**The 8000th Busy Beaver number:**

מאמרם של סקוט ארונסון ואדם ידידיה עוסק ב"התחמקותו" של המספר ה-8000 בבעיית הבונה העסוק מתאוריית ZF של תורת הקבוצות.  
ננסה להבין קודם כל את התאוריה ומשם נבין איך (ולמה דווקא) המספר ה8000 בבעיה שלנו מתחמק מקיומה.  
בתחילת המאמר, סקוט מביע את הערכתו הרבה לפרויקט מוצלח של אדם ידידיה, סטודנט לדוקטורט בMIT שהצליח לבנות מכונת טיורינג בעלת בעלת סרט אחד ושני תווים בעלת לא פחות מ-7918 מצבים, שהתנהגותה בהרצה על סרט ריק, לא ניתנת להוכחה מהאקסיומות הקלאסיות של תורת הקבוצות.

הצהרה מדויקת לתוצאה העיקרית היא,  
יש לנו מכונת טיורינג בעלת 7918 מצבים בשם Z, כך ש:  
1. Z פועלת תמידית, בהנחה שישנה עקביות של התאוריה הקרדינלית SRP.

2. לא ניתן להוכיח שZ רצה לעולם בZFC, בהנחה שתאוריה זו עקבית.

**רקע:**  
כתוצאה מיידית מתאוריית אי השלמות של גֶדֶל, נובע שישנה תכנית מחשב כלשהיא באורך מסוים, ה"חומקת" מכוחה של המתמטיקה הרגילה להוכיח מה היא עושה, כאשר היא רצה עם כמות זיכרון בלתי מוגבלת.  
לדוגמא, תכנית כזו יכלה למנות את כל ההשלכות האפשריות של אקסיומות ZFC זו אחר זו ולעצור אם תמצא סתירה כלשהיא.  
בהנחה שZFC עקבית, התכנית אמורה לרוץ לנצח, אך מצד שני ZFC **לא יכולה להוכיח** שהתכנית רצה לנצח, כי אם היא הייתה עושה זאת, היא הייתה מוכיחה את העקביות **שלה**, ומפרה בכך את משפט אי השלמות השני.  
  
לצערנו, דיון זה משאיר אותנו בחשיכה לגבי המקום בחלל תוכניות המחשב בו   
"הגרמלין הגֶדֶלי" מרים את ראשו הבלתי ניתן להכרעה.  
תכנית שתחפש אי עקביות כלשהיא בZFC לבטח תהיה יצור די מסובך.. נסביר מדוע.  
יהיה עליה לקודד לא רק את סכמת האקסיומה של ZFC, אלא בנוסף גם את השפה וכללי הסקה מהמעלה הראשונה..  
בשפת תכנות קלאסית (c, java ..), תכנית בסגנון זה עשויה להיות בעלת אלפי שורות/הוראות אם מדובר בשפת מכונה אלמנטרית.  
כנראה מעולם לא נתקלנו ולא ניתקל בתכנית כזו, גם אם היה לנו מחשב מוטרף בגודלו של היקום והיינו מריצים תכנית אחר תכנית ברנדומליות שיטתית של עכבר מעבדה.

מכאן, השאלה החשובה במקומה עומדת, שאלה שבודדים שאלו עד כה בהיסטוריה:  
האם האקסיומות של תורת הקבוצות מספיקות בכדי לנתח לנתח את ההתנהגות של כל תוכנית מחשב שאורכה לכל היותר, נניח, 50 הוראות מכונה? או שיש תוכניות סופר קצרות *שכבר מציגות "התנהגות גֶדֶלית"*?

מדעני המחשב התיאורטיים עשויים להתנגד לכך שזו "רק שאלה של קבועים".   
ובכן בסדר, כן, אבל גם מקור החיים ביקום שלנו - פאזל לא לגמרי לא קשור לנושא אך בהחלט מהותי - הוא גם "רק שאלה של קבועים"!  
ביתר פירוט, אנו יודעים שאפשר בחוקי הפיזיקה הנוכחיים שלנו לבנות מכונה המשכפלת את עצמה: לדוגמא, DNA או RNA והאביזרים הנלווים להם.  
אנו גם יודעים שמולקולות זעירות כמו H2O ו- CO2 אינן משכפלות את עצמן,  
אבל אנחנו לא יודעים **כמה קטנה** יכולה להיות המולקולה הקטנה ביותר המשכפלת את עצמה - וזה נושא שמשפיע אם עלינו לצפות *למציאת חיים כלשהם ביקום אי פעם* או שהתכבדנו להיות רק אנחנו לבד.

מה אומרים בעצם חוקי הפיזיקה או בכלל, חוקי הטבע?

מדעי הטבע הם מעשה ידי אדם, נועדו בשביל לנסות להבין את דפוסי ההתנהגות בעולמנו.

לכן, יש פיתוח מודלים בהם אנחנו לא מגלים חוקים, אלא מנסחים אותם.

נקודת ההתחלה של כל חוק מתחילה בהנחת יסוד שלא תמיד (לפעמים בהכרח) לא תואמת את המציאות, לדוגמא:

העיקרון הקוסמולוגי - קובע כי היקום כולו איזוטרופי (אין בו כיוון ייחודי ולכן יש בו סימטריה) והומוגני, אחיד בכל מקום ברמת הגלקסיות.

אך כיום ידוע שהיקום לא הומוגני כלל ועיקר, אלא כל גלקסיה עומדת בפני עצמה במרחקים עצומים זו מזו ללא אחידות כלשהיא.

אז איך בוחרים מודלים?

מצד אחד אם נהיה נאמנים מדיי למציאות - המודל יהיה מסובך ולא נצליח לבנות עליו כלום, כי לעולם לא נבין את היקום לאשורו (על המוח האנושי אנחנו יודעים מעט מאוד ומדענים תמימי דעים שחלקים נרחבים ממנו היו ויישארו בגדר חידה).

מצד שני, אם המודל יהיה פשטני , הוא לא ישקף את מה שקורה מסביבנו בצורה ראויה וטווח הטעות יהיה גדול מאוד.

**עקומת מאמץ מַעֲוָות** מתייחסת לחומר שלא קיים בטבע, ובאה להראות לנו איך לכל חומר יש גרף שונה שמייצג את הקשר בין המאמץ, הכוח שהופעל עליו, לבין השינוי שיתבצע בו.

יש חומרים שאחרי שהופעל עליהם כוח יחזרו למצב גיאומטרי זהה, כמו קפיץ

לעומת זאת, חרסינה שנפעיל עליה כוח דומה, תוכל להגיע לנקודת כניעה ומשם לנקודת שבר, אַל חָזור.

בחזרה לשאלה שלנו,  
ישנם אנשים מסוימים העשויים להתנגד לכך שמה שאנחנו שואלים עליו כבר נחקר,   
במהלך חצי המאה של המאה לעצב את מכונת טיורינג האוניברסלית הקטנה ביותר   
(היה זה נושא הפרס של סטיבן וולפרם בסך 25,000 דולר בשנת 2007).   
אבל אני רואה את זה כשונה במהותו, מהסיבה הבאה.   
למכונת טיורינג אוניברסלית - כלומר מכונה המדמה כל מכונה אחרת שמתוארת לה על קלטת הקלט שלה (לדוגמא, מכונת-טיורינג אוניברסלית U, מסוגלת "לסמלץ" חישוב של כל מ"ט M על קלט x) - יש את הפריבילגיה לפרוק כמעט את כל המורכבות שלה לפורמט התיאור של מכונת הקלט. אז כן, זה בדיוק מה שכל [המכונות האוניברסליות הזעירות](#המכונות_האוניברסליות_הזעירות) הידועות עושות! אבל לתוכנית שבודקת (נניח) את ההשערה של גולדבאך, או את השערת רימן, או את העקביות של תורת הקבוצות, על סרט ריק בתחילה, אין חירות כזו.   
עבור מכונות כאלה, מספר מצבים באמת עושה רושם של מדד מהותי לסיבוכיות,   
משום שהסיבוכיות הזו לא יכולה "להינעץ" בשום מקום אחר.

Some people might also object that what we’re asking about has already been studied, in the half-century quest to design the smallest universal Turing machine (the subject of Stephen Wolfram’s $25,000 prize in 2007, to which I responded with my own $25.00 prize). But I see that as fundamentally different, for the following reason. A universal Turing machine—that is, a machine that simulates any other machine that’s described to it on its input tape—has the privilege of offloading almost all of its complexity onto the description format for the input machine. So indeed, that’s exactly what all known tiny universal machines do! But a program that checks (say) Goldbach’s Conjecture, or the Riemann Hypothesis, or the consistency of set theory, on an initially blank tape, has no such liberty. For such machines, the number of states really does seem like an intrinsic measure of complexity, because the complexity can’t be shoehorned anywhere else.

מושגים:

מכונות אוניברסליות זעירות:  
כשאלן טיורינג העלה את הרעיון של מכונה אוניברסלית, הוא חשב על מודל המחשוב הפשוט ביותר שיהיה חזק דיו בכדי לחשב את כל הפונקציות האפשריות שניתנות לחישוב.  
היה זה קלוד שאנון שבשנת 1956 העלה לראשונה באופן מפורש את שאלת מציאת מכונת טיורינג האוניברסלית הקטנה ביותר האפשרית.  
הוא הראה ששני תווים (בד"כ 0 ו-1) מספיקים כל עוד משתמשים במצבים מספיקים   
(או להפך), ותמיד ניתן להחליף מצבים בתווים הללו.  
 הוא גם הראה שאף לא מכונת טיורינג אוניברסלית אחת של מצב בודד יכולה להתקיים.