פרק י': כריעות ואי כריעות בתחשיב היחסים

10.25 אנו עוסקים כאן בתוכניות בשפת תיכנות כלשהי. כל תוכנית היא, כמובן, מחרוזת של סימנים. ההנחה היא שכל תוכנית מתבצעת עד שהיא נותנת פלט, שהוא מחרוזת ואז היא עוצרת. כשאנו אומרים שביצוע התוכנית נותן פלט כוונתנו בכך היא שהתוכנית כתובה לעשות זאת, אבל יתכן שבמהלך ביצוע התוכנית הביצוע נכנס ללולאה אינסופית ולעולם לא יגיע להוצאת פלט ולסיום. אנו נדון כאן בתוכניות משני סוגים. סוג א'. תוכנית לחישוב מחרוזת ללא כל קלט. דוגמה לתוכנית כזאת היא תוכנית לחישוב π בדיוק של אלף ספרות. בתום ביצועה של תוכנית זאת תופיע על גבי המסך או תודפס במדפסת המחרוזת $3.14\ldots$ ובזה יסתיים ביצוע התוכנית.

סוג ב'. תוכנית המחשבת פונקציה ממחרוזות למחרוזות. זאת תוכנית ϕ הפועלת כמו תוכנית מסוג א' רק שלפני התחלת הביצוע היא מחכה לכך שהמשתמש יכניס קלט ψ שהוא מחרוזת. הפלט שהתוכנית מייצרת תלוי בקלט ψ. אנו מסמנים ב- $\mathrm{output}(\phi,\psi)$ את הפלט המתקבל כאשר התוכנית ϕ פועלת על הקלט ψ. כפי שהזכרנו יתכן שביצועה של התוכנית עבור קלט מסויים ימשיך עד אינסוף מבלי שיווצר פלט כלשהו, ואז $\mathrm{output}(\phi,\psi)$ אינו מוגדר.

ננסה עתה להגיע לסתירה, לא ברור בשלב זה מאלו הנחות, ע"י שנצביע על תוכנית χ מסוג ב' השונה מכל התוכניות מסוג ב' ולכן היא תהיה שונה מעצמה, וזה כמובן סתירה. תהי ϕ תוכנית כלשהי מסוג ב', ואנו צריכים לדאוג לכך ש-χ תהיה שונה מ-ϕ. נעשה זאת ע"י שנכתוב את χ כך שעבור קלט λ מסויים χ תתן פלט שונה מזה ש-ϕ נותנת לקלט זה, כלומר $\mathrm{output}(\chi,\lambda)\neq\mathrm{output}(\phi,\lambda)$. עלינו, אם כן, לבחור לכל תוכנית ϕ מחרוזת λ שעבורה χ תתן תוצאה שונה מזו של ϕ, והבחירה הפשוטה ביותר היא לבחור את התוכנית ϕ עצמה כמחרוזת λ זאת, כשאנו שמים לב שכל תוכנית היא גם מחרוזת. כך נקבע את χ כך ש- $\mathrm{output}(\chi,\phi)\neq\mathrm{output}(\phi,\phi)$. לכן תהי χ התוכנית הבאה. בהינתן קלט ϕ, שהוא מחרוזת כלשהי, התוכנית χ בודקת אם ϕ היא תוכנית מסוג ב'. זאת בדיקה שכל קומפיילר יודע לעשות אותה. אם התשובה היא שלילית אז χ נותנת פלט כלשהו, למשל 0. אם התשובה היא חיובית אז התוכנית χ מבצעת את התוכנית ϕ על הקלט ϕ, רואה מהו הפלט המתקבל ונותנת פלט שונה מזה, למשל, אפשר לקבוע את הפלט כמחרוזת $\mathrm{output}(\phi,\phi)$ בתוספת תו כלשהו בסופה. בכך הצבענו על תוכנית χ מסוג ב' כך שלכל תוכנית ϕ מסוג ב' קיים $\mathrm{output}(\chi,\phi)\neq\mathrm{output}(\phi,\phi)$. מכיוון שזה נכון לכל תוכנית ϕ מסוג ב' זה נכון גם ל-χ עצמה, ולכן $\mathrm{output}(\chi,\chi)\neq\mathrm{output}(\chi,\chi)$, וזאת כמובן סתירה.

האם הצלחנו להוכיח סתירה מתמטית או שישנה טעות בטעון שלנו? כאן באמת נפלה טעות והיא שהנחנו שכאשר התוכנית ϕ מופעלת על הקלט ϕ אז תמיד קיים פלט, ואין לנו כל סיבה להניח זאת, כי קיימת האפשרות שחישוב זה אינו מסתיים. מה קורה כשהחישוב אינו מסתיים? מכיוון שביצוע התוכנית χ מחכה לקבלת הפלט של התוכנית ϕ כדי לתת את הפלט לקלט ϕ הביצוע של χ ימשיך לחכות עד אינסוף ולעולם לא יתן פלט. לכן קיים אי השוויון $\mathrm{output}(\chi,\phi)\neq\mathrm{output}(\phi,\phi)$ רק כאשר החישוב של $\mathrm{output}(\phi,\phi)$ מסתיים ואחרת גם $\mathrm{output}(\phi,\phi)$ וגם $\mathrm{output}(\chi,\phi)$ אינם מוגדרים. מכיוון שאי הש-וויון $\mathrm{output}(\chi,\chi)\neq\mathrm{output}(\chi,\chi)$ אינו יכול להיות נכון לכן ברור שהחישוב של $\mathrm{output}(\chi,\chi)$ לעולם אינו מסתיים, והצלחנו להוכיח עובדה זאת ולא סתירה כלשהי.

ננסה עתה לתקן את הוכחת הסתירה כדי שהיא תהיה נכונה, והמטרה שלנו היא להימנע מכך ש-χ נכנסת למלכודת של חישוב את $\mathrm{output}(\phi,\phi)$ כאשר חישוב זה אינו מסתיים. לשם כך אנו נניח שקבוצת התוכניות מסוג א' שביצוען מסתיים היא כריעה. נקרא לקבוצה זאת **קבוצת העצירה**. כשנאמץ את ההנחה הזאת נצליח לתקן את ההוכחה ולהגיע לסתירה, ואז מה שזה מראה הוא שההנחה אינה נכונה, ולכן קבוצת העצירה אינה כריעה.

לשם כך אנו זקוקים לפעולה הבאה על תוכניות. תהי ϕ תוכנית מסוג ב' ו-ψ מחרוזת כלשהי. אנו רוצים ליצור תוכנית מסוג א' שנסמנה ב-ϕ_ψ הפועלת בדיוק כמו ϕ עם הקלט ψ. לשם כך עלינו לצאת מן התוכנית ϕ ובמקום שלפני ביצוע התוכנית תוכנס למתשתנה x של התוכנית המחרוזת ψ אנו מוסיפים לתחילת התוכנית פקודות הנותנות למשתנה x את הערך ψ. ברור כי מהלך החישוב של ϕ_ψ אחרי השלב של טעינת הערך ψ במשתנה x זהה לזה של מהלך החישוב של ϕ עם הקלט ψ, ואז ששני חישובים אלו

אינם מסתיימים, או ששניהם מסתיימים עם אותה תוצאה. המעבר משתי המחרוזות ϕ ו-ψ למחרוזת ϕ_ψ הוא פשוט ביותר וניתן לביצוע ע"י מחשב.

נחזור עתה להגדרת התוכנית χ מסוג ב', כאשר אנו מניחים שקבוצת התוכניות מסוג א' שביצוען מסתיים היא כריעה. בהינתן קלט ϕ, שהוא מחרוזת כלשהי, התוכנית χ בודקת אם ϕ היא תוכנית מסוג ב'. אם התשובה היא שלילית אז χ נותנת פלט כלשהו, למשל 0. אם התשובה היא חיובית אז התוכנית χ בודקת אם הביצוע של התוכנית ϕ עבור הקלט ϕ מסתיים או לא. בדיקה זאת אינה נעשית ע"י ביצוע התוכנית ϕ, כי אם ביצוע זה לא יסתיים אז גם ביצוע χ לא יסתיים. בדיקה זאת נעשית ע"י יצירת התוכנית ϕ_ϕ, שהיא תוכנית מסוג א', והפעלה עליה של התוכנית הבודקת אם החישוב של תוכנית מסוג א' מסתיים או לא. אם התשובה היא שהחישוב של ϕ_ϕ אינו מסתיים אז ידוע לנו שגם החישוב של ϕ עם הקלט ϕ אינו מסתיים ואז χ תתן ערך כלשהו, למשל 0. אם התשובה היא שהחישוב של ϕ_ϕ מסתיים אז χ מפעילה את ϕ_ϕ ונותנת כתוצאה מחרוזת השונה מן התוצאה של ϕ_ϕ, ולכן מן התוצאה של הפעלת ϕ על הקלט ϕ. כך בנינו את χ שהיא תתן תוצאה לכל מחרוזת ϕ ושלכל תוכנית ϕ מסוג ב' χ תתנהג עם הקלט ϕ באופן שונה מן התוכנית ϕ. כאשר נקח עבור ϕ את χ, נקבל שהחישוב של χ עם הקלט χ מתנהג באופן שונה מן החישוב של χ עם הקלט χ, וזאת היא, כמובן, סתירה. כך הוכחנו שקבוצת העצירה אינה כריעה.

מטרתנו עתה היא להעביר את תופעת אי הכריעות מן התוכניות בשפת מחשב לתחשיב היחסים. לשם כך נעשה מספר צעדים כדי להקל על כך. תחילה נחליט שבמקום לעסוק בחישובים עם מחרוזות נעסוק בחישובים עם מספרים טבעיים. ברור לנו כי בשפה בת מניה אפשר לראות, ע"י התאמת סימני השפה למחרוזות של ביטים, בכל מחרוזת מחרוזת של ביטים, ובכל מחרוזת של ביטים אנו יכולים לראות את המספר הבינרי המתאים. המעברים בשני הכוונים, מן המחרוזות למספרים ומן המספרים למחרוזות, ניתנים לביצוע בנקל ע"י מחשב, כפי שהמחשבים עושים זאת כיום בפועל. כאשר אנו כותבים במחשב קובץ של תווים כלשהם המחשב שומר את הקובץ בזיכרונו כמחרוזת של ביטים, שבכל אחד מהם הוא 0 או 1. כאשר אנו רוצים לראות במסך המחשב את הקובץ המחשב חוזר והופך את מחרוזת הביטים לקובץ של תווים. מחרוזת של ביטים ניתנת לתרגום למספר ע"י הוספת הספרה 1 בתחילתה, ומעבר למספר שזאת ההצגה הבינרית שלו. למשל אם אנו יוצאים ממחרוזת הביטים 0100110 אנו מוסיפים 1 בתחילתה ועוברים למחרוזת 10100110. מחרוזת זאת היא ההצגה הבינרית של המספר 38. המעבר ההפוך גם הוא פשוט. כאשר נתון המספר 38 אנו עוברים להצגתו הבינרית שהיא 10100110, משמיטים את ה-1 שבתחילתה ומקבלים את המחרוזת 0100110. לכן אנו יכולים גם להתייחס לכל תוכנית מחשב, שהיא מחרוזת של תווים, כאל מספר טבעי.

חישובים במספרים יכולים להעשות ע"י כל שפת מחשב עילית, אבל מכיוון שהשפות המקובלות הן עשירות מאוד זה מסובך לתאר את פעולתן בתחשיב היחסים. לכן נשתמש לצרכים שלנו בשפת מחשב פשוטה ביותר שהיא תת שפה של Basic. לשפה זאת ישנה סדרה אינסופית של משתנים $\mathsf{X0}, \mathsf{X1}, \ldots$. פקודות השפה הן הבאות, אשר בהן n ו-k הם מספרים טבעיים כלשהם ו- $k \geq 1$.

א. הוסף 1 ל-$\mathsf{X}n$.

ב. החסר 1 מ-$\mathsf{X}n$.

ג. אם ב-$\mathsf{X}n$ נמצא 0 עבור לביצוע פקודה מספר k בתוכנית.

ד. סיים את ביצוע התוכנית.

תוכנית זאת סדרה ממוספרת, במספרים מ-1 ומעלה, של פקודות. תוכנית ϕ מחשבת פונקציה של m משתנים טבעיים כך. עבור ערכים $x_1, \ldots, x_m$ נתונים טוענים את הערכים הללו במשתנים X1 עד Xm וערכי יתר המשתנים הם 0, ואז מפעילים את התוכנית. פקודות התוכנית מתבצעות בסדר בו הן כתובות, אלא אם כן מבוצעת פקודה מסוג ג' והיא מורה להמשיך במקום אחר בתוכנית. ביצוע התוכנית מסתיים כאשר מתבצעת פקודת הסיום ד', ותוצאת החישוב היא המספר הנמצא אז במשתנה X0. לחישוב ללא קלט לא טוענים דבר במשתנים בתחילת הריצה.

שפת חישוב זאת היא דלה מאוד אבל לא קשה להראות כי היא יכולה לעשות את כל החישובים האפשריים, גם אם לא באופן יעיל. כדי להראות זאת מראים שאפשר לחשב בה חיבור וכפל ואפשר להפעיל בה רקורסיה ופעולות מסוג while.

כעת נדון בשאלה של תאור פעולת התוכנית ע"י השפה $\{0, \mathsf{S}, \dot{+}, \circ\}$ במבנה המספרים הטבעיים $\mathbf{N}^+ = \langle 0, s, +, \cdot\rangle$. בשפה זאת ניתן לטפל לא רק במספרים בודדים אלא גם בסדרות כלשהן של מספרים כאשר מספרים מצפינים סדרות של מספרים. הדרך המקובלת להוכיח שאפשר להצפין סדרות של מספרים ע"י מספרים משתמשת במשפט השאריות הסיני. מכיוון שבהתחלת ריצת המחשב נמצאים מספרים שונים מ-0 רק במספר סופי של משתנים אז מצב זה ממשיך לשרור במשך כל הריצה כי ביצוע של פקודה בודדת יכול לגרום רק למשתנה אחד לקבל ערך שונה מ-0. לכן מצב של המחשב בשלב מסויים של הריצה נתון ע"י זוג שרכיבו הראשון הוא מספר הפקודה העומדת להתבצע, והמספר 0 אם החישוב הסתיים, ורכיבו השני הוא הסדרה הסופית של ערכי המשתנים עד המשתנה האחרון שערכו אינו 0. לפי תאור זה המצב ההתחלתי של המחשב, עם קלטים $x_1, \ldots, x_m$ הוא $\langle 1, \langle 0, x_1, \ldots, x_m\rangle\rangle$ ריצה מסתיימת של המחשב היא סדרת מצבים שהמצב הראשון בה הוא המצב ההתחלתי, כל מצב אחר בה מתקבל מקודמו לפי הפקודה המתבצעת, והמצב האחרון בה הוא מצב מהצורה $\langle 0, \langle y, \ldots\rangle\rangle$ ותוצאת החישוב היא y. למשל, אם מצב מסויים בריצה הוא $\langle k, \langle y_0, y_1, \ldots, y_n, \ldots\rangle\rangle$ והפקודה ה-k בתוכנית היא "הוסף 1 ל-$\mathrm{X}n$" אז המצב הבא הוא $\langle k+1, \langle y_0, y_1, \ldots, y_n+1, \ldots\rangle\rangle$. כך מצב של המחשב הוא זוג של מספר וסדרת מספרים. את סדרת המספרים אפשר לייצג ע"י מספר, למשל את הסדרה $\langle k, l, m\rangle$ אפשר לייצג ע"י המספר $2^{k+1}3^{l+1}5^{m+1}$, ולכן אנו יכולים לראות במצב זוג מספרים. למשל במצב $\langle n, \langle k, l, m\rangle\rangle$ אנו יכולים לראות את זוג המספרים $\langle m, 2^{k+1}3^{l+1}5^{m+1}\rangle$ זוג מספרים $\langle p, q\rangle$ ניתן לייצוג ע"י מספר, למשל ע"י המספר 2^p3^q, ולכן מצב ניתן לייצוג ע"י מספר. ריצה זאת סדרת מצבים, ולכן היא ניתנת לייצוג ע"י סדרת מספרים שגם היא ניתנת לייצוג ע"י מספר. כפי שראינו לעיל, גם כל תוכנית מחשב ניתנת לייצוג ע"י מספר. לאור כל זאת אפשר לכתוב נוסחה $\rho(x, y)$ בשפת $\mathbf{N}^+$ ה"אומרת" ש-y היא ריצה מסתיימת של התוכנית x.

נבהיר מה אנו מתכוונים כאשר אנו אומרים ש-$\rho(x, y)$ "אומרת" ש-y היא ריצה מסתיימת של התוכנית x. בכך אנו מתכוונים שלמספרים k, l כלשהם קיים $\mathbf{N}^+ \models \rho[k, l]$ אםם k הוא מספר של תוכנית ו-l מספר של ריצה מסתיימת של אותה תוכנית. איך אנו אומרים ע"י פסוק ש-k הוא מספר של תוכנית ו-l מספר של ריצה מסתיימת של אותה תוכנית? לשם כך אנו זקוקים לשמות עצם שערכיהם ב-$\mathbf{N}^+$ הם k ו-l. לכל מספר m נסמן ב-$\mathbf{m}$ (אותה האות בגופן עבה) את שם העצם $\mathsf{S}(\mathsf{S}(\ldots\mathsf{S}(0)\ldots)$ עם m סימני S. קל מאוד לראות, באינדוקציה על m כי $\mathbf{N}^+(\mathbf{m}) = m$. לכן לפי משפט ההצבה $\mathbf{N}^+(\rho(bfk, \mathbf{l})) = \mathbf{N}^*(\rho)[k, l]$.
כך k הוא מספר של תוכנית ו-l מספר של ריצה מסתיימת של אותה תוכנית אםם $\mathbf{N}^+ \models \rho(\mathbf{k}, \mathbf{l})$.

10.36 אי כריעות תורת המספרים. קבוצת הפסוקים האמיתיים במבנה המספרים הטבעיים $\mathbf{N}^+$ אינה כריעה.

הוכחה. נניח שקיים אלגוריתם העונה על השאלה אם פסוק נתון אמיתי ב-$\mathbf{N}^+$ ונראה שקיים אלגוריתם העונה על בעיית העצירה, בעוד שידוע לנו שאין אלגוריתם כזה. האלגוריתם לתשובה על שאלת העצירה יהיה כך. בהנתן תוכנית ϕ נעבור למספר k המייצג אותה, ויהי d_k שם העצם שהוגדר לעיל וערכו ב-$\mathbf{N}^+$ הוא k. נתבונן בפסוק $\exists \mathsf{y}\rho(d_k, \mathsf{y})$, ונפעיל עליו את האלגוריתם העונה על השאלה אם הוא אמיתי ב-$\mathbf{N}^+$. אם התשובה היא חיובית זה אומר שקיים מספר l כך ש- $\mathbf{N}^+ \models \rho(d_k, \mathsf{y})[l]$, ולכן $\mathbf{N}^+ \models \rho(d_k, d_l)$. לפי מה שנאמר לעיל על ρ המספר l מייצג ריצה מסתיימת של התוכנית ϕ, ולכן ריצת ϕ מסתיימת. אם התשובה היא שלילית אז הריצה של ϕ אינה מסתיימת, כי אילו היא היתה מסתיימת ו-l היה המספר המייצג את הריצה המסתיימת הזאת היה קיים $\mathbf{N}^+ \models \rho[d_k, d_l]$ ולכן $\mathbf{N}^+ \models \exists \mathsf{y}\rho(d_k, \mathsf{y})$, בניגוד לתשובת האלגוריתם.

הענין המרכזי שלנו כאן הוא בשאלה אם קבוצת הפסוקים האמיתיים לוגית היא כריעה. נלך עתה צעד אחד בכוון זה. נניח שישנה מערכת אקסיומות סופית קטגורית ל-$\mathbf{N}^+$, כלומר קבוצת פסוקים $\chi_1, \ldots, \chi_n$ שהמודלים שלה בשפת $\mathbf{N}^+$ הם בדיוק המבנים האיזומורפיים ל-$\mathbf{N}^+$. כמובן שבמקום $\chi_1, \ldots, \chi_n$ אנו יכולים לקחת את הגימום שלהם χ ואז המודלים של χ בשפת $\mathbf{N}^+$ הם בדיוק המבנים האיזומורפיים ל-$\mathbf{N}^+$. נראה כי בהנחה זאת פסוק ϕ הוא אמיתי ב-$\mathbf{N}^+$ אםם $\models \chi \to \phi$. אם $\models \chi \to \phi$ אז כמובן $\mathbf{N}^+ \models \chi \to \phi$, ומכיוון ש- $\mathbf{N}^+ \models \chi$ לכן $\mathbf{N}^+ \models \phi$. בכוון השני, נניח כי $\mathbf{N}^+ \models \phi$. כדי להוכיח כי $\models \chi \to \phi$ עלינו להראות כי לכל מבנה $\mathcal{A}$ בשפת $\mathbf{N}^*$ אם $\mathcal{A} \models \chi$ אז $\mathcal{A} \models \phi$. אם $\mathcal{A} \models \chi$ אז מכיוון ש-χ אקסיומה קטגורית ל-$\mathbf{N}^+$ $\mathcal{A}$ איזומורפי ל-$\mathbf{N}^+$, ומכיוון ש- $\phi \models \mathbf{N}^+$ קיים גם $\mathcal{A} \models \phi$. לאור מה שראינו אנו יכולים להסיק כי אילו קבוצת הפסוקים האמיתיים לוגית היתה כריעה אז גם קבוצת

הפסוקים האמיתיים במבנה $\mathbf{N}^+$ היתה כריעה, כי כדי לדעת אם $\mathbf{N}^+ \models \phi$ צריך רק לענות על השאלה אם $\models \chi \rightarrow \phi$. הבעייה עם מה שקבלנו היא שאין לנו כל סיבה להניח שקיימת מערכת אקסיומות סופית קטגורית ל-$\mathbf{N}^+$, ובהמשך אפילו נראה שבאמת לא קיימת מערכת אקסיומות כזאת. עם זאת, מה שעשינו מדריך אותנו מה עלינו לעשות כדי להווכח שקבוצת הפסוקים האמיתיים לוגית. מתברר שעלינו לחפש קבוצת פסוקים סופית $\chi_1, \ldots, \chi_n$ שאמנם אינה קטגורית כי יש לה מודלים שאינם איזומורפיים זה לזה אבל מודלים אלו הם מספיק דומים זה לזה כדי שאפשר יהיה לטעון לגביהם משהו דומה למה שטענו על $\boldsymbol{A}$ ו-$\mathbf{N}^+$ לעיל. קבוצת פסוקים כזאת, למרות שאינה קטגורית ל-$\mathbf{N}^+$ צריכה לתאר היטב היבטים חשובים של $\mathbf{N}^+$. לכן נתבונן עתה בפסוקים שהם אמיתיים ב-$\mathbf{N}^+$ והמתארים היבטים כאלו. הפסוקים P1–P7 שנביא עתה נקראים **אקסיומות פיאנו**.

0 אינו עוקב	P1 $\forall \mathsf{x}(0 \not\approx \mathsf{S}(\mathsf{x}))$
פונקצית העוקב חד חד ערכית	P2 $\forall \mathsf{x}\forall \mathsf{y}(\mathsf{S}(\mathsf{x}) \approx \mathsf{S}(\mathsf{y}) \rightarrow \mathsf{x} \approx \mathsf{y})$
סכימת אקסיומת האינדוקציה	P3 $\forall \mathsf{x}_1 \ldots \forall \mathsf{x}_n(\phi(0) \wedge \forall \mathsf{x}(\phi(\mathsf{x}) \rightarrow \phi(\mathsf{S}(\mathsf{x})) \rightarrow \forall \mathsf{x}\phi(\mathsf{x}))$
הגדרה רקורסיבית של החיבור	P4 $\forall \mathsf{x}(\mathsf{x} \dotplus 0 \approx \mathsf{x})$
	P5 $\forall \mathsf{x}\forall \mathsf{y}(\forall \mathsf{x}\forall \mathsf{y}(\mathsf{x} \dotplus \mathsf{S}(\mathsf{x}) \approx \mathsf{S}(\mathsf{x} \dotplus \mathsf{y}))$
הגדרה רקורסיבית של הכפל	P6 $\forall \mathsf{x}(\mathsf{x} \circ 0 \approx 0)$
	P7 $\forall \mathsf{x}\forall \mathsf{y}(\mathsf{x} \circ \mathsf{S}(\mathsf{y}) \approx \mathsf{x} \circ \mathsf{y} \dotplus \mathsf{x})$

נסביר עתה את סכימת אקסיומת האינדוקציה. תחילה נתבונן בה מנקודת ראות תחבירית. כאן אנו מדברים על סכימה כי אקסיומה זאת אינה פסוק בודד אלא קבוצה איסופית של פסוקים, כי הנוסחה ϕ בסכימה זאת "עוברת" על כל הנוסחאות של השפה. למשל, אם ϕ היא הפסוק $\mathsf{S}(\mathsf{x}) \not\approx \mathsf{x}$ אז אקסיומת האינדוקציה המתאימה היא

$$\mathsf{S}(0) \not\approx 0 \wedge (\forall \mathsf{x}(\mathsf{S}(\mathsf{x}) \not\approx \mathsf{x}) \rightarrow \mathsf{S}(\mathsf{S}(\mathsf{x})) \not\approx \mathsf{S}(\mathsf{x}))$$

ואם ϕ היא הפסוק $\mathsf{x} \dotplus \mathsf{x} \approx \mathsf{x} \circ \mathsf{x}$ אז אקסיומת האינדוקציה המתאימה היא

$$0 \dotplus 0 \approx 0 \circ 0 \wedge \forall \mathsf{x}(\mathsf{x} \dotplus \mathsf{x} \approx \mathsf{x} \circ \mathsf{x} \rightarrow \mathsf{S}(\mathsf{x}) \dotplus \mathsf{S}(\mathsf{x}) \approx \mathsf{S}(\mathsf{x}) \circ \mathsf{S}(\mathsf{x})) \rightarrow \forall \mathsf{x}(\mathsf{x} \dotplus \mathsf{x} \approx \mathsf{x} \circ \mathsf{x})$$

מהי המשמעות של $\forall \mathsf{x}_1 \ldots \forall \mathsf{x}_n$ בסכימת אקסיומת האינדוקציה? אם לנוסחה ϕ אין משתנים חופשיים פרט ל-x, כמו בדוגמאות שראינו זה עתה, אז הנוסחה $P3$ המתאימה היא פסוק ואין לנו צורך בכמתים נוספים, כלומר $n = 0$ ב- $\forall \mathsf{x}_1 \ldots \forall \mathsf{x}_n$. לעומת זאת, אם יש ל-$\phi$ משתנים חופשיים בנוסף על x יש צורך לכמת עליהם כדי לקבל פסוק, וב-$P3$ $\mathsf{x}_1, \ldots, \mathsf{x}_n$ מסמנים את המשתנים החופשיים של ϕ השונים מ-x. למשל, אם ϕ היא הנוסחה $\mathsf{x} \approx 0 \vee \mathsf{y} \dotplus \mathsf{x} \not\approx \mathsf{y}$ אז המשתנה החופשי של ϕ הנוסף על x הוא y ולכן פסוק האינדוקציה המתאים הוא

$$\forall \mathsf{y}(0 \approx 0 \vee \mathsf{y} \dotplus 0 \not\approx \mathsf{y} \wedge \forall \mathsf{x}((\mathsf{x} \approx 0 \vee \mathsf{y} \dotplus \mathsf{x} \not\approx \mathsf{y}) \rightarrow (\mathsf{S}(xx) \approx 0 \vee \mathsf{y} \dotplus \mathsf{S}(\mathsf{x}) \not\approx \mathsf{y})) \rightarrow \forall \mathsf{x}(\mathsf{x} \approx 0 \vee \mathsf{y} \dotplus \mathsf{x} \not\approx \mathsf{y}))$$

כעת נתבונן בסכימת אקסיומת האינדוקציה מנודת הראות של משמעותה. מה שהאינדוקציה, במובנה המתמטי המקובל, אומרת על המבנה $\mathbf{N}^+$ הוא שלכל תת קבוצה W של ω אם $0 \in W$, ואם לכל n אם $n \in W$ אז גם $n+1 \in W$ אז לכל n $n \in W$, כלומר $W = \omega$. הבעייה היא שהשפה של תחשיב היחסים מסדר ראשון אינה מאפשרת לנו לדבר על כל הקבוצות החלקיות ל-ω אלא רק על הקבוצות מהצורה $W_{\phi,s} = \{n \in \omega \mid \mathbf{N}^+ \models \phi\left[s\binom{\mathsf{x}}{n}\right]\}$, שהיא קבוצת כל המספרים n המקיימים במבנה $\mathbf{N}^+$ את הנוסחה ϕ עבור הערכים שההשמה s נותנת למשתנים השונים מ-x. המקרה של $P3$ המתאים לנוסחה ϕ מביע את טענת האינדוקציה לכל הקבוצות $W_{\phi,s}$ עם ϕ זאת ועם כל ההשמות s. כך $P3$ היא חלשה במידה משמעותית מן האינדוקציה המתמטית. האינדוקציה המתמטית מדברת על כל הקבוצות W החלקיות ל-ω, וכידוע מספר הקבוצות הללו אינו בן מנייה, ואילו $P3$ מדברת רק על הקבוצות $W_{\phi,s}$ ומספר הקבוצות הללו הוא בן מניה, כי מספר הנוסחאות ϕ הוא בן מניה, ולכל נוסחה ϕ עם n משתנים חופשיים $\mathsf{x}_1, \ldots, \mathsf{x}_n$ פרט ל-x החלק המשמעותי של ההשמה s הוא $\begin{pmatrix} \mathsf{x}_1 & \ldots & \mathsf{x}_n \\ s(1) & \ldots & s(n) \end{pmatrix}$, ומכיוון ש- $s(1), \ldots, s(n) \in \omega$ מספר האפשרויות לחלק זה גם הוא בן מנייה. את ההשלכה של חולשה זאת של $P3$ ביחס לאינדוקציה נראה מאוחר יותר כאשר נדון בתחשיב היחסים מסדר שני.

10.21 משפטי השיכון של המספרים הטבעיים. א. יהי $\mathbf{N}$ המבנה $\langle \omega, 0, s \rangle$, היכן ש-$s$ היא פעולת העוקב,

ותהי Γ קבוצת הפסוקים $P1$ ו-$P2$. לכל מודל $\boldsymbol{A}$ של Γ קיים שיכון יחיד H של מבנה המספרים הטבעיים $\mathbf{N}$ לתוך $\boldsymbol{A}$.

ב. תהי Γ^+ קבוצת הפסוקים P1,P2,P4–P7 לכל מודל $\boldsymbol{A}$ של Γ^+ קיים שיכון יחיד H של מבנה המספרים הטבעיים $\mathbf{N}^+$ לתוך $\boldsymbol{A}$.

ההוכחה נמצאת בסעיף 10.21 בספר.

נתבונן עתה על מה שקורה במודל $\boldsymbol{A}$ של Γ^+. לפי 10.21ב' ישנו שיכון יחיד של מבנה המספרים הטבעיים $\mathbf{N}^+$ לתוך $\boldsymbol{A}$. מכיוון שאנו יכולים לזהות את $\mathbf{N}^+$ עם תמונתו בתוך $\boldsymbol{A}$ אז $\boldsymbol{A}$ הוא מבנה הכולל את המספרים הטבעיים, על פעולות העוקב, החיבור והכפל שלהם, ואולי עוד עצמים נוספים. נקרא לעצמים נוספים אלו בשם **מתחזים**, כי האקסיומות Γ^+ באו מן המספרים הטבעיים והאיברים הנוספים עליהם ב-$\boldsymbol{A}$ מתנהגים במידה מרובה כמו המספרים הטבעיים אבל הם לא מספרים טבעיים. המספרים הטבעיים ב-$\boldsymbol{A}$ הם הערכים של שמות העצם $\mathsf{0}, \mathsf{1}, \mathsf{S(S(0))}, \mathsf{S(S(S(0)))}, \ldots$. נסמן שמות עצם אלו ב- $0, 1, 2, 3, \ldots$ ונקרא להם **סיפרנים**, ואת הסיפרן המתאים למספר טבעי כלשהו אנו מסמנים באות המסמנת את המספר אבל בגופן עבה. כך הספרן המתאים ל-k, שהוא $\mathsf{0}$ עם k סימני S לפניו, נסמן ב-$\mathbf{k}$. נראה שגם ב-$\boldsymbol{A}$ אנו יכולים לבצע את פעולות החשבון במספרים הטבעיים ולקבל אותן תוצאות כמו ב-$\mathbf{N}^+$. למשל, נחשב את $2+3$. אנו מקבלים, לפי P5,

$$3+2 = \mathsf{S(S(S(0)))} \dot{+} \mathsf{S(S(0))} \approx \mathsf{S(S(S(S(0)))} \dot{+} \mathsf{S(0))}$$

ושוב לפי P5 $\approx S(S(S(S(S(0))) \dot{+} 0))$

ולפי P4 $\approx S(S(S(S(S(0))))) = 5$

באותו אופן, לכל שלושה מספרים k,l,m כך ש- $k+l=m$ אפשר להוכיח, באינדוקציה על l כי $\Gamma^+ \models \mathbf{k} \dot{+} \mathbf{l} \approx \mathbf{m}$, ולכן, כמובן, גם $\Gamma^+ \models \mathbf{k} \dot{+} \mathbf{l} \approx \mathbf{m}$. אלו הן עובדות חשבוניות. לעומת זאת, אפשר לראות כי Γ^+ אינה גוררת את הפסוק $\forall \mathsf{x} \forall \mathsf{y}(\mathsf{x} \dot{+} \mathsf{y} \approx \mathsf{y} \dot{+} \mathsf{x})$. פסוק זה אינו מביע עובדה חשבונית אלא עובדה מתמטית כללית כי אינו מדבר על מספרים מסויימים אלא על כל המספרים. כיצד זה אפשרי שהחיבור לא יהיה חילופי במודל $\boldsymbol{A}$ של Γ^+? בחלק של $\boldsymbol{A}$ האיזומורפי לטבעיים החיבור הוא חילופי, כי $H(k)+H(l)=H(k+l)=H(l+k)=H(l)+H(k)$, אולם בחלק של $\boldsymbol{A}$ שהוא מחוץ לתמונת הטבעיים אין שום סיבה שהחיבור יהיה חילופי.

כדי לטפל בריצה של מחשב אנו זקוקים למערכת של חשבונות שהיא חזקה במידת מה מזו של Γ^+. לשם כך נוסיף לשפה את סימן יחס הסדר $\prec$ ונבחר בקבוצת הפסוקים $\Gamma^{+,<}$ שהיא Γ^+ בתוספת הפסוקים הבאים.

<1 $\forall \mathsf{x} \neg(\mathsf{x} \prec \mathsf{0})$

<2 $\forall \mathsf{x} \forall \mathsf{y}(\mathsf{x} \prec \mathsf{S(y)} \leftrightarrow (x \prec \mathsf{y} \vee \mathsf{x} \approx \mathsf{y}))$

<3 $\forall \mathsf{x} \forall \mathsf{y}(\mathsf{x} \prec \mathsf{y} \vee \mathsf{x} \approx \mathsf{y} \vee \mathsf{y} \prec \mathsf{x})$

נסמן ב-γ את הגימום של תשעת הפסוקים של $\Gamma^{+<}$.

נסמן ב- $\mathbf{N}^{+,<}$ את מבנה המספרים הטבעיים $\mathbf{N}^+$ בתוספת יחס הסדר $<$. ראינו לעיל כי אם k הוא מספר של תוכנית ו-l מספר של ריצה מסתיימת של אותה תוכנית אז $\mathbf{N}^+ \models \rho(\mathbf{k},\mathbf{l})$, ובוודאי שגם $\mathbf{N}^{+,<} \models \rho(\mathbf{k},\mathbf{l})$. הבדיקה של עובדה זאת היא בדיקה חשבונית גרידא כי כל מה שהיא דורשת זה פרוק k ו-l לרכיביהם ובדיקה אם רכיבים אלו הם נדרש. לכן בדיקה זאת ניתנת להיעשות בכל מודל של $\Gamma^{+,<}$ ולכן $\Gamma^{+,<} \models \rho(\mathbf{k},\mathbf{l})$. כדי לתת הוכחה מלאה של טענה זאת צריך להכנס הרבה יותר לפרטים, וכאן הובא רק רעיון ההוכחה. מכיוון ש- $\Gamma^{+,<} \models \rho(\mathbf{k},\mathbf{l})$ קיים כמובן גם $\Gamma^{+,<} \models \exists \mathsf{y} \rho(\mathbf{k},\mathsf{y})$. גם, לפי מה שראינו לעיל, אם הריצה של התוכנית שמספרה k אינה נעצרת אז אין מספר l כך ש- $\mathbf{N}^{+,<} \models \rho(\mathbf{k},\mathsf{y})[l]$, ולכן $\mathbf{N}^{+,<} \models \neg \exists \mathsf{y} \rho(\mathbf{k},\mathsf{y})$, ומכיוון ש-$\mathbf{N}^{+,>}$ הוא אחד המודלים של $\Gamma^{+,<}$, לכן לא קיים $\Gamma^{+,<} \models \exists \mathsf{y} \rho(\mathbf{k},\mathsf{y})$. כך ראינו כי k הוא מספר של תוכנית שחישובה מסתיים אסם $\Gamma^{+,<} \models \exists \mathsf{y} \rho(\mathbf{k},\mathsf{y})$, ומכיוון ש-$\gamma$ הוא הגימום של פסוקי $\Gamma^{+,<}$ זה קיים אסם הפסוק $\gamma \to \exists \mathsf{y} \rho(\mathbf{k},\mathsf{y})$ אמיתי לוגית.

11.1.3 משפט צ'רץ' (Church). תהי L שפה של תחשיב היחסים מסדר ראשון המכילה לפחות קבוע אישי אחד, קבוע פעולה חד מקומי אחד, שני קבועי פעולה דו מקומיים וקבוע יחס דו מקומי אחד. קבוצת הפסוקים האמיתיים לוגית ב-L אינה כריעה.

הוכחה. נניח שקבוצת הפסוקים האמיתיים לוגית בשפה L היא כריעה ונוכיח שגם קבוצת העצירה היא

כריעה, בניגוד למה שהוכחנו. ל-L יש את כל הקבועים הדרושים לקבוצת הפסוקים $\Gamma^{+,<}$. בהינתן תוכנית ϕ אנו רוצים לבדוק אם היא נעצרת או לא. אנו עוברים מ-ϕ למספר k שלה, ואז לפסוק $\gamma \rightarrow \exists y\rho(\mathbf{k}, y)$. על הפסוק הזה אנו מפעילים את האלגוריתם הבודק אם פסוק הוא אמיתי לוגית או לא. מכיוון שראינו לעיל כי פסוק זה הוא אמיתי לוגית אםם התוכנית ϕ נעצרת, לכן התשובה לשאלה אם פסוק זה הוא אמיתי לוגית עונה גם על השאלה אם התוכנית ϕ נעצרת.

תכונת אי הכריעות אינה קיימת רק לקבוצת הפסוקים האמיתיים לוגית בשפה הממלאת אחר התנאים של 11.1.3 אלא לכל שפה של תחשיב היחסים המכילה לפחות קבוע יחס או פעולה דו מקומי אחד או שני קבועי פעולה חד מקומיים. אלו הן השפות שהן מספיק עשירות לפיתוח ענפים של המתמטיקה. הוכחת אי הכריעות של קבוצת הפסוקים האמיתיים לוגית בשפות אלו מסתמכת על 11.1.3. לעומת זאת, בשפות דלות, כגון שפה המכילה רק סימני יחס חד מקומיים קבוצת הפסוקים האמיתיים לוגית היא כריעה.