתקבל $\beta_2 = \varepsilon$. כלומר הכלל $A \to \alpha$ מתאים לפריט $[A \to \beta_1., u]$ ולכן בשימוש בטענת העזר מתקבל ש $\alpha = \beta_1$ והוא סיפא של תוכן המחסנית.

4. אם $f(u) = \text{ACCEPT}$ אז $u = \varepsilon$, תכולת המחסנית היא $T_0 ST$, ו־ $[S' \to S\bullet, \varepsilon] \in T$.

הדבר היחיד שאינו ישירות ע"פ ההגדרה הוא שתכולת המחסנית היא $T_0 ST$.

5. אם $f(u) = \text{ERROR}$ אז $g(u) = \text{ERROR}$.

תהי T מכונת טיורינג בעלת סיבוכיות מקום $L(n)$ ויהי R_n המספר המקסימאלי של תצורות סרט העבודה של T על קלט באורך n.
נניח כי התצורות הנ"ל ממוספרות במספרים מ 1 ועד R_n.
עבור מילה u ומספר n המקיימים $|u| \le n$, נגדיר את המטריצה: $\mathrm{T}^n(u) = (t_{i,j})$ בגודל $R_n \times R_n$ באופן הבא:
הכניסות של $\mathrm{T}^n(u)$ הן 00, 01, 10 או 11.
תהי $w = vu$ מילה כך ש $|w| = n$ קלט של המכונה T.

אם T נמצאת בתצורה i בסרט העבודה, עם הראש על סרט הקלט בסימן הראשון (השמאלי ביותר) של u ויכולה להגיע לתצורה j בסרט העבודה, מבלי לצאת מ u, אזי הסימן הראשון של $t_{i,j}$ הוא 1.
אחרת (כלומר T חייבת לצאת מ u בשביל להגיע לתצורה j או לא יכולה בכלל להגיע לתצורה j מתצורה i), הסימן הראשון של $t_{i,j}$ הוא 0.

אם מהתצורה i המכונה T יכולה לצאת מ u, ובפעם הראשונה שהיא יוצאת היא מגיעה לתצורה j, אזי הסימן השני של $t_{i,j}$ הוא 1. אחרת הסימן השני של $t_{i,j}$ הוא 0.

<u>למה:</u> תהי T מכונת טיורינג בעלת סיבוכיות מקום $L(n)$, ויהיו $w_1, w_2 \in L(T)$ מילים המתקבלות על ידי T. אם $|w_1| = |w_2| = n$ וגם $w_1 = v_1u_1$, $w_2 = v_2u_2$ וגם $\mathrm{T}^n(u_1) = \mathrm{T}^n(u_2)$ אזי $v_1u_2 \in L(T)$.

<u>הוכחה:</u> נתבונן בסדרת התצורות של T שמקבלת את v_1u_1.
אם הראש על הסרט הראשון לא חוצה את הגבול בין v_1 לבין u_1, ברור שאם על הסרט הראשון היה v_1u_2, המכונה הייתה מבצעת בדיוק את אותם צעדים שעשתה עבור v_1u_1 ולכן הייתה מקבלת גם את v_1u_2, ולכן $v_1u_2 \in L(T)$.

נראה מה קורה אם הראש על הסרט הראשון כן חוצה הגבול הנ"ל, ומגיע ל u_2/u_1.
ברגע זה, התצורות זהות (כי עד כה עברנו רק על v_1 שהוא ההתחלה של שתי המילים v_1u_1, v_1u_2).
אם כעת על v_1u_1 המכונה יכולה להגיע למצב מקבל בלי לצאת מ u_1, אזי גם על v_1u_2 היא תגיע למצב זה, כי התצורות זהות עבור שתי המילים כי הסימן הראשון בשתי הטבלאות זהה.
אם ראש המכונה יוצא, וחוזר אל v_1 בסרט הקלט, אז ברגע היציאה הוא מגיע בדיוק אל אותה התצורה עבור שתי המילים, כי הסימן השני בשתי הטבלאות זהה.
ומכאן ממשיכים כמו בהתחלה, עד שנגיע למצב מקבל, או עד שנחצה שוב ונגיע עם הראש על סרט הקלט ל u_2/u_1 ונמשיך באותו אופן.

הוכחת המשפט מההרצאה הקודמת:

<u>משפט:</u> תהי T מכונת טיורינג בעלת סיבוכיות מקום $L(n)$, כאשר הפונקציה $L(n)$ איננה חסומה.

אזי: $\limsup \frac{L(n)}{\log\log n} > 0$.

כלומר קיים קבוע c כך שעבור אינסוף n-ים מתקיים: $L(n) \geq c\log\log n$.

נגדיר: $n_k = \min\{n \mid L(n) \geq k\}$ - כלומר המספר n הקטן ביותר, שקיימת עבורו מילה באורך n, שעליה המכונה צריכה להשתמש בלפחות k תאים בסרט העבודה.

(לדוגמה, אם $L(n) = n^2$, אז $n_k = \sqrt{n}$)

n_k מוגדר היטב משום שהפונקציה $L(n)$ איננה חסומה. (לא קיים k ש $L(n)$ לא עוברת אותו).

יהי R מספר תצורות סרט העבודה שמכילות לכל היותר $L(n_k)$ תאים.

יהי w קלט באורך n_k כך ש T חייבת להשתמש בלפחות k תאים בסרט העבודה שלה.

נראה שאי אפשר לכתוב את w כ: $w = v_1v_2v_3$ כך ש: $v_2 \neq \varepsilon$ ו $\mathrm{T}^{n_k}(v_3) = \mathrm{T}^{n_k}(v_2v_3)$.

נניח בשלילה שהמצב הנ"ל כן אפשרי ונראה כי על הקלט v_1v_3, T חייבת להשתמש בלפחות k תאים בסרט העבודה. זה יסתור את ההגדרה של n_k כי על פי הגדרתו, w היא המילה הקצרה ביותר כך ש T חייבת להשתמש בלפחות k תאים בסרט העבודה בריצה עליה, בעוד ש v_1v_3 הקצרה ממנה מקיימת גם כן את הדרישה הזאת.

אם המכונה בריצתה על $w = v_1v_2v_3$ מגיעה לתצורה בעלת k תאים על סרט העבודה לפני שחצתה את הגבול בין v_1 לבין v_2, הרי שגם בריצה על v_1v_3 היא תגיע לתצורה זאת.

אם היא כן חוצה את הגבול הזה מספר פעמים לפני שהיא מגיעה לתצורה בעלת k תאים, אז כמו בלמה, שם היא בשני המקרים למצב מקבל, כאן היא תגיע לתצורה בעלת k תאים בסרט העבודה.

לכן לא קיימות ל w שתי סייפות $v_2v_3 \neq v_3$ כך ש $\mathrm{T}^{n_k}(v_3) = \mathrm{T}^{n_k}(v_2v_3)$.

לכן קיימות לפחות $n_k - 1$ מטריצות $\mathrm{T}^{n_k}(u)$ בגודל $R \times R$.

מספר המטריצות מהצורה הזאת חסום על ידי 4^{R^R}.

משום ש $R \leq s \cdot L(n_k) \cdot t^{L(n_k)}$ (בכל תצורה המכונה יכולה להיות בכל אחד מ s המצבים, הראש יכול להיות בכל אחד מ $L(n_k)$ התאים, ובכל אחד מ $L(n_k)$ התאים יכולה להיות כתובה אחת מ t אותיות)

נובע כי $n_k - 1 \leq 4^{\left(s \cdot L(n_k) \cdot t^{L(n_k)}\right)^2}$.

נוציא לוג בפעמים ונקבל: $\log\log(n_k - 1) \leq 2\left(\log(s) + \log(L(n_k)) + L(n_k)\log(t)\right)$.

לכן עבור $L(n_k)$ מספר גדול נקבל ש: $\frac{L(n_k)}{\log\log(n_k)} \geq c$ קבוע.

לכן המשפט נכון עבור אינסוף n-ים שהם ה- n_k-ים עבור $k = k_1, k_1+1, k_1+2, \ldots$.

מודל החישוב: מכונת טיורינג בעלת סרט אחד עם ראש אחד. הסרט הוא אינסופי רק בכיוון אחד.

<u>הגדרה</u>: מכונת טיורינג נקראת בעלת סיבוכיות מקום $L(n)$, אם על כל קלט באורך n היא משתמשת לכל היותר ב $L(n)$ תאים.

<u>הגדרה</u>:
$PSPACE = \{L \mid$ כאשר T היא מ"ט דטרמיניסטית בעלת סיבוכיות מקום פולינומי $L = L(T)\}$
כלומר אוסף כל השפות שיש להן מכונות הדורשות מקום פולינומי ביחס לגודל הקלט שלהן.

$NPSPACE = \{L \mid$ כאשר T היא מ"ט **לא** דטרמיניסטית בעלת סיבוכיות מקום פולינומי $L = L(T)\}$

ברור כי $PSPACE \subseteq NPSPACE$.

<u>משפט</u>: $PSPACE = NPSPACE$

<u>למה:</u> $(Savitch)$: אם שפה L מתקבלת על ידי מ"ט לא דטרמיניסטית בעלת סיבוכיות מקום n^k, אזי L מתקבלת על ידי מ"ט דטרמיניסטית בעלת סיבוכיות מקום $O(n^{2k})$.

<u>הוכחה:</u> תהי $L = L(M_1)$, כאשר M_1 היא מ"ט לא דטרמיניסטית בעלת סיבוכיות מקום n^k.

מספר התצורות ש M_1 יכול להיות בהן בהינתן קלט באורך n הוא לכל היותר c^{n^k} כאשר c הוא קבוע שתלוי ב M_1.

לכן אם M_1 מקבלת קלט באורך n, היא יכולה לקבל אותו תוך לכל היותר c^{n^k} תנועות (אחרת היא חוזרת על אותה תצורה פעמים, וזה אומר שהיא נכנסה ללולאה אינסופית).

נסמן על ידי $I_1 \mid -^i I_2$ את העובדה שמתצורה I_1 ניתן להגיע לתצורה I_2 ב 2^i או פחות תנועות.
על מנת לבדוק האם המכונה מקבלת את המילה, מספיק לבדוק האם ניתן להגיע מהתצורה ההתחלתית I_i אל התצורה המקבלת I_f, תוך c^{n^k} צעדים.
עבור $i \geq 1$ ניתן לבדוק האם $I_1 \mid -^i I_2$ על ידי בדיקת $I_1 \mid -^{i-1} I' \mid -^{i-1} I_2$ עבור כל תצורה I'.
לכן המקום הדרוש לבדיקת $I_1 \mid -^i I_2$ הוא חסום על ידי המקום הדרוש כדי לזכור את כל ה I' שנבדקו ועוד המקום הדרוש לבדוק האם $I' \mid -^{i-1} I_2$.
הערה: במקום שמשתמשים בו כדי לבדוק האם $I' \mid -^{i-1} I_2$, ניתן להשתמש לבדיקה נוספת לאחר מכן.
אם נניח כי התצורות מסודרות על פי הסדר הלקסיקוגרפי, אז מספיק לזכור רק את התצורה האחרונה שנבדקה.
בכל שלב של הרקורסיה צריך לזכור את i, I_1, I_2, I' שדורש $O(n^k)$ תאים.

משום שהרקורסיה נקראת לכל היותר $\log(c^{n^k})$ פעמים, שהוא $O(n^k)$, סה"כ המקום הדרוש לביצוע הרקורסיה הוא $O(n^{2k}) = O(n^k) \cdot O(n^k)$.

הוכחת המשפט: $PSPACE = NPSPACE$

נסמן:

$SPACE\left(n^k\right) = \{L \mid L = L(T)$ כך ש n^k בעלת סיבוכיות מקום T דטרמיניסטית מ"ט קיימת$\}$

$NSPACE\left(n^k\right) = \{L \mid L = L(T)$ כך ש n^k בעלת סיבוכיות מקום T דטרמיניסטית מ"ט **לא** קיימת$\}$

נקבל:

$$PSPACE = \bigcup_{k=1}^{\infty} SPACE\left(n^k\right)$$

$$NPSPACE = \bigcup_{k=1}^{\infty} NSPACE\left(n^k\right)$$

לכן אם $L \in NPSPACE$ אז קיים k כך ש $L \in NSPACE\left(n^k\right)$ ולכן על פי הלמה, $L \in SPACE\left(n^{2k}\right)$ ולכן $L \in PSPACE$. מש"ל.

הגדרה: השפה $L \subseteq \Sigma^*$ נקראת PSPACE שלמה ביחס לזמן פולינומי, אם $L \in PSPACE$ ולכל $L' \subseteq \Sigma^*$ כך ש $L' \in PSPACE$, קיימת טרנספורמציה פולינומית: $f : \Sigma^* \to \Sigma^*$ כך ש $f(w) \in L$ אם ורק אם $w \in L'$.

טרנספורמציה פולינומית, משמעותיה כי ניתן לחשב את $f(w)$ בזמן פולינומי ביחס לאורך של w.

נוסחאות בוליאניות עם כמתים מתקבלות ממשתנים בוליאניים פשוטים על ידי קשרים לוגיים, $\wedge, \vee, \neg$ וכמתי $\exists$ (קיים) ו $\forall$ (לכל).

1. כל משתנה פשוט x (או הקבוע 0,1) הוא QBF. הכניסה של x היא חופשית.
2. אם F_1, F_2 הם QBF אזי $\neg F_1$, $F_1 \wedge F_2$, $F_1 \vee F_2$ הם QBF.

הכניסה של x בנוסחאות הנ"ל היא חופשית אם ורק אם הכניסה היא חופשית ב F_1 או ב F_2.

אם F היא QBF, אזי $\exists xF$ ו $\forall xF$ הן QBF גם כן.
הכניסות של x בנוסחאות הנ"ל הן קשורות.

לכל QBF ללא משתנים חופשיים יש ערך אמת 1 (אמת) או 0 (שקר).
כדי לחשב את ערך האמת של QBF צריך להחליף כל תת נוסחה בצורה $\exists xF$ בנוסחה: $E_0 \vee E_1$ וכל תת נוסחה בצורה $\forall xF$ בנוסחה $E_0 \wedge E_1$ כאשר E_0 ו E_1 מתקבלות מ E ע"י הצבת 0 ו 1 בהתאמה עבור x.

דוגמה: $\forall x\left(\overbrace{\forall x \exists y\left(x \vee y\right)}^{F} \wedge \neg x\right)$

ערך האמת של $\forall x\left(F \wedge \neg x\right)$ שווה לערך האמת של: $\left(F \wedge \neg 0\right) \wedge \left(F \wedge \neg 1\right)$ ששווה ל $F \wedge 0$ שהוא כמובן 0.

משפט: השפה: $L=\{F \mid$ F היא QBF שערך האמת של הוא 1$\}$ היא PSPACE שלמה ביחס לזמן פולינומי.

נוכיח קודם כי $L \in PSPACE$.
ניתן לחשב את ערך האמת של נוסחת QBF על ידי הרקורסיה הבאה:

1. אם הנוסחה לא מכילה משתנים בוליאניים אזי מחשבים את ערך האמת בצורה "הרגילה" - בכל שלב מחפשים את הנוסחאות האטומיות ומחליפים אותן בערך האמת שלהן (למשל $0 \vee 1$ מוחלף ב 1).

2. אם הנוסחה היא בצורה $\exists xF$ או $\forall xF$ אזי מחשבים את ערך האמת של F_0 ואחר כך בשימוש באותו המקום מחשבים את ערך האמת של F_1.
מערך האמת של F_0 ושל F_1 מחשבים במקום קבוע את ערך האמת של $\exists xF$ ושל $\forall xF$.

אם אורך הנוסחה הוא n, אז עומק הרקורסיה הוא לכל היותר n. משום שבכל שלב של הרקורסיה צריך לזכור נוסחה אחת באורך n, המקום הדרוש לביצוע הרקורסיה הוא $O\left(n^2\right)$. לכן $L \in PSPACE$.