

אוגדן בעיות מפורסמות במדעי המחשב

האם $N=NP$ או איך לארוז תרמיל ולהרוויח מליון דולר?

כתב שמוליק שוורץ

תיאור הבעיה

הבעיה המתוארת בפרק זה עוסקת במיליון דולר המונחים על השולחן במכון המתמטי "קליי"¹. זוהי אחת מ-7 בעיות בלתי פתירות שהקהילה המתמטית שמה לעצמה מטרה לפתור אותן בשנים הקרובות.

אז מה נותר לכם לעשות, קודם כל ללמוד קצת על הרקע של הבעיה ואז לנסות למצוא אלגוריתם הפותר אותה.

יתכן ואתם תוהים מדוע דווקא אתם תצליחו לפתור את הבעיה, אם היא מספיק קשה כדי להציע עליה מליון דולר, התשובה נמצאת בסיפורו של ווייל הנטינג².

"אני מסתכל בפסנתר ואני רואה קלידים לבנים ושחורים, שלוש דוושות ארגז מעץ. בטהובן,

מוצארט הסתכלו בו והוא נראה להם ברור מאליו... הם ראו פסנתר ויכלו לנגן. אני לא יכול

לנגן ולא יוכל לצייר אבל אם אני פוגש חומר כזה אני פשוט מנגן".

אם לא ראיתם את הסרט – זה הזמן, בכל מקרה הרעיון הוא שיתכן ואחד מכם יוכל פשוט לנגן את התשובה...

לפני שנגדיר את הבעיה נציג שאלה שתעזור לנו להבין במה מדובר.



שאלת אריזת התרמילים לטיול השנתי (bin packing):

נתונים N בקבוקי מים בעלי קיבולות שונות הקטנות מ-10 ליטר.

ברשותי מספר לא מוגבל של תרמילים המסוגלים להחזיק משקל של עד 10 ק"ג (כלומר 10 ליטר).

מהו המספר המינימאלי של תרמילים הדרוש לאריזת N הבקבוקים הללו. (לא העביר מים מבקבוק לבקבוק)

תיאור גרפי:



בדוגמה שלפנינו (1,5,4,1,9) נדרשים שני תרמילים.

אני משאיר לכם להבין מהו הסידור הרצוי...

¹ Clay Mathematics Institute of Cambridge, Massachusetts (CMI), לקוח מ: <http://www.claymath.org/millennium>

² במקור השם הוא (good will hunting) וניתן למצוא את הטקסט בכתובת: http://www.geocities.com/movie_starzz/BenandMatt/goodwillhunting.txt

יש למצוא אלגוריתם כללי הפותר את בעיית הסידור המינימלי ולהגדיר את התלות שלו ב- N הבקבוקים, כלומר בהינתן N בקבוקים בגדלים שונים, כמה תרמילים נצטרך.

אז כיצד ניתן לבדוק את זה? האלגוריתם באופן כללי הוא די פשוט, יש לבדוק את כל הסידורים האפשריים ב- K תרמילים ולקבוע האם ניתן לסדר את הבקבוקים ב- K תרמילים. שאלה מסוג זה נקראת שאלת הכרעה. התשובה היא כן או לא. אם כן אז הסידור ניתן מייד. בפרק פתרון הבעיה תראו כיצד זה מתבצע, השאלה היא כמה זמן זה ייקח? או במילים אחרות מהי הסיבוכיות.

הסיבוכיות במקרה זה נמדדת במונחים שדורשים הסבר. נכיר תחילה את המונחים:

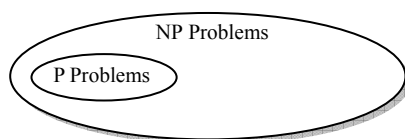
1. סיבוכיות ברמה P – סיבוכיות זו משמעותה שזמן הפיתרון הינו פולינומיאלי, כלומר הסיבוכיות היא מסדר גודל של $O(n^k)$. זוהי סיבוכיות זמן סבירה לפתרון בעיות וישנו מספר רב של בעיות השייכות לקבוצה זו, למשל סוגי המיון השונים הנלמדים בשיעורי המחשב. (האות P מייצגת את המילה Polynomial).

2. סיבוכיות ברמה NP – סיבוכיות זו משמעותה שזמן הפיתרון אינו פולינומיאלי כלומר הסיבוכיות גדולה מ- $O(n^k)$ אבל אם יוצע פיתרון ניתן להכריע האם הוא נכון או לא בזמן פולינומיאלי. למשל בבעיית הבקבוקים, בהינתן סידור שנטען לגביו שהוא מסדר את הבקבוקים ב- K תרמילים, ניתן לבדוק בזמן פולינומיאלי (במקרה זה $O(n^1)$) האם הטענה נכונה. האות N שבסיבוכיות NP מייצגת מושג קצת מורכב יותר, המונח הוא:

Non-Deterministic והמשמעות היא שאם יש לנו מחשב שיכול להריץ את הבדיקות לכל ה- K ים האפשריים במקביל, אז היינו יכולים למצוא את הפיתרון בזמן פולינומיאלי. הבעיה היא שאין מחשב כזה... אבל תיאורטית ניתן לדון בכך.

ניתן להגדיר הגדרה שקולה בעזרת מכונות טיורינג³ לא דטרמיניסטית.

באופן גרפי ניתן לסכם זאת כך:



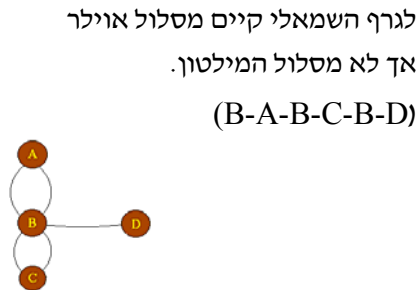
הפתרון שהוצע לפיו נדע מהו מספר התרמילים המינימלי הדרוש על ידי בדיקת כל הסידורים האפשריים נקרא אלגוריתם "ממצה" (exhaustive) הבעיה בשיטה זו היא הסיבוכיות, ל- n איברים יש $n!$ סידורים אפשריים. מדובר בסיבוכיות מסדר גודל אקספוננציאלי שאינו ניתן לפתרון כללי בזמן סביר.

³ הרחבה על אלן טיורינג בכתבה שפורסמה בעיתון הבטים ומשם למקורות נוספים: <http://cse.proj.ac.il/hebetim/5/turing.htm>

לפני שנחזור לשאלת מליון הדולר נציג דוגמא נוספת שתעזור לנו להבין את נושא מחלקות הסיבוכיות.

לשאלה שנציג שני חלקים :

- א. בהינתן גרף, האם קיים מסלול בגרף אשר מבקר בכל קשת בדיוק פעם אחת (מסלול אוילר)?
- ב. בהינתן גרף, האם קיים מסלול בגרף אשר מבקר בכל קודקוד בדיוק פעם אחת (מסלול המילטון)?



שאלה זו מוכרת גם כבעיית הגשרים של קניגסבורג⁴.

שימו לב שאין צורך למצוא את המסלול אלא רק להכריע האם קיים מסלול כזה. גם כאן ניתן להשתמש באלגוריתם של בדיקת כל המסלולים האפשריים בעזרת חיפוש ממצה גם כאן הסיבוכיות ל n איברים היא $O(n!)$.

מסתבר שלהכרעה לגבי הימצאותו של מסלול אוילר אין צורך בחיפוש ממצה ויש אלגוריתם חליפי בסיבוכיות פולינומיאלית, אולם לגבי הימצאותו של מסלול המילטון בגרף, לא ידוע על אלגוריתם חליפי כזה.

הגדרת הבעיה הכללית:

השאלות שהוצגו כאן מייצגות את הבעיה הכללית יותר, והיא בעת מליון הדולר. הבעיה עוסקת בקשר בין שתי קבוצות, קבוצה אחת נקראת קבוצת P והיא מכילה את כל השאלות שפתרוןן הוא בסיבוכיות פולינומיאלית, הקבוצה שנייה נקראת NP והיא מכילה את כל הבעיות שלא ידוע כיצד לפתור אותן בסיבוכיות פולינומיאלית למשל בעית התרמילים (bin packing) אבל ידוע שאם היה לנו מחשב שיכול להריץ את הבדיקות לכל ה K -ים האפשריים במקביל, אז היינו יכולים למצוא את הפיתרון בזמן פולינומיאלי. שתי הקבוצות הללו נראות שונות מהותית זו מזו אבל אף אחד לא הצליח להוכיח שהן באמת שונות. אף אחד גם לא הצליח להוכיח שהן דומות כך שבעצם הקשר ביניהן לא ברור.

אם תצליחו להוכיח ש $P=NP$ (למשל להוכיח ששאלת התרמילים שייכת ל P), או ש $P \neq NP$ (למשל להוכיח שאין אפשרות לפתור את בעית התרמילים בזמן פולינומיאלי, ושימו לב – זה שעד עכשיו לא הצליחו אינה הוכחה!!!) לא רק שתקבלו את סכום הכסף העגול שהוזכר, סביר להניח שתקבלו גם דוקטורט כבונוס.

כדי להקל עליכם בדרככם עליכם להכיר תת קבוצה חשובה בתוך קבוצת ה NP , תת קבוצה זו היא בעצם קבוצת הבעיות הקשות ביותר שנמצאות בתוך NP . הבעיות בתת קבוצה זו נקראות NP -שלמות (Complete).

⁴ התמונה מהכתובת <http://www2.eitan.ac.il/ds/more/np.asp> בה תמצאו הרחבה על מסלול אוילר והסבר קצר נוסף למחלקות ה NP .

קבוצת הבעיות NP-שלמות מכילה מספר רב של בעיות הדומות לשאלת האריזה בתרמילים ולמסלול המילטון.

ואולי אתם תוכיחו ש :

גרפית מדובר כרגע ב :



המונח הוגדר בנפרד על ידי Cook ו Levin באמצע שנות ה 70 של המאה הקודמת. קוק הגדיר בעיה האופינית לקבוצה NP - שלמות אשר אם נפתור אותה בזמן פולינומיאלי אז נוכל בעזרת אלגוריתם שסיבוכיותו היא לכל היותר פולינומיאלית לפתור כל בעיה אחרת ב NP. נסכם : אם תמצאו פיתרון פולינומיאלי למציאת המספר המינימלי של תרמילים אריזה או למציאת מסלול המילטון או לכל בעיה אחרת ב NP-Complete הוכחתם ש $P=NP$.

אבל מדוע מציאת אלגוריתם זה שווה מליון דולר?

אלגוריתם כזה אינו בעיה תיאורטית בלבד, הוא יפתור בעיות אמיתיות בניהול חומרה, בתעשיית האלקטרוניקה, בהצפנה ובתחומים נוספים, העובדה שקיים מגוון של בעיות השקולות זו לזו תגרום לסדרת פתרונות שתתפשט כמו אבני דומינו ותביא למהפכה ממש ביכולת החישוב. כדי להבין את עוצמת המהפכה – שימו לב מה המשמעות של פתרון ממצה למקרה של $n=20$, עלינו לחשב $20!$ כלומר : $1*2*.....*18*19*20$.

התשובה היא : $2,432,902,008,176,640,000$, מה משמעותו של מספר תמים זה? נניח שהמחשב שלכם מסוגל לבצע $1,000,000,000$ פעולות בשנייה כלומר ידרשו למחשב בודד כ $2,432,902,008$ שניות לבצע מספר כזה של פעולות.

בשעה יש 3600 שניות כלומר ידרשו לו כ 675806 שעות, שהן כ 28158 ימים שהן כ 77 שנים.⁵

פתרון הבעיה (או ניסיונות לפתרון)

אם היה לי פתרון לבעיה אז ברור שלא הייתי כותב אותו כאן לפני שהייתי מקבל את מיליון הדולר. אבל לפניכם ניתוח שיקרב אתכם לצורת החשיבה שתוביל לפיתרון (אולי).

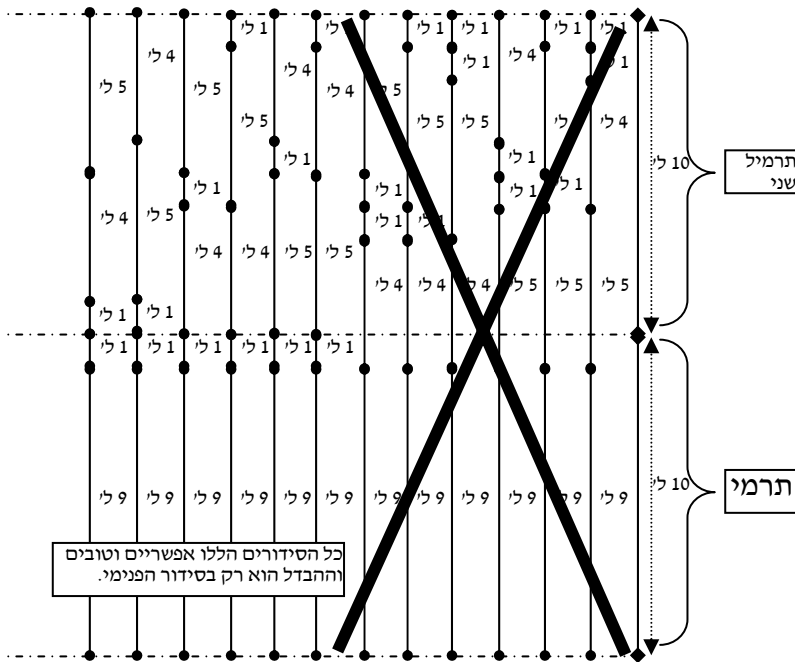
בעיית התרמילים בניסוח רשמי :

"נתונה לנו קבוצה של n עצמים כאשר הגודל של כל עצם הוא גודל חיובי הקטן מ 1 . ברצוננו לארוז את כל העצמים בתוך המספר המינימלי של תיבות בגודל יחידה. כאשר כל תיבה יכולה להכיל כל תת-קבוצה של עצמים שגודלה הכולל אינו עולה על 1 . מצא אלגוריתם שסיבוכיותו היא פולינומיאלית המוצא את סידור מינימלי זה".

⁵ טבלה מסכמת תוכלו למצוא בספר "אלגוריתמיקה – יסודות מדעי המחשב" מאת דוד הראל, האוניברסיטה הפתוחה בפרק הדין בבעיות אלו.

סיבוכיות הפיתרון בשיטת בדיקת כל האפשרויות אינה פולינומיאלית. בניסוח הרשמי גודל תיבות האריזה הוא 1 אנו נישאר עם התרמילים בקיבולת של 10 מטעמי נוחות, הניתוח שקול לחלוטין. נתחיל בפתרון ממצה, לפניכם **חלק** מהסידורים האפשריים של בעיית הבקבוקים שהוצגה (1,5,4,1,9):

השיטה שתוצג כאן היא לסדר את כל הסידורים האפשריים של הבקבוקים ולבדוק איזה מהם ניתן לחלוקה שלמה לתרמילים של 10 ליטר. בקבוק ייוצג כקו בעל אורך מתאים והקווים המקווקווים יצינו נסיונות חלוקה לתרמילים.



שימו לב – אנחנו אמורים לבדוק את כל הסידורים האפשריים...

אפשר לראות שקיימת סדרה של פתרונות שגויים ושיש יותר מסידור אחד שמקיים את התנאי.

תרגיל:

נתונים בקבוקים בגדלים: 3,6,2,1,5,7,2,4,1,9. נסו לבדוק מה המספר המינימלי של תרמילים הנדרש, חישבו כיצד להוכיח זאת.

למעשה, לא ביקשו מאיתנו לפתור אלא לבנות אלגוריתם. המחשב יעשה את העבודה בשבילנו!!!

אין טעם להסתכל על המספרים ולהתחיל לסדר בשיטה של ניסוי ותהייה/טעייה שכן כאשר מספר הבקבוקים גדול כבר ראינו ששיטה זו אינה ישימה. נסו לתכנן אלגוריתם – שיעבוד תמיד....

לפניכם הצעה לאלגוריתם פתרון:

עבור כל בקבוק בצע: "אם יש מקום בתרמיל פתוח – תוסיף, אם אין אז תפתח תרמיל חדש".

אם נקרא את הנתונים מימין לשמאל, אזי לפי אלגוריתם זה יתבצעו הפעולות הבאות:

9 – אין לו מקום באף תרמיל קודם (כי אין אף תרמיל פתוח) לכן נפתח תרמיל חדש.

1 – יש לו מקום בתרמיל של ה 9, נוסיף אותו.

4 – אין לו מקום באף תרמיל קודם לכן נפתח תרמיל חדש.

2 – יש לו מקום בתרמיל של ה 4, נוסיף אותו.

7 – אין לו מקום באף תרמיל קודם לכן נפתח תרמיל חדש.

5 – אין לו מקום באף תרמיל קודם לכן נפתח תרמיל חדש.

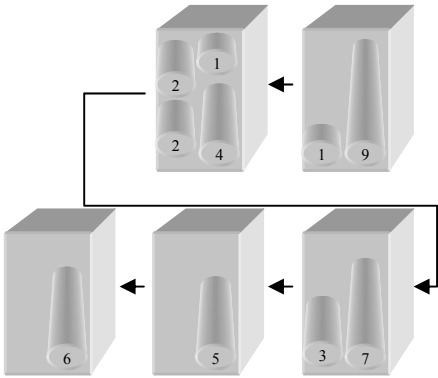
1 – יש לו מקום בתרמיל של ה 4 וה 2, נוסיף אותו.

2 – יש לו מקום בתרמיל של ה 4, ה 2 וה 1, נוסיף אותו.

6 – אין לו מקום באף באף תרמיל קודם לכן נפתח תרמיל חדש.

3 – יש לו מקום בתרמיל של ה 7, נוסיף אותו.

אין יותר בקבוקים - סיימו.



אלגוריתם זה נקרא גם First Fit או בקיצור FF⁶. שמו מתמצת את מהותו – מציבים את הבקבוק הבא בתור במקום הפנוי הראשון האפשרי לו. אולם האם אלגוריתם זה נותן לנו את התשובה לשאלת מינימום התרמילים? קל לראות שנדרשים רק 4 תרמילים ולא חמישה כדי לארוז את הבקבוקים, אולם האלגוריתם לא מצא זאת.

שימו לב שיש כאן תלות בסדר ה"הגעה" של הנתונים.

נביא 3 תרגילים להמחשת הרעיון.

תרגיל: תוכלו לנסות כתרמיל שיטה לשיפור של האלגוריתם, נניח שאתם לא מחלקים מיד לתרמילים אלא ממתינים לקליטת כל הנתונים, מסדרים אותם מגדול לקטן ורק אז מתחילים למלא את התרמילים. האם לסדרה הנתונה האלגוריתם ימצא את הסידור המינימלי? האם הוא ימצא את הסידור המינימלי לכל סדרה?⁷ החלטה זו אגב, אינה ישימה בחיי היום יום. למשל אם האריזות מייצגות משאיות אשר יוצאות לדרכן ברגע שהן מלאות, אין אפשרות להמתין עד אשר יהיו בידנו כל הנתונים ורק אז להתחיל לסדר שכן תחזיקו צי של משאיות בהמתנה. יתכן ואין ברשותכם מספיק משאיות להעביר את כל הבקבוקים ועליכם להמתין עד שמשאית תחזור כדי להעמיסה שוב, ואז ממש לא כדאי להשהות את כל המשאיות עד שתסיימו למיין.

תרגיל: תאר כיצד יפעל האלגוריתם הממצה לבדיקת המספר המינימלי של תרמילים הנחוץ לסדרת הבקבוקים 3,6,2,1,5,7,2,4,1,9? (הסידור הוא - (1,9) (3,7) (4,6) (1,2,2,5))

תרגיל: נוכל גם להקטין את הסיבוכיות אם במקום להתחיל מ 1 נחשב תחילה את מספר התרמילים המינימלי שבהם ניתן היה לסדר את הבקבוקים אם הם היו כולם בעלי גודל של 1 ליטר. כלומר במקרה שלנו הנפח הכולל של הבקבוקים הוא 40 ליטר ולכן ידרשו לא פחות מ 4 תרמילים כדי לארוז את כל הבקבוקים, האם תוכלו למצוא סדרת בקבוקים שהנפח הכולל שלהם הוא 40 אבל לא ניתן לסדר אותם ב 4 תרמילים?⁸

⁶ תוכל לקרוא על אלגוריתמים נוספים מסוג זה בכתובת:

⁷ נמייין את הנתונים, כלומר הסדר מימין לשמאל הוא מגדול לקטן והוא: 1,1,2,2,3,4,5,6,7,9. האלגוריתם מצליח, ידרוש 4 תרמילים בלבד!! אולם אלגוריתם זה לא ימצא **תמיז** את הסידור המינימלי. למשל כאשר הנתונים הם: 1,2,2,2,3,4,5,6,6,9 האלגוריתם יצור את הסידור הבא: (1+9)(4+6)(3+6)(2+2+5)(2) ולא את הסידור המינימלי (1+9)(4+6)(2+2+6)(2+3+5)
⁸ כמובן שלא ניתן להסיק מכך דבר שכן לסדרת הבקבוקים 4,6,6,6,6,6 שגם נפחה הכולל הוא 40 ידרשו לפחות 6 תרמילים

על Cook (בהרחבה) ו Levin (בקצרה).

סטיבן קוק (Steven Cook) נולד בבאפלו, ניו-יורק בשנת 1939. בשנת 1957 החל בלימודיו באוניברסיטת מישיגן ובמסגרת לימודיו התמחה במתמטיקה. הוא סיים את התואר הראשון בשנת 1961. את התואר השני סיים באוניברסיטת הארווארד בשנת 1962 והמשיך במסלול לדוקטורט אותו סיים בשנת 1966. קוק עסק בתחום שנקרא "תחשיב פסוקים" (propositional calculus) העוסק בקביעת אמת לוגית של פסוק. פסוק שעבורו קיימת השמה שהופכת אותו לפסוק אמת נקרא "ספיק".

למשל : הפסוק $(x \text{ AND } y)$ הוא פסוק אמת אם ערכי שני המשתנים true, ולכן הפסוק ספיק.

הפסוק $(x \text{ AND NOT } y)$ הוא פסוק אמת לכל הצבה של המשתנים, ולכן בודאי שהוא ספיק.

לעומתם הפסוק $(x \text{ AND NOT } y \text{ AND NOT } x)$ אינו ספיק שכן לא קיימת הצבה בה הוא יהיה true.

קוק חקר אלגוריתמים שמטרתם לקבוע האם פסוק הוא ספיק, רבים מהם דרשו מספר פעולות מסדר גודל 2^n . עובדה שגרמה לקוק להרחיב את נושא מחקרו מבדיקת אלגוריתמים לשאלת הספיקות לשאלה רחבה יותר – "אילו בעיות אנחנו בכלל מסוגלים לפתור".

לאחר שסיים את לימודי הדוקטורט שהה קוק בברקלי, קליפורניה וב 1970 הצטרף לצוות אוניברסיטת טורונטו. בהרצאה שנתן ב 1971 הוא דן קבוצת שאלות מיוחדת, שאלות שאנו לא יכולים לפתור בזמן פולינומיאלי אך בהינתן "מועמד" להיות הפיתרון נוכל לבדוק אותו בזמן פולינומיאלי. הוא הציג רעיון כיצד למצוא את המועמד המתאים, צריך פשוט לנחש.

הניחוש כידוע אינו דבר מוחלט (deterministic) ולכן הוא קרא לקבוצה זו Nondeterministic Polynomial כלומר בעיות שפתרון משולב, תחילה ניחוש שהוא אינו דטרמיניסטי ואז אימות שהוא פולינומיאלי.

קוק הראה שבעיית הספיקות היא אחת מבין הבעיות הקשות בקבוצה ויתרה מכך – אם ימצא בעתיד פתרון פולינומיאלי לבעיית הספיקות אזי ניתן יהיה לפתור בצורה דומה כל בעיה ב NP. בכך הגדיר למעשה את הבעיה הראשונה בקבוצה אשר אליה הצטרפו בהמשך בעיות רבות נוספות. קוק קיבל את פרס טיורינג ב 1982 ואת פרס ה"תרומה להוראת מדעי המחשב" בשנים 1989 ו-1995.

לאוניד לוי (Leonid Levin) נולד בדנפרופטרובסק שבאוקראינה בשנת 1948. בשנת 1971 הוא הגיש את התזה שלו לדוקטורט שעסקה בסיבוכיות קולמוגורוב⁹. למרות שוועדת ההערכה שלו אישרה את התזה הוא לא קיבל את התואר הנכסף. הסיבה הייתה פוליטית – לוי פשוט לא היה חבר במפלגה. בהמשך כתב מאמר בו הראה שקילות בין מספר בעיות ביניהן בעיית הספיקות ומבלי להיות מודע לעבודתו של קוק הוא קרא לבעיותיו בעיות כלליות (universal). בשנת 1978 הוא היגר לארצות הברית וכיום חבר בסגל אוניברסיטת בוסטון¹⁰.

⁹ תוכל למצוא חומר עיון נוסף על האיש ב: <http://www-history.mcs.st-andrews.ac.uk/history/Mathematicians/Kolmogorov.html>

¹⁰ מידע נוסף על האדם ניתן למצוא בדף הבית שלו בכתובת : <http://www.cs.bu.edu/fac/lnd>

מקורות

1. Denis Shasha and Cathy Lazere. Out of Their Minds: The Lives and Discoveries of 15 Great Computer Scientists, NY: Copernicus, 1995, pp. 139-156.
2. <http://www.claymath.org/millennium>
3. http://www.geocities.com/movie_starzz/BenandMatt/goodwillhunting.txt

האתר של קוק

<http://www.cs.toronto.edu/~sacook/>