אלגוריתמים

סיכם: נריה אור.
עוד סיכומים ניתן למצוא ב http://www.cs.huji.ac.il/~neryao/

ע"פ הרצאות של פרופ' נתי ליניאל וצוות המתרגלים. אין המרצים קשורים לסיכום זה בשום אופן.

גירסא 1.0 - סוכמו כל ההרצאות (ותרגול אחד...).

אין אחריות לתוכן הסיכום ולדיוקו - ייתכנו טעויות. השימוש על אחריות הקורא בלבד.

תוכן עניינים

1 אלגוריתמים חמדניים . . . 4
1.1 בעיות תזמון (*scheduling*) . . . 4
1.1.1 בעיה 1#: תזמון משימות . . . 4
1.1.2 בעיה 2#: ארגון חדרים לפעילויות . . . 5
1.1.3 בעיה 3#: ניהול זיכרון מטמון . . . 5
1.2 קודי Huffman (מהתרגול) . . . 7
1.2.1 הבעיה . . . 7
1.2.2 עצי קידוד, והאלגוריתם של Huffman . . . 8
1.2.3 אי־שויון Kraft . . . 11
1.3 עץ פורש מינימלי . . . 11
1.3.1 הבעיה . . . 11
1.4 ההקשר הרחב יותר ־ מטרואידים . . . 11
1.4.1 דוגמא: בעיית הזיווג הממושקל המקסימלי . . . 12
1.4.2 האלגוריתם החמדן המוכלל (שלא תמיד עובד...) . . . 12
1.4.3 מטרואידים . . . 13
1.4.4 דוגמא: על וקטורים, קבוצות בת"ל ומטרואידים . . . 14
1.4.5 דוגמא נוספת: על יערות ומטרואידים . . . 14
1.5 סיכום מטרואידים . . . 15
1.5.1 בסיס, מעגל, חתך . . . 15
2 תכנון דינאמי (Dynamic Programming) . . . 15
2.1 הכפלה יעילה של סדרת מטריצות . . . 15
2.1.1 מוטיבציה \ הקדמה . . . 15
2.2 בעיה בביו־אינפורמטיקה . . . 16
2.2.1 "המבנה השניוני של מולקולת RNA" ־ הקדמה . . . 16
2.2.2 ניסוח מפורש של הבעיה . . . 17
2.3 התאמת זמן דינאמית *Dynamic time warping* . . . 18
2.3.1 הבעיה . . . 18
2.4 חישוב מרחקים בין רצפי חלבונים . . . 18
3 זרימה ברשת . . . 19
3.0.1 בעיית הזרימה ברשת (Network flow problem) . . . 19
3.0.2 טיפה על בעיות אופטימיזציה . . . 20
3.0.3 עוד על זרימה . . . 21
3.0.4 הקדמה לאלגוריתם . . . 21
3.0.5 אלגוריתם Ford-Fulkerson (סכימה) . . . 22
3.0.6 טענות והוכחות . . . 23
3.0.7 מסקנות: . . . 25
3.0.8 למה $O(mc)$ זה לא פולינומי?????? . . . 26
3.1 שימושים של $MFMC$ . . . 26
3.1.1 קצת על זיווגים (הקדמה למשפט Hall) . . . 26
3.1.2 משפט החתונה של Hall . . . 27
3.1.3 קצת על תכנון ליניארי . . . 28
3.2 מימושים של Ford-Fulkerson . . . 28
3.2.1 אלגוריתם מדורג לבעיית הזרימה . . . 28
3.2.2 אלגוריתם Edmonds-Karp לזרימה ברשת . . . 30

4 קירובים לבעיות קשות 32
4.0.3 הקדמה 32
4.0.4 בעיית הכיסוי הקודקודי ($vc =$ vertex cover) 32
4.0.5 בעיית הכיסוי הקבוצתי (set-cover problem) 33
4.0.6 בעיית הסוכן הנוסע (Travelling Salesman Problem) 34
4.1 שימוש בהסתברות למציאת פתרונות מקורבים לבעיות קשות 36
4.1.1 הקדמה - רדוקציה 36
4.1.2 3 דוגמאות לבעיות NP-קשות שניתנות לקירוב ע"י שימוש בהסתברות: 36
4.1.3 אלג' הסתברותי לקירוב בעיית $max3SAT$ 36
5 טרנספורם פורייה 39
5.0.4 כפל של מטריצה בוקטור 39
5.0.5 טעימה מיסודות התחום של עיבוד אותות 39
5.0.6 פולינומים ופעולות בפולינומים 40
5.0.7 DFT 41
5.0.8 עוד על טיפול בפולינומים 43
5.0.9 קונבולוציה 44
5.0.10 FFT 45
5.0.11 שיעור החזרה מה-7.1.2010: עוד על FFT. 46
6 אלגוריתמים בתורת המספרים ובקריפטוגרפיה 48
6.0.12 הקדמה 48
6.0.13 gcd, lcm 48
6.1 שדות, חבורות (Fields, Groups) 49
6.1.1 החבורה $\mathbb{Z}_n^*$ 50
6.1.2 תת-חבורות 53
6.2 RSA ומציאת ראשוניים 55
6.2.1 כמה מילים על מספרים ראשוניים 55
6.2.2 משפט המספרים הראשוניים (PNT) 55
6.2.3 RSA 55
6.2.4 תזכורת קצרה לגבי מה שעשינו עד כה (ועוד קצת דברים חדשים): 56
6.2.5 אלגוריתם $Miller-Rabin$, הקדמה 57
6.2.6 מבחן הראשוניות של $Miller-Rabin$ 58
7 אלגברה ליניארית 60
7.0.7 אלגוריתם $Strassen$ להכפלת מטריצות 60
7.0.8 על פתרון מערכות משוואות ליניאריות 61
7.0.9 קירוב בריבועים פחותים 63
7.0.10 הנורמה האופרטורית 64
7.0.11 ערכים עצמיים לעומת ערכים סינגולריים 65
8 נספח: נוסחאות \ דברים שכדאי לזכור למבחן 68
8.0.12 משפט השאריות הסיני 68
8.0.13 פירוק LUP 69
8.0.14 קונבולוציה 69

1 אלגוריתמים חמדניים

בפרק זה נדון באלגוריתמים חמדניים, שהם בעיקרון אלגוריתמים שבכל רגע נתון בוחרים לבצע את מה שנראה הכי "משתלם". מסתבר שיש לא מעט בעיות שהאסטרטגיה הזאת פותרת היטב, או לעיתים מביאה לקירוב סביר של פתרון אופטימלי.

1.1 בעיות תזמון (*scheduling*)

1.1.1 בעיה $\#1$: תזמון משימות

יש לנו פרק זמן נתון (מזמן 0 עד לזמן T), ויש משימות שברצונינו לבצע.
לכל משימה J_i יש זמן התחלה a_i וזמן סיום b_i.
אין ביכולתינו לבצע בו־זמנית יותר ממשימה אחת, והיינו רוצים לבצע כמה שיותר משימות.

האלגוריתם החמדן:

התחל מהמשימה בעלת זמן הסיום המוקדם ביותר, ז"א נביט ב־i כך שb_i מזערי,
בצע את משימה i, פסול כל משימה אחרת שמתנגשת בה, והמשך באותו האופן.

טענה 1.1 האלגוריתם הנ"ל מבצע את המספר המירבי האפשרי של משימות.

טענה 1.2 הטענה נובעת מ**טענת עזר:**
יהיו $i_1, ..., i_k$ האינדקסים של המשימות שאותן יבצע האלגוריתם החמדן. (נסמנו $Greedy$).
ויהיו $j_1, ..., j_m$ המשימות שמבצע אלגוריתם (מותר וחוקי) אחר. (נסמנו Alg).
הקטעים שבהם $Greedy$ מטפל הם, אם כן, $[a_{i_1}, b_{i_1}], [a_{i_2}, b_{i_2}], ...$
ושל Alg: $[a_{j_1}, b_{j_1}], [a_{j_2}, b_{j_2}], ...$
הטענה שלנו היא:
לכל אינדקס $r \leq k$ מתקיים $b_{i_r} \leq b_{j_r}$.
(ובמילים אחרות, האלג' החמדן גומר את משימתו ה־r־ית לא יאוחר מהזמן שבו מסיים האלג' האחר את המשימה ה־r־ית שלו).

הוכחה: את הטענה הזאת אנו נוכיח באינדוקציה על r.
מקרה בסיס: $r = 1$ הוא בדיוק התנאי המגדיר את הצעד הראשון של $Greedy$.
צעד האינדוקציה: נניח כעת (בשלילה) שיש דוגמא נגדית, ונביט ב־r המזערי שעבורו הטענה נכשלת.
יהיה X הקטע האחרון (ה־r) שבו טיפל $Greedy$, ו־Y הקטע האחרון שבו טיפל Alg.
באופן דומה, יהיו X', Y' הקטעים ה $r-1$ שלהם.
ע"פ הנחת האינדוקציה, X' מסתיים לא מאוחר מ־Y', ולכן $Greedy$ היה אמור (וגם יכול) להעדיף, את Y כמשימתו ה־r על פני X.
ולכן הגענו לסתירה, והוכחנו את הנדרש. ■

כעת, נראה שטענת העזר אומרת ש$Greedy$ אכן מבצע את המס' המירבי של משימות: כלומר $k \geq m$.
הוכחה: לשם כך, נשתמש בטענת העזר עבור $r = k$.

לצורך המחשה, נתבונן: בגלל טענת העזר, ידוע שאת המשימה ה־k האלג' האחר יסיים לבצע בזמן $\leq$ מהאלג' החמדן.

$[1]\quad [2]\quad \cdots\quad [k]$ $\qquad$ *Greedy algorithm*
$[k]\quad ...?$ $\qquad$ *Other algorithm*

נניח בשלילה ש־$k > m$.

ע"פ טענת העזר, זמן הסיום של האלג' החמדן $\geq$ הזמן שבו האלג' האחר סיים את משימתו ה־k.
ואם $m > k$, משמע שעוד היתה משימה נוספת שהאלג' האחר ביצע. אבל, זו גם משימה שהחמדן היה יכול לקחת על עצמו, **סתירה**. ■

1.1.2 בעיה $\#2$: ארגון חדרים לפעילויות

להנהלת בי"ס יש צורך לארגן חדרים לפעילויות הוראה.
כל פעילות היא קטע בזמן, ואי אפשר לקיים שתי פעילויות בו־זמנית באותו החדר.
אנו רוצים למזער את מס' החדרים שישמשו להוראה.
אם יש איזשהו רגע שבו מתקיימות לפחות d פעילויות, נדרשים בוודאי לפחות d חדרים.

טענה 1.3 יהיה $d*$ המס' המירבי של פעילויות הוראה המתקיימות בו־זמנית. אז ניתן לשכנן ב־$d*$ חדרים.
יתר על כן, ניתן למצוא השמה של שיעורים לחדרים ע"י אלגוריתם חמדן.

האלגוריתם:
נעבור על כל הקטעים משמאל לימין (=בכיוון ציר הזמן) ע"פ שעת ההתחלה שלהם.
כל שיעור נפנה לחדר הפנוי הראשון, מתוך ה־$d*$ חדרים.
(נשים לב שאפשר לדעת מראש מהו $d*$ ע"י חישוב פשוט).

הוכחה: (לטענה).
ברור שנדרשים לפחות $d*$ חדרים על מנת לשכן את כל השיעורים. נראה שהאלג' אכן נותן לנו שיכון שכזה.
נעבור על הקטעים כסידרם (ע"פ הקצה השמאלי ־ ההתחלה), ונביט במקרה הראשון שבו אנו נכשלים (אם אכן יש כזה).
זהו המצב הראשון שבו כל החדרים $1, ..., d*$ תפוסים, ויש לנו שיעור נוסף שצריך להתחיל.
אבל הרי במצב זה היינו מקבלים ש־$d*$ גדול ב־1 ממה שהוא, וזוהי סתירה! (ישירות מהגדרת $d*$ לעיל).
(הערה: בהרצאה ראינו וציירנו על הלוח מקרה פרטי עבור $d* = 4$, ונאמר בע"פ שזוהי ההוכחה גם למקרה הכללי. ולכן ייתכן שהניסוח של סוף ההוכחה הוא לא מדוייק, כי הוא פרשנות שלי).

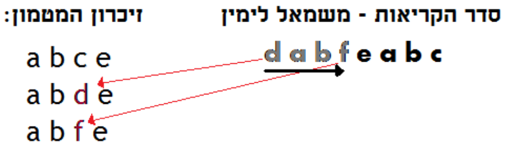
■

1.1.3 בעיה $\#3$: ניהול זיכרון מטמון

הקדמה: יש לנו זיכרון מהיר $cache$ ("זיכרון מטמון") המכיל, נאמר, k יחידות זיכרון ("דפים").
אנו נזקקים במהלך הריצה של המחשב ליחידות שחלקן נמצאות כרגע במטמון, ומפעם לפעם גם יחידות שנמצאות בזיכרון איטי יותר וגדול יותר.
כל פניה לדף יכולה להיות "קליעה" (כלומר הדף הרצוי נמצא במטמון), או "החטאה" ($miss$).
במקרה כזה (של החטאה), אנחנו **מסלקים** איזשהו דף מן המטמון ומכניסים במקומו את הדף החדש ("הטמנה").

הבעיה: יש לנו זיכרון הכולל k דפים, ומאוכלס בתחילה ע"י הדפים $x_1, ..., x_k$.
בהמשך תבואנה דרישות לדפים מתוך הקבוצה $P = p_1, ..., p_n$.
נשים לב: $x_1, ..., x_k \in P$ וגם $k \ll n$. (כלומר n הרבה יותר גדול...).
כאשר מגיעה דרישה לדף $p \in P$ שאינו נמצא כרגע בזיכרון המטמון, עלינו להכניס את p אליו, ולסלק דף אחר ("החטאה").
המטרה: למזער את המס' הכולל של ההחטאות.

האלג' החמדן: "Farthest into the future" (ובקיצור FF).
אנו נסלק מן המטמון את אותו הדף שיידרש שוב בעתיד הרחוק ביותר.



משפט 1.4 לכל $k < n$ ולכל סדרה של דרישות דפים, לאלג' FF יהיה מספר ההחטאות המזערי האפשרי.

הערה 1.5 באופן עקרוני, אנחנו מתירים לאלג' גם לבצע פעולות סילוק והכנסה של דף שאינו דרוש כרגע (וזה נספר כהחטאה).
אלגוריתם **שאיננו** עושה צעדים כאלה ייקרא **אלגוריתם מצומצם**.

הערה 1.6 יש אלג' טוב ביותר שהוא מצומצם.
למה זה נכון? נביט באלג' לא מצומצם A, המכניס ברגע מסוים דף שאינו נדרש t לתוך זיכרון המטמון.
נבנה אלג' אחר A' שמחקה את הפעילות של A, אבל איננו מבצע את הצעד הנ"ל.
אם הדף t אינו נדרש כלל בעתיד, אז $cost(A') < cost(A)$.
ואם הוא כן יידרש, **אז** כאשר הוא יידרש לראשונה - A' יכניס אותו ויהיה לו אותו מחיר כמו ל־A.

נרצה כעת להוכיח את המשפט:

טענה 1.7 יהיה S אלג' מצומצם שפעולותיו מזדהות עם אלה של FF עד זמן j.
אז יש אלג' S' המתלכד עם FF לפחות עד זמן $j+1$, המקיים $cost(S') \leq cost(S)$.

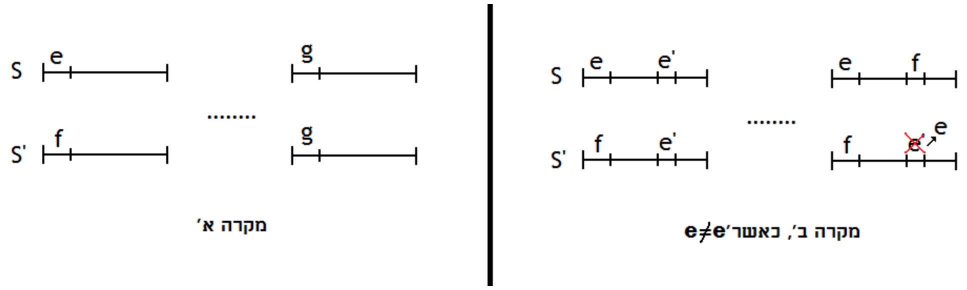
קל לראות שהטענה (+העובדה שדי להתבונן באלגוריתם מצומצם) גוררת את המשפט ש־FF אלגוריתם אופטימלי.

(הסבר שלי: באופן אינדוקטיבי, אם יש לנו אלגו' אחר S שבכלל לא מסכים עם FF, אז $j = 0$. נשתמש בטענה ואז יש לנו אלגוריתם S' שהוא לא פחות טוב מהקודם, כלומר $cost(S') \leq cost(S)$ וגם S' זהה לFF בפעילותו, בצעד הראשון (כלומר הצעד ה$j = 1$). כעת נשתמש בטענה שוב, ועוד פעם יש לנו אלגוריתם S'' שעלול להיות רק יותר טוב, כלומר $cost(S'') \leq cost(S') \leq cost(S)$ וגם 2 הצעדים הראשונים של S'' זהים לאלו של FF (כלומר כעת FF, S'' מסכימים ב$j = 2$ הצעדים הראשונים שלהם). ככה ממשיכים עוד ועוד, עד שמגיעים לאלגו' שהוא זהה לFF בכל צעדיו, ומקבלים שFF הוא קטן־שווה לכל אלגוריתם אחר, ביעילות שלו).

הוכחה: (לטענה האחרונה).
נמשיך ונריץ את S ואת FF ביחד.
אם גם בזמן $j+1$ הם זהים, אז ניקח $S' = S$ וגמרנו.
אחרת:
נניח ש־S מסלק את f מהמטמון, בעוד ש־FF מסלק את e מהמטמון. $(f \neq e)$ (מדובר כאן על הצעד ה$j+1$).
הערה שלי - נשים לב (חשוב לסוף ההוכחה): FF העדיפה לסלק את e **על פני** f, ולכן ידוע לנו שתתבצע בקשה ל־f לפני שתתבצע בקשה ל־e, אם בכלל.
הרעיון הכללי למה שאנו דורשים מ־S' הוא:
S' יסלק גם הוא את e מהמטמון, ומכאן ואילך הוא "ינסה" לחקות את S טוב ככל האפשר, ללא הגדלה במחיר.
וכעת נמשיך, באופן פורמלי יותר:
S' יסלק גם הוא את e מהמטמון, ויבצע מכאן והלאה את אותן הפעולות שעושה S, עד שלראשונה קורה אחד הדברים הבאים:

1. **מקרה א'**: ברגע מסוים נדרש g, כאשר $g \neq e, f$ ואינו במטמון של S, ו־S מסלק את e לכבודו.
   במקרה זה, S' יכניס את g במקומו של f. מכאן ואילך הם זהים.

2. **מקרה ב':** ברגע מסוים, נדרש f.
כאן S חייב להשמיט איבר כלשהו, נאמר e'.
אם $e'=e$, אז S' אינו עושה דבר וממשיך כמו S, תוך חיסכון של 1 במחיר.
אם $e'\neq e$, אז S' גם הוא יסלק את e' לטובת e, ושוב הם בשויון. במקרה זה, S' כבר לא מצומצם: נצמצם אותו, ללא עליה במחיר (לפי מה שראינו בהתחלה).



נטען, שחייבת להתרחש אחת מהאופציות הנ"ל בלבד.
אפשרות אחת למה שיכול לקרות היא שעד סוף הריצה S' יכול להמשיך לחקות את S. כלומר מהצעד שבו S' חיקתה את FF, אין כל בעיה ושני האלגוריתמים S,S' יכולים לבצע את אותן פעולות בלי "לגעת" בתא הזיכרון שבו נמצא ההבדל (e,f ב־S,S' בהתאמה).
אנחנו ניתקל בבעיה רק אם יש איזשהו **רגע שבו** e **נדרש**. במצב כזה, S איננו משלם, אבל S' כן משלם.
מההגדרה של FF, נובע שהזמן הראשון שבו f נדרש, הוא קודם לזמן הראשון שבו e נדרש (ראשון - כלומר לאחר j).
כלומר המצב שבו נדרש e יקרה רק לאחר שיידרש f, ולכן אנו קודם כל חוזרים למקרה ב' הנ"ל.

> הערה\הסבר שלי: (ולכן ייתכן שתיבת טקסט זאת היא שקר מוחלט...)
> להלן פירוט של כל האופציות:
>
> - אם $g\neq f,e$ אבל הוא כן במטמון של S, זה לא מעניין אותנו כי לא מתבצעת הטמנה.
> - (מקרה א'): אם $g\neq f,e$ אבל הוא לא במטמון של S, ו־S מסלק את e לכבודו, אז טיפלנו בזה במקרה א'.
> - אם $g\neq f,e$ אבל הוא לא במטמון של S, ו־S מסלק את $k\neq e$ לכבודו, אז כמובן בגלל שS' מחקה את S, אז זו התנהגותו גם כן.
> - (מקרה ב'): אם $g=f$, טיפלנו בזה במקרה ב'.
> - אם $g=e$, כאן כל הרעיון של ההוכחה - FF בחר לסלק את e מהמטמון כי זהו הדף שיידרש שוב בעתיד הרחוק ביותר, ולכן בהכרח אחד מהמקרים א' או ב' חייב לקרות לפני מצב זה.

■

הערה 1.8 הבעיה החשובה באמת ב"חיים האמיתיים" בתחום הטיפול בזכרונות מהירים היא הבעיה המקוונת - שבה סדרת הדרישות **אינה** ידועה לנו מראש. הוזכר בהרצאה, אך לא פורט, האלגוריתם $LRU-least\,recently\,used$, שכל פעם בוחר את הדף שהשתמשנו בו בעבר הכי רחוק.

1.2 קודי Huffman (מהתרגול)

1.2.1 הבעיה

נתון א"ב $A=\{a_1,...,a_n\}$. לכל אות ב־A נתונה התדירות שלה $f(a_i)$. ומתקיים: $\sum_{i=1}^{n}f(a_i)=1$.

נרצה לקודד טקסט שתדירויותיו נתונות ע"י f, ע"י קוד בינארי:

$$C : A \to \{0,1\}^*$$

כך שאורך הקידוד הממוצע: $L = \sum_{i=1}^{n} f(a_i) l(C(a_i))$ יהיה מינימלי. (כאשר $l(C(a_i))$ הוא מס' הביטים ב־$C(a_i)$).
דרישה על C:

1. הדרישה המינימלית היא ש־C תהיה חח"ע (כלומר **שלא** ייקרה, למשל, $C(a) = 0,\ C(b) = 0$).

הערה: זוהי דרישה לא מספקת על מנת שיהיה ניתן לפענח את הקוד. לדוגמא: $\begin{cases} C(a_1) = 0 \\ C(a_2) = 00 \end{cases}$ אז הקידוד "00" יכול להתפרש או כ־"a_1a_1" או כ־"a_2". ולכן -

2. נדרוש ש־C תהיה ניתנת לפענוח יחיד.
כלומר שההרחבה של C לרצפים של אותיות מ־A,
כלומר $C' : A^* \to \{0,1\}^*$ כך ש־ $C'(\alpha_1, ..., \alpha_n) = C(\alpha_1)....C(\alpha_n)$, תהיה חח"ע.

הגדרה 1.9 קוד C ייקרא **חסר־רישא** אם לכל שתי מילות קוד $w_1, w_2 \in Im(C)$ מתקיים שw_1 איננו רישא של w_2. כלומר לא קיים $\beta \in \{0,1\}^*$ כך ש־ $w_2 = w_1\beta$.

טענה 1.10 קוד חסר־רישא ניתן לפענוח יחיד.

הוכחה: (זו לא הוכחה פורמלית!) נקרא אות־אות מתחילת הקוד. בכל שלב - או שקראנו מילת קוד מלאה, או שקראנו רישא של מילת קוד מלאה. אם קראנו מילה מלאה, ניתן לפענח אותה ולהמשיך משם (וזו בטוח האפשרות היחידה). ■

מעתה נדון רק בקודים חסרי־רישא. (נאמר בכיתה שאפשר להוכיח שקיים קוד אופטימלי שהוא חסר רישא - כל זאת ועוד בקורס שאפשר לקחת בתורת האינפורמציה...).

1.2.2 עצי קידוד, והאלגוריתם של Huffman

עצי קידוד הם ייצוג לקודים חסרי רישא. לקוד נתון, נתאים עץ בינארי, שעליו יתאימו למילות הקוד שלו.
נתאים לכל קודקוד בעץ מחרוזת בינארית, כך שהשורש יקבל ε (=מחרוזת ריקה), ולכל בן שמאלי, המחרוזת שלו תתקבל ממחרוזת אביו ע"י שרשור 0 (וכנ"ל לימני, עם 1).

הרעיון: נקודד אותיות נדירות ע"י מילים ארוכות, ואותיות נפוצות ע"י מילים קצרות.

קידוד $Huffman$:

1. נמצא את האותיות x, y ב־A שמופיעות הכי מעט.

2. נאחד אותן לאות אחת z בעלת תדירות $f(z) = f(x) + f(y)$.

3. נגדיר א"ב חדש: $A' = (A \backslash \{x, y\}) \cup \{z\}$ ונריץ שוב את קוד האפמן על A' עם f המתקבלת. בעץ שהתקבל, נחליף את העלה של z בקודקוד בעל שני ילדים - אחד x והשני y.

(עשינו דוגמה בכיתה, אבל לדעתי אחרי שעשינו את הנושא בקורס דאסט שנה שעברה זה מיותר לאייר אותה ולהשקיע את המאמץ...)
פסיאודו קוד:

Q←priority queue(A,f)

```
for i=1... (|A|-1) :
    x←extract_min(Q)
    y←extract_min(Q)
    add z to Q with priority f(x) + f(y)
    z.right←x
    z.left←y
return root
```

ניגש להוכיח את אופטימליות הקידוד.

טענה 1.11 בעץ אופטימלי, לכל קודקוד יש שני בנים או אפס.

הוכחה: נניח בשלילה שבעץ אופטימלי T יש קודקוד v עם בן אחד. אז נבנה T' עץ המתקבל מחיבור ישיר של בנו של v לאביו. עץ זה מקצר את אורכי הקידוד של חלק מהאותיות מבלי להאריך את של האחרות, כלומר T' טוב יותר מ־T, וקיבלנו סתירה להיות T אופטימלי. ■

טענה 1.12 (1): קיים עץ אופטימלי שבו x, y (האותיות הנדירות ביותר) הם אחים ברמה התחתונה.

הוכחה: יהי T עץ אופטימלי. לפי הטענה הקודמת, יש לו שני קודקודים אחים ברמה התחתונה (עלים). נסמנם: z_1, z_2.
נבנה עץ T' שבו נחליף את x ב־z_1. ונראה ש־T' אופטימלי. (ונזכור שגם x הוא עלה). נחשב, כאשר $d_T(x)$ הוא העומק ב־T של x:

$$L(T) - L(T') = \sum_{i=1}^{n} f(a_i) d_T(a_i) - \sum_{i=1}^{n} f(a_i) d_{T'}(a_i) =$$

מכיוון שיש שויון בין כל האיברים בעצים חוץ מאלה שהחלפנו, זה שווה ל־

$$= f(x)d_T(x) + f(z_1)d_T(z_1) - f(x)d_{T'}(x) - f(z_1)d_{T'}(z_1) =$$

וזה שווה לביטוי הבא, מהגדרת T':

$$= f(x)d_T(x) + f(z_1)d_T(z_1) - f(x)d_T(z_1) - f(z_1)d_T(x) =$$

$$= (f(x) - f(z_1)) \cdot (d_T(x) - d_T(z_1)) \geq 0$$

כאשר המעבר האחרון הוא נכון כי האיבר בסוגריים השמאליים הוא ≤ 0 וכך גם האיבר בסוגריים הימניים הוא ≤ 0.
לכן קיבלנו ש־T' הוא אופטימלי.
ניתן לבצע אותו הדבר עם z_2, y ולקבל עץ אופטימלי כנדרש. ■

טענה 1.13 (2) (הוכחת נכונות האלג'): בהינתן A, f נגדיר $A' = (A \backslash \{x, y\}) \cup \{z\}$.
(x, y האותיות הנדירות ב־A, ו־$f(z) = f(x) + f(y)$).
אזי, אם T' הוא אופטימלי עבור A', אז T המתקבל מ־T' ע"י החלפת העלה של z בקודקוד שלו שני בנים x, y, הוא אופטימלי עבור A.

הוכחה: ראשית כל, נחשב:

$$\bigstar\, L(T) = \left(\sum_{a \in A' \backslash \{z\}} f(a) d_T(a) \right) + f(x) d_T(x) + f(y) d_T(y) = \left(\sum_{a \in A' \backslash \{z\}} f(a) d_T(a) \right) + f(x)(d_{T'}(z)+1) + f(y)(d_{T'}(z)+1) =$$

$$= \left(\sum_{a \in A' \backslash \{z\}} f(a) d_T(a) \right) + (f(x) d_{T'}(z) + f(y) d_{T'}(z)) + f(x) + f(y) =$$

$$\left(\sum_{a \in A' \backslash \{z\}} f(a) d_T(a) \right) + (f(z) d_{T'}(z)) + f(x) + f(y) = L(T') + f(x) + f(y)$$

יהי P עץ אופטימלי עבור A, כך ש־ x, y עלים בו ברמה התחתונה (קיים לפי טענה קודמת).
יהי P' עץ המתקבל מ־P ע"י החלפת הקודקוד שהוא האב של x, y, בעלה z.
אז מאופטימליות T' נקבל:

$$\bigstar\bigstar\, L(T') \leq L(P')$$

ומחישוב זהה לקודם נקבל גם:

$$\bigstar\bigstar\bigstar\, L(P) = L(P') + f(x) + f(y)$$

ועל כן:

$$L(T) = L(T') + f(x) + f(y) \leq L(P') + f(x) + f(y) = L(P)$$

(כאשר השויון השמאלי הוא מ־$\bigstar$, אי השויון הוא מ־$\bigstar\bigstar$, והשויון הימני הוא בגלל $\bigstar\bigstar\bigstar$).
ולכן מאופטימליות P, סיימנו. ■

1.2.3 אי־שוויון Kraft

טענה 1.14

1. אם $l_1, ..., l_n$ הם אורכי מילות קוד של קוד חסר־רישא, אז $\sum 2^{-l_i} \leq 1$.
2. אם $l_1, ..., l_n \in \mathbb{N}$ כך ש־ $\sum 2^{-l_i} \leq 1$ אז קיים קוד חסר־רישא שאלה אורכי מילות הקוד שלו.

הוכחה: (רק ל־1, בערך...)
נניח ש $l_1 \geq l_2 \geq ... \geq l_n$. נתבונן בעץ המתאים לקוד, ונשלים אותו לעץ שלם בעומק l_1. לעץ השלם יש 2^{l_1} עלים.
כל עלה בעץ המקורי ברמה l_i תורם $2^{l_1-l_i}$ עלים לרמה התחתונה. ולכן,

$$\sum_{i=1}^{n} 2^{l_1-l_i} \leq 2^{l_1}$$

(האי־שויון הוא בגלל שלא בהכרח כל העלים בעץ המלא הם צאצאים של עלים בעץ המקורי).
ולכן נסיק שמתקיים $\sum_{i=1}^{n} 2^{-l_i} \leq 1$.
(כי $2^{l_1} \sum 2^{-l_i} = \sum 2^{l_1-l_i} \leq 2^{l_1}$ ונחלק את האגפים ב־ 2^{l_1}). ■

בזאת נגמר החומר מהתרגול ־ מעתה ואילך החומר הוא מההרצאות.

1.3 עץ פורש מינימלי

1.3.1 הבעיה

אנו נדון באלגוריתם החמדן למציאת עץ פורש מינימלי.
הקלט: גרף קשיר $G = (V, E)$ ופונקציית משקל $w : E \to \mathbb{R}^+$.
הפלט הרצוי: עץ פורש T ב־G, בעל משקל כולל מזערי.
(המשקל של עץ T מוגדר כ־$\sum w(e)$ כלומר סכום משקל כל הצלעות e ב־T).
הערה 1.15 בעיית העץ הפורש המינימלי והמקסימלי שקולות.
מדוע זה נכון ־
בכל עץ פורש בגרף עם n קודקודים יש $n-1$ צלעות. לכן, אם נחליף בכל צלע e את המשקל $w(e)$ במשקל $\omega(e) = M - w(e)$ כש־M קבוע גדול מספיק, נקבל שפתרון עץ פורש מינימלי עם משקלות אלו נותן למשעה עץ פורש מקסימלי. (הערה: הניסוח כפי שהעתקתי אותו מהלוח בהרצאה לא היה ברור בעיניי, אז שיניתי אותו קצת).

האלגוריתם החמדן לבניית עץ פורש מקסימלי:
בכל צעד, צרף את הצלע בעלת המשקל המירבי, כך שבידך עדיין יער.

1.4 ההקשר הרחב יותר ־ מטרואידים

יש קבוצת בסיס E,
ומשפחה $\mathcal{F}$ של תת־קבוצות של E.
(בדוגמא הקודמת ־ $\mathcal{F}$ אוסף כל היערות ב־G).
נניח ש$\mathcal{F}$ היא **<u>תורשתית</u>**, זאת אומרת **סגורה למעבר לתת־קבוצה** $\Leftrightarrow$ **סגורה להשמטת איברים**.
ז"א, $A \in \mathcal{F}$, $B \subseteq A$ $\Leftarrow$ $B \in \mathcal{F}$.
כמו כן, יש פונקצית משקל $w : E \to \mathbb{R}^+$ ואנו רוצים לפתור את הבעיה הבאה:
בעיה: מצא $A \in \mathcal{F}$ כך ש־$\sum_{e \in A} w(e)$ מקסימלי.

1.4.1 דוגמא: בעיית הזיווג הממושקל המקסימלי

הגדרה 1.16 יהיה G גרף. **זיווג** ($matching$) ב־G זהו אוסף צלעות ללא קודקודים משותפים. (כלומר שלאף שתיים מהן אין קודקוד משותף).

$\mathcal{F}$ - אוסף כל הזיווגים ב־G.
בעיית הזיווג הממושקל המקסימלי:
קלט: גרף $G=(V,E)$ ופונק' $w:E\to\mathbb{R}^+$.
פלט: מצא זיווג M ב־G בעל משקל מירבי, ז"א

$$max_{M\, is\, a\, matching}\left(\sum_{e\in M} w(e)\right)$$

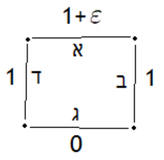
1.4.2 האלגוריתם החמדן המוכלל (שלא תמיד עובד...)

האלגוריתם החמדן לבעיה הנ"ל (שלא נותן בהכרח פתרון אופטימלי!!)

נתחיל מ־ $S=\emptyset$.
בכל צעד, צרף ל־S איבר e כך ש־

1. $S\cup\{e\}\in\mathcal{F}$
2. $w(e)$ מקסימלי בתנאי זה.

נראה דוגמה שבה האלגוריתם לא יימצא פתרון מקסימלי:



$\mathcal{F}$ ={{א},{ב},{ג},{ד},{א,ג},{ב,ד}}

שלבי הרצת האלג' החמדן הם:

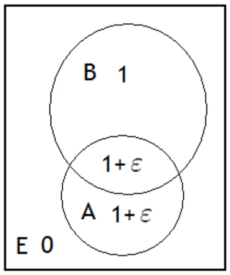
1. $S=\emptyset$
2. S ={א}
3. S ={א,ג} וזהו לא זיווג מקסימלי!! ({ב,ד} הוא כמובן המקסימלי).

קל לראות שהדוגמה שראינו זה עתה ניתנת להכללה.
נניח שיש ב־$\mathcal{F}$ שתי קבוצות $A,B\in\mathcal{F}$ כך ש־$|A|>|B|$.
וגם, אין אף איבר $e\in B\backslash A$ כך שגם $A\cup\{e\}\in\mathcal{F}$.
(הערה שלי: בדוגמה לעיל, למשל $B=$ {ב,ד} , $A=$ {א})
נראה שאם יש ב־$\mathcal{F}$ שתי קבוצות A,B כנ"ל אזי האלג' החמדן **ייכשל** על $\mathcal{F}$.

הרעיון: נמצא משקלות w כך ש:

1. B עדיף על A: $w(B) > w(A)$ (כלומר סכום ה־w של האיברים)
2. אף על פי כן, החמדן יילך על A.

המשקלות:
על איברי A, $a \in A$, $w(a) = 1 + \varepsilon$.
על איברי $B \backslash A$, $e \in B \backslash A$, $w(e) = 1$.
על כל איבר אחר e ב־E, $w(e) = 0$.



מה יעשה האלג' החמדן?
הוא יצרף בזה אחר זה את האיברים שב־A עד שיגיע ל־ $S = A$ (כי אלו האיברים ה**כבדים ביותר**), שם הוא ייעצר, כי אין אף איבר ב־B שניתן לצרף ל־A ולהשאר ב־$\mathcal{F}$. המשקל שהוא קיבל: $|A| \cdot (1 + \varepsilon)$.
אבל,
בין המועמדים לבחירה נמצא גם $B \in \mathcal{F}$. במקרה זה, המשקל שנקבל הוא $|B| \leq$.
אם $|B| > (1 + \varepsilon) \cdot |A|$, (*) אז האלג' החמדן **לא מצא** את הפתרון הטוב ביותר.
היות שע"פ ההנחה $|B| > |A|$, יש $\varepsilon > 0$ שבשבילו (*) תקף.

משפט 1.17 תהיה $\mathcal{F}$ משפחה תורשתית של תת־קבוצות של הקב' E.
אזי, האלג' החמדן מצליח למצוא $S \in \mathcal{F}$ בעלת משקל מירבי לכל $w : E \to \mathbb{R}^+$,
$\Leftrightarrow$אם"ם
לכל $A, B \in \mathcal{F}$ כך ש־$|A| < |B|$ יש $e \in B \backslash A$ כך ש־ $A \cup \{e\} \in \mathcal{F}$.

1.4.3 מטרואידים

תהיה $\mathcal{F}$ משפחה תורשתית של תת־קבוצות של הקבוצה E.
אומרים ש־$\mathcal{F}$ **<u>מקיימת את תכונת ההחלפה</u>** אם לכל $A, B \in \mathcal{F}$ כך ש־$|A| < |B|$, יש $e \in B \backslash A$ כך שגם $A \cup \{e\} \in \mathcal{F}$.

הגדרה 1.18 משפחה תורשתית של קבוצות $\mathcal{F}$ המקיימת את תכונת ההחלפה נקראת **אוסף הקבוצות הבלתי תלויות של מטרואיד.**

משפט שכבר ניסחנו קודם -
תהיה $\mathcal{F}$ משפחה תורשתית של קבוצות.
האלגוריתם החמדן מוצא את $A \in \mathcal{F}$ בעלת משקל מירבי **לכל** פונק' משקל w $\Leftrightarrow$ $\mathcal{F}$ היא אוסף הקבוצות הבלתי תלויות של מטרואיד.

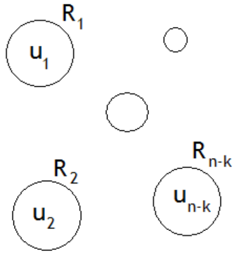
1.4.4 דוגמא: על וקטורים, קבוצות בת"ל ומטרואידים

יהיה E אוסף סופי של וקטורים במרחב וקטורי כלשהו.
נגדיר ש: $A \in \mathcal{F}$ כאשר $A \subseteq E \Leftrightarrow A$ בלתי תלויה ליניארית.
נשים לב ש־$\mathcal{F}$ היא אוסף הקבוצות הבלתי תלויות של מטרואיד:
תורשתיות: אם S בת"ל, $W \subseteq S$ אז גם W בת"ל.
תכונת ההחלפה: אם $A, B \in \mathcal{F}$ (ז"א שתיהן קבוצות בת"ל ־ הכוונה כאן שאיברי A בת"ל זה בזה, ואיברי B בת"ל זה בזה), וגם $|B| > |A|$, אז $dim(B) > dim(A)$ ולכן יש $x \in B \backslash A$ כך שגם $A \cup \{x\}$ בת"ל.

1.4.5 דוגמא נוספת: על יערות ומטרואידים

טענה 1.19 יהיה $G = (V, E)$ גרף, ויהיה $\mathcal{F}$ אוסף כל ה־$S \subseteq E$ שהן יער.
אז גם $\mathcal{F}$ היא אוסף הקב' הבלתי תלויות של מטרואיד.

פירוש הטענה:
יהיה $G = (V, E)$ גרף קשיר (אם הוא לא, מסתכלים על כל רכיב קשירות לחוד...), ויהיו $A, B \subseteq E$ שהן יערות.
וגם $|B| > |A|$. אזי, יש $e \in B \backslash A$ כך שגם $A \cup \{e\}$ יער.
הוכחה: נסמן ב־n את מספר הקודקודים ב־G.
ונסמן ב־k את מספר הצלעות ב־A: $|A| = k$.
מהו מספר רכיבי הקשירות ביער (V, A)? $n - k$.
(זה נכון כי אפשר להתבונן על זה ככה: נתחיל ב־n רכיבי קשירות כאשר אין צלעות בכלל, ואז בכל פעם שאנו נוסיף צלע, אז מכיוון שאסור לנו ליצור מעגלים, בכל פעם כזו נוריד את מס' רכיבי הקשירות ב־1. וכך נמשיך להוסיף את כל k הצלעות עד שנקבל $n - k$ רכיבים).



נסמן: u_i מס' הקודקודים ברכיב ה־i של היער (V, A). וכל רכיב נסמן ב־R_i.
אז $|R_i| = u_i$, וגם $\sum u_i = n$.
צלע ב־B פסולה מלהצטרף ל־$A \Leftrightarrow$ שני קודקודיה הם באותו רכיב קשירות של (V, A). ז"א, מטרתינו היא להראות שיש ב־B צלע הקושרת בין שני רכיבים שונים של (V, A).
מספר הצלעות של B ששני קודקודיהן ב־R_i הוא $\geq u_i - 1 = |R_i| - 1$ (כי אין מעגלים ב־R_i...).
ולכן נפסלות לכל היותר:

$$\sum_{i=1}^{n-k}(u_i - 1) = \left(\sum_{i=1}^{n-k} u_i\right) - (n-k) = n - (n-k) = k$$

מן הצלעות שב־B.
אבל $|B| > |A| = k$ ולכן נותרה ב־B לפחות צלע אחת מיותרת.
הוכחנו: אוסף היערות החלקיים בגרף הוא אוסף הקבוצות הבלתי תלויות של מטרואיד. ■

1.5 סיכום מטרואידים

תהיה E קבוצה סופית.
אומרים ש־$\mathcal{I}$ הוא אוסף הקבוצות הבלתי תלויות של מטרואיד בבסיס E,
אם $\mathcal{I}$ היא **משפחה תורשתית** (ז"א $A \in \mathcal{I}$, $B \subseteq A \Leftarrow B \in \mathcal{I}$),
המקיימת את **תכונת ההחלפה**, ז"א אם $A, B \in \mathcal{I}$ ו־ $|B| > |A|$ אז יש $x \in B \backslash A$ כך שגם $A \cup \{x\} \in \mathcal{I}$.

ראינו שאם $\mathcal{F}$ משפחה תורשתית של קבוצות אשר איננה מקיימת את תנאי ההחלפה, אז יש מערכת משקלות w על E המכשילה את האלג' החמדן כשהוא מנסה למצוא קבוצה ב־$\mathcal{F}$ בעלת משקל מירבי.
אמרנו, מאידך גיסא (ולא הוכחנו), שאם תנאי ההחלפה מתקיים, ז"א $\mathcal{F}$ היא משפחת הקבוצות הבלתי תלויות של מטרואיד, אז האלג' החמדן יצליח על כל מערכת של משקלות.

1.5.1 בסיס, מעגל, חתך

במטרואיד, מעבר למושג הבסיסי של משפחת הקב' הבלתי תלויות $\mathcal{I}$, ניתן להגדיר עוד מערכת שלמה של מושגים **אנלוגיים** למה שמוכר מגרפים, ובעזרתם מוכיחים שאכן האלג' החמדן עובד במטרואידים.

- **בסיס:** קבוצה בלתי תלויה מקסימלית (שאינה מוכלת הכלה של ממש בקבוצה ב"ת אחרת).

  (זה אנלוגי לעץ פורש). באלגברה ליניארית - בסיס ליניארי ל־E.

- **מעגל** C**:** קבוצה תלויה (ז"א איננה ב־$\mathcal{I}$) שהיא מינימלית בתנאי זה, ז"א אם $C \subsetneqq B$ אז $B \in \mathcal{I}$.

- **חתך:** זו קבוצה שיש לה חיתוך לא ריק עם כל בסיס, והיא מינימלית בתנאי זה.

2 תכנון דינאמי (Dynamic Programming)

2.1 הכפלה יעילה של סדרת מטריצות

2.1.1 מוטיבציה \ הקדמה

המחיר של הכפלת שתי מטריצות:

$$A_{r\times s} \cdot B_{s\times t} = C_{r\times t}$$

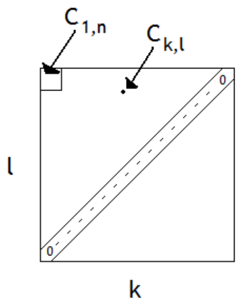
הוא באופן נאיבי $r \cdot s \cdot t$.
נניח שאנו רוצים להכפיל $A_1 \cdot A_2 \cdot \ldots \cdot A_n$, כאשר A_i היא מטריצה בעלת מימדים $\alpha_{i-1} \times \alpha_i$.
נזכור שכפל מטריצות הוא אסוציאטיבי: $(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$ כלומר, עצם התוצאה אינו תלוי בסדר ההכפלות.
אבל המחיר תלוי בהחלט!
כאמור מתקיים $cost(A_{r\times s} \cdot B_{s\times t}) = r \cdot s \cdot t$.
מספר המסגורים המותרים באורך n (כלומר מס' האפשרויות לסדרות של סוגריים שמייצגות אופציה "חוקית" של סידור הסוגריים) הוא מספר $Catalan$, ומספרים אלו גדלים בצורה מעריכית עם n. ולכן נרצה למצוא את הסידור האופטימלי לסוגריים בדרך יעילה.
כעת,

- יש מושג מוגדר היטב של תת־בעיה.
- כל תת־בעיה כזאת עלינו לפתור בפני עצמה באופן אופטימלי.
- פתרונות גלובאלים שונים עשויים להזדקק לפתרונה של אותה תת־בעיה.

רעיון מרכזי:

לזהות תת־בעיות כאלה, לרשום את פתרונותיהן, ולבנות מהם פתרון גלובאלי אופטימלי.

בעצם, אנחנו נחשב לא רק את המכפלה $A_1 \cdot ... \cdot A_n$, אלא את כל המכפלות $A_k \cdot ... \cdot A_l$ לכל $k < l$.
ונחשב כל אחד מהביטויים האלה בצורה האופטימלית.
נגדיר: לכל $k \le l$, את $c_{k,l}$=מס' הפעולות המזערי לחישוב המכפלה $A_k \cdot ... \cdot A_l$.
בסה"כ, יש $\binom{n}{2}$ מחירים. (זה מס' האפשרויות לבחור זוגות k, l).
אז הגודל שבו התעניינו איננו אלא $c_{1,n}$.
נבנה מערך דו־מימדי $n \times n$ כשבמקום ה־k, l שלו נרצה לרשום את $c_{k,l}$.



אתחול: מתקיים $c_{k,k+1} = \alpha_{k-1}\alpha_k\alpha_{k+1}$ וגם $c_{k,k} = 0$.
על מנת לחשב את $c_{k,l}$ אנחנו נעבור על כל הדרכים האפשריות שבהן יכולה להיראות פעולת הכפל האחרונה בחישוב אופטימלי של $A_k \cdot ... \cdot A_l$. נאמר שזה קורה במקום t.
כלומר אנו מסתכלים על מצבים מהסוג: $(A_k \cdot ... \cdot A_t) \cdot (A_{t+1} \cdot ... \cdot A_l)$
ולכן, ההגדרה הרקורסיבית היא

$$c_{k,l} = min_t \ \ (c_{k,t} + c_{t+1,l} + \alpha_{k-1}\alpha_t\alpha_l)$$

(המחובר הימני ביותר הוא עלות הפעולה האחרונה).
לגבי הסיבוכיות של האלג' הזה:
יש בערך $\frac{n^2}{2}$ שלבי חישובים כדי לחשב את כל המידע (סדרה חשבונית של $1, ..., n-1, n$), מהאלכסון של האפסים ועד לפינה השמאלית־עליונה.
כל חישוב של $c_{k,l}$ מחירו $\le n$, ובסה"כ נקבל שהמחיר $\le \frac{n^3}{2}$, כלומר $O(n^3)$.

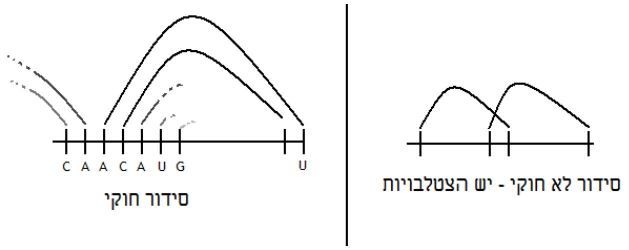
2.2 בעיה בביו־אינפורמטיקה

2.2.1 "המבנה השניוני של מולקולת RNA" - הקדמה

על מולקולה של RNA ניתן לחשוב כמילה באלף־בית A, C, G, U ("בסיסים").
בגלל התכונות הכימיות של הבסיסים, הם עשויים ליצור ביניהם קשרים ע"פ הכלל שמתקיים $A - U$, $C - G$ (כלומר אלו הקשרים שיכולים להיווצר).
קשר שכזה מקטין את האנרגיה הפוטנציאלית, והמולקולה נוטה להתארגן בצורה שממזערת את האנרגיה הזו.

הבעיה: נתון רצף באותיות A, C, G, U. מצא את זיווג הבסיסים בעל האנרגיה המזערית.

1. מידת הגמישות של המולקולה מוגבלת, ולכן על מנת ששני בסיסים יזווגו, עליהם להיות במרחק ≤ 4 בסדרה.
2. אין "פסיאודו קשרים" ־ נניח ש $i_1 < i_2 < i_3 < i_4$. אז לא ייתכן שזווגו הבסיסים במקומות i_1, i_3 וכן במקומות i_2, i_4.



איך ניגשים לפתרון הבעיה?

במקום לנסות ולמצוא בבת אחת את זיווג הבסיסים האופטימלי, אנחנו נחשב את הפתרון לתת־בעיות המתאימות לתת־קטעים של סדרת ה־RNA הנתונה.

דהיינו, נגדיר $OPT(i,j)$ כך שזוהי האנרגיה המזערית האפשרית בזיווג בסיסי של הקטע ממולקולת ה־RNA הנתונה, ממקום i עד מקום j.

2.2.2 ניסוח מפורש של הבעיה

נתון רצף באותיות A, C, G, U. רוצים לזווג כמה שיותר זוגות, בכפוף לתנאים הבאים:

1. $A - U$, $C - G$ (ולכל אחד לכל היותר בן־זוג יחיד).
2. אין "קשת" קצרה מ־4.
3. אין הצטלבויות.

נקרא בשם $OPT(i,j)$ לערך המיטבי (כמה זוגות יצרנו) המתאים לקטע בסדרה הנתונה המתחיל במקום ה־i ומסתיים במקום ה־j.

אם $j - i \leq 3$ אז $OPT(i,j) = 0$.

כעת, עבור $l > k$:

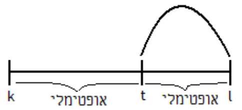
$$OPT(k,l) = max\{OPT(k,l-1),\ (1 + max_t\ [OPT(k,t-1) + OPT(t+1,l-1)])\}$$

כאשר t בביטוי בצד ימין מוגדר כך שמרחקו מ־l הוא ≤ 4 ויש בו בסיס משלים לזה שב־l.

במילים, מה שעשינו הוא זה:

האופציה של $OPT(k,l-1)$ מייצגת את האפשרות שבה l אינו מזווג.

האופציות שבודקים עם t מייצגות מצבים שבהם l מזווג למקום ה־t.



המחשה של המצב בו l מזווג

זמן הריצה הוא $O(n^3)$, כמו בהכפלת סדרת מטריצות ממקודם.

2.3 התאמת זמן דינאמית $Dynamic\, time\, warping$

הבעיה הזו באה מהתחום של עיבוד דיבור.

2.3.1 הבעיה

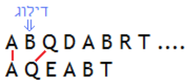
נתונות לנו שתי סדרות של מספרים ממשיים $x_1, ..., x_N$, $y_1, ..., y_M$.
אנו רוצים להתאים ביניהן בצורה מיטבית.
(הסדרות מייצגות דגימות בהפרשים קבועים של צורת־גל של קול, ואנחנו רוצים לדעת האם שתי צורות הגל מייצגות את אותה המשמעות, למשל אם מישהו מדבר לאט וגם מהר, אנחנו רוצים להיות יכולים להבחין שהוא אומר את אותו משפט).
נבנה מערך $N \times M$ שבמקום ה־i, j שלו כתוב $|x_i - y_j|$.
נחפש בגרף ממושקל שנתאר להלן, מסילה זולה ביותר מן המקום ה־$(1,1)$ למקום ה־(N, M).
מסילה מותרת עושה מידי צעד את אחת התנועות הבאות:

$$\underset{one\, diagonal}{\nearrow} \qquad \underset{one\, upwards}{\uparrow} \qquad \underset{one\, to\, the\, right}{\rightarrow}$$

כאשר בצעד ימינה מתעכבים ב־x ומתקדמים ב־y, בצעד למעלה מתעכבים ב־y ומתקדמים ב־x, ובצעד באלכסון מתקדמים בשניהם.
מחיר המסילה הוא $\sum a_{ij}$ על המקומות שבהם היא מבקרת.
(לא דיברנו יותר על הבעיה הזאת בהרצאות).

2.4 חישוב מרחקים בין רצפי חלבונים

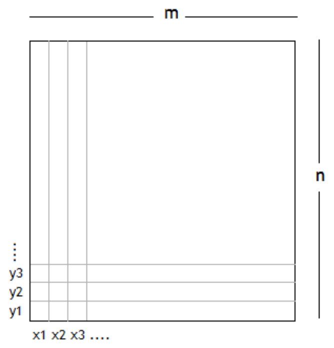
יש לנו אלף־בית של 20 אותיות ־ חומצות אמינו.
רוצים לדעת אם שני רצפים הם "דומים".
הבעיה: נתונות לנו שתי מילים $\begin{matrix} x_1, ..., x_m \\ y_1, ..., y_n \end{matrix}$ באלף־בית Σ.
מתקיים $|\Sigma| = 20$, ונסמן $\Sigma = \{\sigma_1, ..., \sigma_{20}\}$.
כמו כן, נתונה לנו טבלה (קבועה, כחלק מנתוני הבעיה) a_{ij} המספקת הערכה למידת השוני בין האות ה־i ב־Σ לאות ה־j ב־Σ.



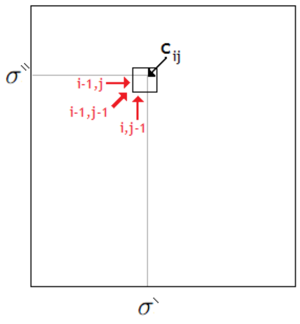
באופן כללי, אנו מחפשים התאמה לאו דווקא מלאה בין האותיות $x_1, x_2, ...$ ל־ $y_1, y_2, ...$ תוך מתן אפשרות גם ל"דילוגים".

- ההתאמה צריכה להיות ללא הצלבות.
- על התאמה בין האות σ_i המופיעה ב־x לאות σ_j ב־y "נשלם מחיר" של a_{ij}.
- על כל השמטה (אות ב־x או אות ב־y שאינן מזווגות) אנחנו נשלם מחיר של Δ.

המטרה: למצוא את ההתאמה "הזולה ביותר".
(הערה לגבי הסימונים: אנו עובדים בציון קורדינטות של (x, y), בטבלה).
נתבונן בטבלה בגובה n ורוחב m ־ לגובה רשומים $y_1, ..., y_n$ ולרוחב רשומים $x_1, ..., x_m$.



התאמה ניתנת לתיאור חד־ערכי ע"י מהלך היוצא מהמקום ה־$(1,1)$ בטבלה,
מגיע למקום ה־(m,n) בטבלה, ובכל צעד נע $\rightarrow$ ימינה, $\uparrow$ למעלה, או $\nearrow$ באלכסון.
האינטרפרטציה היא זו:
צעד ימינה: דילוג על אות ב־x.
למעלה: דילוג על אות ב־y.
אלכסון: התאמה בין אות מ־x לאות מ־y.



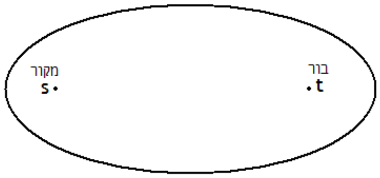
כעת, c_{ij} הוא המחיר המינימלי של התאמה בין $x_1, ..., x_i$ לבין $y_1, ..., y_j$.
c_{mn} יהיה התשובה לשאלה.
כרגיל בתכנון דינאמי, אנו רוצים לבטא את c_{ij} באמצעות c־ים קודמים.

$$c_{i,j} = min\{c_{i,j-1} + \Delta,\ c_{i-1,j} + \Delta,\ c_{i-1,j-1} + a_{x_i,y_j}\}$$

3 זרימה ברשת

3.0.1 בעיית הזרימה ברשת (Network flow problem)

הקלט: רשת, כלומר גרף מכוון $G = (V, E)$, יחד עם שני קודקודים מצויינים $s \neq t \in V$.
s נקרא **המקור**, t נקרא **הבור**, וכן נתונה פונקציית **קיבול** $c : E \to \mathbb{R}^+$.
אנו מניחים שדרגת הכניסה של s היא 0, ושדרגת היציאה של t היא 0.
לצורך הדיון בהרצאה (כלומר לא באופן כללי!), הנחנו ש־c מקבלת רק ערכים שלמים $(1, 2, 3, ...)$.



הגדרה 3.1 זרימה (מותרת):
זוהי פונק' $f : E \to \mathbb{R}^+$ המקיימת:

1. לכל צלע e מתקיים $c(e) \geq f(e) \geq 0$.
2. לכל $v \neq s, t$ מתקיים:

$$\sum_{e\,goes\,into\,v} f(e) = \sum_{e\,goes\,out\,from\,v} f(e)$$

השטף של זרימה מוגדר להיות:

$$\nu(f) = \sum_{e\,goes\,out\,from\,s} f(e)$$

הפלט הרצוי: זרימה מותרת f בעלת שטף מירבי.

3.0.2 טיפה על בעיות אופטימיזציה

זוהי בעיית אופטימיזציה. יש איזשהו תחום D (קבוצה) ופונקציה $g : D \to \mathbb{R}$, והבעיה: למצוא את $max_{x \in D} g(x)$.
בעיית הזרימה היא בעיית אופטימיזציה.

$$D = \{(f(e_1), ..., f(e_n)) \;\mid e_i \in E,\; such\,that\,conditions\,(1)\,and\,(2)\,hold\}$$

הגודל שאותו אנחנו רוצים למקסם הוא ν:

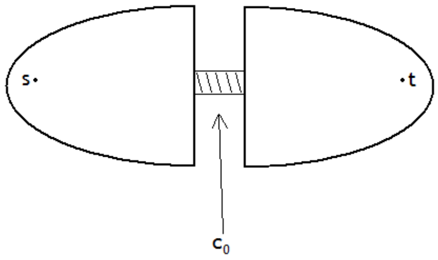
$$\sum_{e\,goes\,out\,from\,s} f(e)$$

בעיה אפשרית: נתונה הרשת ונתון מספר $k > 0$. האם יש זרימה בעלת שטף $\leq k$?

קל לנו לאמת שפתרון נתון עומד בדרישות של הבעיה הזאת. האתגר הוא למצוא פתרונות טובים.

3.0.3 עוד על זרימה

נראה הגיוני (ואח"כ נראה שזה גם נכון), שבמצב כבציור, אין זרימה מותרת f עם שטף גדול מ־ c_0.



הגדרה 3.2 חתך (cut)
חלוקה של V לשתי קבוצות זרות:

$$V = A \dot{\bigcup} B$$

כך ש־$s \in A$, $t \in B$, נקראת חתך.
הקיבול של החתך מוגדר ע"י

$$c(A,B) = \sum_{e \in (A \to B)} c(e)$$

הגיוני (וגם נכון, כפי שנראה), שלכל חתך (A,B) בגרף, ולכל זרימה מותרת f, מתקיים $\nu(f) \leq c(A,B)$.

משפט 3.3 משפט השטף והחתך ($max-flow\, min-cut\, theorem$)
בכל רשת זרימה, השטף המירבי של זרימה מותרת = הקיבול המזערי של חתך ברשת.
(נוכיח בהמשך).

- נובע מכל הנאמר עד כה, שאם נוכל למצוא זרימה f וחתך (A,B) ברשת נתונה כך ש: $\nu(f) = c(A,B)$, אזי f אופטימלית (בעלת שטף מירבי) וכן (A,B) חתך מינימלי (ז"א קיבולו מזערי).

3.0.4 הקדמה לאלגוריתם

אם נתונה לנו זרימה f ואנחנו רוצים לשפר אותה, אז נשים לב שכשאנחנו בונים מסילה P מ־s המיועדת להיות מסילת הרחבה (עוד על כך בהמשך), יש צלע ב־P המהווה "צוואר בקבוק". (כלומר מכל הצלעות ב־P היא המטילה את החסם העליון הנמוך ביותר על הערך שנוכל להזרים לאורך P במאמצינו לשפר את f).

הרעיון הכללי:

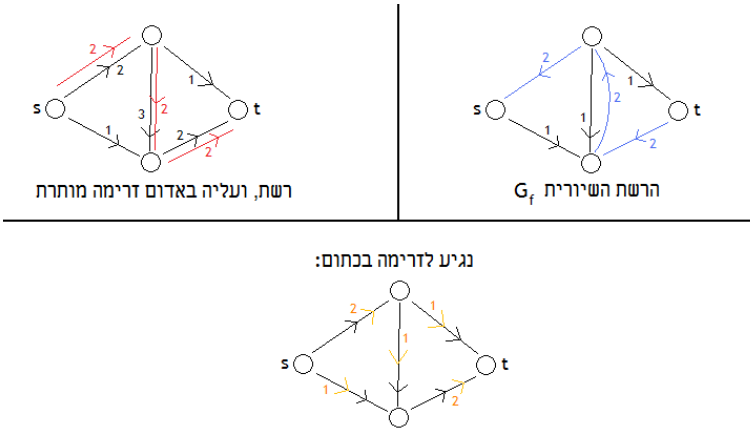
נצא מהזרימה $f \equiv 0$ (זהותית אפס) ומידי צעד ננסה לשפר אותה (להגדיל את $\nu(f)$) ע"י מציאת מסילה P מ־s ל־t.
נמצא מהו a ערך "צוואר הבקבוק" ב־P, ואז נזרים את a לאורך P.

הגדרה 3.4 תהיה G רשת, ו־f זרימה מותרת ב־G.
הרשת השיורית ($residual$) המתאימה G_f נבנית כך:

- קב' הקודקודים V.
- המקור והבור s,t זהים לאלו של G.
- קב' הצלעות כוללת חלקים מ־E, וחלקים מ־$\overleftarrow{E}$ (כלומר הצלעות ב־E, רק שהפכנו את הכיוון שלהן):

$$\{\overrightarrow{e} \mid \overrightarrow{e}\in E,\ f(e)<c(e)\} \bigcup \{\overleftarrow{e} \mid \overrightarrow{e}\in E,\ f(e)>0\}$$

- הקיבולים:
  - אם $e\in E$ אז הקיבול שלה ב־G_f הוא $c(e)-f(e)$.
  - אם $e\in\overleftarrow{E}$ אז הקיבול שלה ב־G_f חיובי רק אם $f(e)>0$, ואז הקיבול הוא $f(e)$.



(הרעיון כאן של הגדרת הקיבולים של $\overleftarrow{E}$ $e\in$, הוא שאם הזרמנו ב־f כמות מסוימת בצלע e, אז אפשר "לבטל" אותה ע"י שליחת מספר שהוא $\leq$לכמות זו, בכיוון השני).

הגדרה 3.5 מסילה מכוונת מ־s,t ב־G_f שכל קיבולי הצלעות בה חיוביים נקראת **מסילת הרחבה**.

3.0.5 אלגוריתם Ford-Fulkerson (סכימה)

1. צא מ־ $f\equiv 0$.
2. בנה בכל צעד רשת שיורית G_f, מצא בה מסילה מכוונת P מ־s אל t, והזרם עליה את הערך המירבי האפשרי ב־P ("צוואר הבקבוק"), a. עדכן את f.

   כלומר, על כל צלע קדמית $\overrightarrow{e}\in E$ החלף את $f(e)$ ב־$f(e)+a$, על כל צלע אחורית $\overleftarrow{e}\in\overleftarrow{E}\cap P$ החלף את $f(e)$ ב־$f(e)-a$, ולכל צלע אחרת, השאר את f בעינה.
3. חזור על 1,2 בלולאה. עוצרים כאשר (ואם) ב־G_f אין מסילה מכוונת מ־s אל t. (בקיבולות ממשיים ($\mathbb{R}$) יש מצבים שהאלג' לא יעצור, כפי שראינו בתרגיל הבית).

(ערך צוואר הבקבוק ב־P זהו הקיבול המזערי של צלע ב־P ברשת G_f).

טענה 3.6 אם ברשת G כל הקיבולים הם מספרים טבעיים, אז אלגוריתם Ford-Fulkerson ימצא זרימה מיטבית בזמן $\sum_{e\in E} c(e)$.
(לא נוכיח את הטענה הזו).
הרעיון: בכל צעד של האלגוריתם, משתפר $\nu(f)$ לפחות ב-1. ברור שלכל f מותרת, $\nu(f) \leq \sum c(e)$.

הערה 3.7 אנחנו חושבים על אלגוריתם כ"יעיל" אם זמן הריצה שלו הוא פולינומי באורך הקלט.
זאת אומרת, יש קבועים c, r כך שאם ניתן לאלגוריתם קלט בעל N ביטים אז זמן הריצה שלו לא יעלה על $c \cdot N^r$.

יש שתי דרכים עיקריות שבהן ניתן לממש את הרעיון של Ford-Fulkerson בצורה הנותנת אלג' בעל זמן ריצה פולינומי.
מסתבר שיש גירסאות יעילות ל-FF (Ford-Fulkerson), למשל, למצוא מסילת הרחבה "רחבה" ככל האפשר, ז"א שערך צוואר הבקבוק שלה גדול ככל האפשר (זהו אלגוריתם מדורג - $scaling$).
דרך אחרת (אלגוריתם Edmonds-Karp) - לבחור תמיד במסילת הרחבה בגרף השיורי בעלת אורך מזערי.
עוד על נושא הסיבוכיות ועל המימושים הנ"ל - בהמשך.

3.0.6 טענות והוכחות

עוד סימונים:

אם f זרימה ברשת, ואם $A \subseteq V$, נסמן

$$f^{out}(A) = \sum_{e\ goes\ out\ from\ A} f(e)$$

וכיוב':

$$f^{in}(A) = \sum_{e\ goes\ into\ A} f(e)$$

טענה 3.8 (כרגע ללא הוכחה)
אם f זרימה ברשת G, ואם אין ב-G_f מסילה מכוונת מ-s ל-t, אז יש ב-G חתך (A, B) (נזכור: $s \in A, t \in B$) כך ש-$\nu(f) = c(A, B)$.
ובפרט, f אופטימלית והחתך (A, B) מינימלי.

טענה:

תהיה G רשת, f זרימה מותרת בה, ו-(A, B) חתך. אזי

$$\nu(f) = f^{out}(A) - f^{in}(A)$$

הוכחה: היות ש $\nu(f) = f^{out}(\{s\})$ והיות ש $f^{in}(\{s\}) = 0$ (הנחנו שאין צלעות נכנסות ל-s), ניתן לרשום:

$$\nu(f) = f^{out}(\{s\}) - f^{in}(\{s\})$$

ומכיוון ש $f^{out}(\{v\}) = f^{in}(\{v\})$ לכל $v \neq s, t$, נקבל:

$$\nu(f) = f^{out}(\{s\}) - f^{in}(\{s\}) + \sum_{v\in A\setminus\{s\}} \left(f^{out}(\{v\}) - f^{in}(\{v\})\right) = \sum_{u\in A} \left(f^{out}(\{u\}) - f^{in}(\{u\})\right) =$$

ומכיוון שכל הקודקודים ש"בתוך A" מתבטלים, נקבל רק את הסכום של אלה היוצאים או נכנסים לA, כלומר -

$$= f^{out}(A) - f^{in}(A)$$

■

מסקנה 3.9 נביט בטענה הנ"ל כאשר $A = V\backslash\{t\}$.
מכיוון שהנחנו שדרגת היציאה של t היא 0, מתקיים:

$$f^{in}(A) = 0$$

ולכן

$$\nu(f) = f^{out}(A) = f^{in}(\{t\})$$

מסקנה 3.10 לכל זרימה f, מתקיים גם:

$$f^{out}(\{s\}) = \nu(f) = f^{in}(\{t\})$$

(השויון השמאלי הוא ע"פ ההגדרה...)

כעת, נזכור שהגדרנו **קיבול של חתך** ע"י $c(A,B) = \sum_{e\in A\to B} c(e)$**.**

מסקנה 3.11 לכל זרימה f ב־G, ולכל חתך (A,B), מתקיים $\nu(f) \le c(A,B)$.

הוכחה: (למסקנה):

$$\nu(f) = f^{out}(A) - f^{in}(A) \le f^{out}(A) = \sum_{e\in A\to B} f(e) \le \sum_{e\in A\to B} c(e) = c(A,B)$$

■

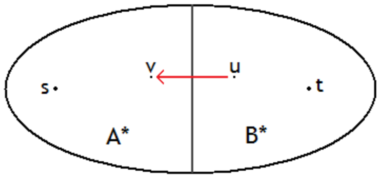
הערה 3.12 נשים לב ששויון מתקיים אם ורק אם:

I. לכל צלע $e : B \to A$ מתקיים $f(e) = 0$.
II.לכל צלע $e : A \to B$ מתקיים $f(e) = c(e)$.

משפט 3.13 תהיה f זרימה ברשת G. ונניח שברשת השיורית G_f אין מסילה מכוונת מ־s ל־t.
אז יש חתך (A^*, B^*) ברשת, כך ש־ $\nu(f) = c(A^*, B^*)$.
בפרט, f היא זרימה בעלת שטף מירבי, ו־(A^*, B^*) חתך בעל קיבול מזערי.
(ניתן לראות שהשורה האחרונה מתקבלת ישירות מאי השויון של המסקנה האחרונה).

הוכחה: נגדיר את A^* כקבוצת כל הקודקודים $v \in V$ כך שב־G_f יש מסילה מכוונות מ־s ל־v. וגם, $B^* = V \backslash A^*$.
ראשית, (A^*, B^*) הוא חתך, כי $s \in A^*, t \notin A^*$ ולכן $t \in B^*$.
על מנת להוכיח את טענתינו, יש שני תנאים שעלינו לבדוק (מההערה לעיל): I, II.

תנאי I: עלינו לוודא שאם $e : B^* \to A^*$ אז $f(e) = 0$.
בציור הבא, $e = (u, v)$, $u \in B^*$, $v \in A^*$.



ע"פ הגדרת הרשת השיורית, אילו היה מתקיים $x = f(e) > 0$, הייתה צלע מכוונת ב־G_f מ־v ל־u שקיבולה x.
נזכור שיש מסילה מכוונת מ־s אל v ב־G_f (זו הסיבה שבגללה $v \in A^*$).
אם כך, יש גם מסילה מכוונת מ־s ל־ u (הולכים מ־s ל־v וממשיכים בצלע (v, u)). וזה בניגוד לכך ש־$u \notin A^*$. ולכן $f(e) = 0$ בהכרח.
תנאי II: יש להראות שכל צלע $e : A^* \to B^*$ רוויה, ז"א $f(e) = c(e)$.
אם נניח אחרת, אז אם $c(e) - f(e) = y > 0$, היינו מוצאים ברשת השיורית G_f צלע מכוונת מ־v ל־u, בקיבול y.
שוב, כמקודם, יש מסילה מכוונת מ־s ל־v, וניתן להאריך אותה ולהגיע גם ל־u, אף כי ע"פ ההנחה $u \notin A^*$. ולכן $f(e) = c(e)$ בהכרח. ■

3.0.7 מסקנות:

1. כאשר אלגוריתם FF עוצר, יש בידינו זרימה אופטימלית.
2. כאשר ברשת G כל הקיבולים שלמים,
   (א) אלגוריתם FF פותר את בעיית הזרימה בזמן $O(m \cdot c)$ (כאשר m מס' הצלעות, c סכום כל הקיבולים).
   (ב) כאשר ברשת G כל הקיבולים שלמים, אז יש זרימה אופטימלית שכל ערכיה הם שלמים (ואלג' FF מוצא זרימה כזו).
3. **משפט השטף והחתך** ($MFMC$)
   לכל רשת זרימה $G = (V, E, s, t, c)$, השטף המירבי האפשרי ב־G שווה לקיבול המזערי של חתך ברשת.
   (כשהקיבולים שלמים ־ הוכחנו את המשפט כבר (?)).

הערה 3.14 עוד כמה מילים על רשתות עם קיבולים לא שלמים.
הטיפול שהצגנו במקרה של קיבולים שלמים פותר גם את הבעיה במקרה של קיבולים רציונאליים (מרחיבים את כל הבעיה במכנה המשותף של הקיבולים).
לטובת המקרה הכללי: אם הקיבולים הם כלשהם, נקרב אתם עד כמה שנרצה ע"י מספרים רציונאליים, נשתמש בגירסה הזו של $MFMC$, וע"י זה נראה ש־

$$\forall \varepsilon > 0 \;\; max\, flow \geq min\, cut - \varepsilon$$

היות שזה נכון לכל $\varepsilon > 0$, המשפט נובע.
(הקטע האחרון הוא "חומר העשרה, לא למבחן", כנראה).

3.0.8 למה $O(mc)$ זה לא פולינומי??????

(שישה סימני שאלה, כמספר האנשים שבאו אל נתי בהפסקה לשאול על זה).
קריטריון עיקרי ליעילות של אלגוריתם הוא שזמן הריצה שלו חסום ע"י פולינום **באורך הקלט**.
לצורך תיאור הגרף, נזדקק ל- $O(m)$ ביטים.
על מנת לרשום את הקיבולים, נדרשים לנו $\sum\lceil log_2(c(e))\rceil$.
האורך הכולל של הקלט הוא $O(m \cdot log(c))$.
אבל $c = 2^{log_2 c}$, כלומר $O(mc)$ זה לא פולינומי!

בהמשך - נציג שני אלגוריתמים נוספים (שהם מימושים שונים של FF).

- אלגוריתם של דירוג ($scaling$): כאן אנו ננסה למצוא תמיד מסילה מ- s ל- t ב- G_f, שערך צוואר הבקבוק שלה גדול ("מסילת הרחבה רחבה").

  כאן נקבל זמן ריצה פולינומי: $O(m^2 log(c))$.

- אלגוריתם של $Edmonds - karp$: נשתמש במסילה הקצרה ביותר בין s ל- t ב- G_f.

  זהו אלג' פולינומי **חזק**, ז"א זמן הריצה שלו איננו תלוי כלל בערכים המספריים המופיעים בו. זמן הריצה של האלג' הספציפי הזה הוא $\geq m^3$.

3.1 שימושים של $MFMC$

3.1.1 קצת על זיווגים (הקדמה למשפט Hall)

הגדרה 3.15 זיווג ($matching$) בגרף הוא אוסף של צלעות זרות.
אנחנו נצטמצם למקרה של גרף דו-צדדי.

גם בהקשר הזה יש אוסף גדול וחשוב של בעיות אופטימיזציה.

1. בהנתן גרף G, מצא את הגודל המירבי של זיווג ב- G.

2. גירסה משוקללת: לכל צלע יש ערך נומרי $w(e) > 0$.

   הבעיה: מצא זיווג שסכום המשקלות של צלעותיו מירבי.

זה שימושי, למשל, אם יש לנו גרף דו-צדדי $G = (A, B, E)$ כך שכל קודקוד ב- A מייצג עובד, וכל קודקוד ב- B מייצג משימה, וצלע בין עובד למשימה משמע שהעובד יודע לבצע את המשימה. בעיית הזיווג המקסימלי כאן פירושה שאנו רוצים שיתבצעו כמה שיותר משימות. אם יש לנו צלע $e = \{x, y\}$, אז $w(e)$ הוא הרווח שמתקבל כשעובד x מבצע משימה y.

נטפל במקרה הפרטי הבא:

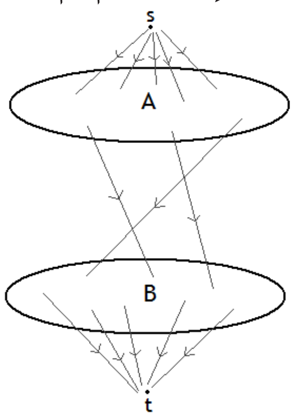
$G = (A, B, E)$ גרף דו-צדדי, ונניח $|A| = |B| = n$.
זיווג מושלם זהו זיווג הפוגש את כל קודקודי הגרף.

- איך נדע אם יש ב- G זיווג מושלם?
- אם יש, איך נמצא אותו?

3.1.2 משפט החתונה של Hall

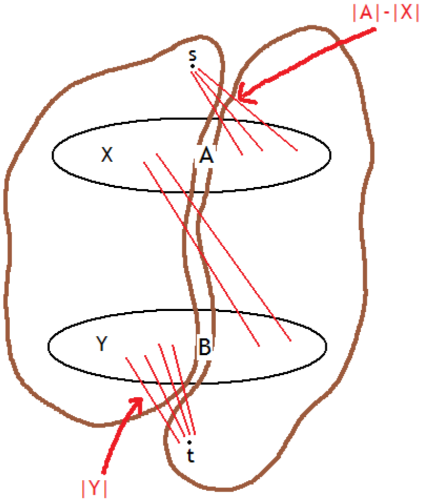
משפט 3.16 בגרף הדו־צדדי $G=(A,B,E)$ יש זיווג מושלם $\Leftrightarrow$ לכל $X \subseteq A$ מתקיים $|\Gamma(X)| \geq |X|$,
כאשר $\Gamma(X)=\{y\in B \mid \exists x\in X\ \{x,y\}\in E\}$ (קבוצת כל השכנים של X. מכך שזהו גרף דו־צדדי, הם כולם ב־B).

מטרתינו: להסיק את משפט החתונה מ־$MFMC$.
הוכחה: כיוון אחד קל וברור. נוכיח את הכיוון השני: נניח שלכל $X \subseteq A$ מתקיים $|\Gamma(X)| \geq |X|$, ונראה שיש זיווג מושלם.
נהפוך את הגרף לרשת זרימה $\mathcal{G}$ כדלקמן:
תהיה צלע מ־s לכל קודקוד ב־A, וצלע מכל קודקוד ב־B ל־t.



כל הצלעות יכוונו $s\to A\to B\to t$ ולכולן קיבול של 1.
היות שמדובר בקיבולים שלמים, יש זרימה אופטימלית בשלמים.
כשמצמצמים את המבט ל־E (כלומר רק הצלעות בין A ו־B), מתברר שמתקבל זיווג ב־$G=(A,B,E)$.
(הקיבולות הן כולן 1 ולכן לא ייתכן שנזרים מקודקוד אחד ב־A שתי צלעות...)
ברור גם שערך הזרימה המירבי $\geq n$ (נזכור: $|A|=|B|=n$ ואנחנו מוציאים לכל היותר $|A|$ קודקודים מ־s)
ויש זיווג מושלם ב־$G$$\Leftrightarrow$השטף המירבי $=n$. (כי אז לכל קודקוד ב־A יהיה זיווג ב־B).
בגלל $MFMC$, יש זרימה עם שטף n $\Leftrightarrow$ לכל חתך ברשת יש קיבול $\leq n$.
אז אנחנו נראה שלכל חתך ברשת יש קיבול $\leq n$, וזה יהיה שקול למה שאנחנו רוצים להוכיח, כמו שראינו זה עתה.

הקבוצה שמכילה את s בחתך כללי היא מהטיפוס: $\{s\}\cup X\cup Y$, כאשר $X\subseteq A,\ Y\subseteq B$.
מהו הקיבול של החתך המתאים? היות שלכל צלע כאן יש קיבול 1, די לנו לספור את הצלעות היוצאות מקבוצה זאת.
נשים לב שהרשת שלנו מורכבת מ־4 "איזורים": s,A,B,t. נספור את הצלעות שיוצאות מכל אחד משלושת האיזורים s,A,B בחתך שלנו:



מ־$\{s\}$ יש לנו $|X| - n = |X| - |A|$ צלעות שיוצאות אל $A \backslash X$.
מ־$A \supseteq X$ יוצאות לפחות (אולי בדיוק?) $|\Gamma(X) \backslash Y|$ צלעות.
ומ־$B \supseteq Y$ אל $\{t\}$ יוצאות $|Y|$ צלעות.
קיבלנו שקיבול החתך הוא לפחות:

$$(n - |X|) + (|Y| + |\Gamma(X) \backslash Y|) \geq (n - |X|) + (\Gamma(X)) \geq n$$

כאשר אי השויון האחרון הוא מההנחה ש־ $|\Gamma(X)| \geq |X|$.
קיבלנו, שאם אכן לכל $A \supseteq X$ מתקיים $|\Gamma(X)| \geq |X|$ אז יש ב־G זיווג מושלם. ■

3.1.3 קצת על תכנון ליניארי

(הערה: אני אישית לא ממש הצלחתי לעקוב אחרי ההרצאה כמו שצריך כאן, ומה שכתוב לי במחברת הוא לא במיוחד קריא. בכל מקרה זה היה כמה פסקאות של הסבר על תכנון ליניארי, ואיך בעיית הזרימה קשורה לזה. לא נראה לי רציני שזה לא מוקלד כאן בכל אופן). (23.11.2009)

3.2 מימושים של Ford-Fulkerson

3.2.1 אלגוריתם מדורג לבעיית הזרימה

זהו אלגוריתם בעל זמן ריצה $O(m^2 log(c))$ כאשר m מס' הצלעות, c סכום הקיבולים. ⇐פולינומי באורך הקלט.
זהו אלג' מטיפוס FF, ובעצם נראה שהוא מבצע לכל היותר $O(mlog(c))$ איטרציות ־ מחיר כל איטרציה $O(m)$.
הרעיון: למצוא ב־G_f מסילה מ־s ל־t שערך צוואר הבקבוק שלה הוא "גדול".
אם זו מטרתינו, עלינו להמנע מצלעות ב־G_f שקיבולן הוא קטן. ננסה להשאיר ב־G_f רק צלעות שקיבולן הוא "גדול".
מאידך גיסא, אסור לנו להציב רף גבוה מידי, כי בזה אנו עלולים לגרום לכך שלא תהיה כלל מסילה מ־s ל־t ב־G_f.
בפועל, אנו נציב את הרף בגובה שהוא חזקה של 2.

סימון: אם Δ מספר חיובי, נסמן ב־ $G_f(\Delta)$ את הרשת המתקבלת מ־G_f ע"י מחיקת כל צלע שקיבולה $\Delta >$.

__האלגוריתם:__

אתחול: הזרימה הריקה $f \equiv 0$. ונתחיל מ־

$\Delta_0 = max\{2^k \mid 2^k \leq$ יש צלע היוצאת מ־s וקיבולה$\}$

(ונגדיר ־ פאזה: פרק הזמן באלג' שבו Δ מקבל ערך נתון).

האיטרציה: מצא מסילות ב־$G_f(\Delta)$ ועדכן את f כל עוד יש ב־$G_f(\Delta)$ מסילה מ־s ל־t.

כשאין: הקטן את Δ ל־ $\frac{\Delta}{2}$ וחזור לאיטרציה.

עצור ב־ $\Delta = 1$.

החלק הברור: האלג' מוצא זרימה אופטימלית, ולו בגלל הפאזה שבה $\Delta = 1$.

טענה 3.17 האלגוריתם מבצע $O(mlog(c))$ איטרציות, ולכן זמן הריצה הכולל שלו $O(m^2log(c))$ (זהו זמן ריצה פולינומי).

עלינו להראות רק שיש $O(mlog(c))$ איטרציות.

ברור שהערך ההתחלתי של Δ הוא $\leq c$, ובכל פעם Δ יורד ל־$\frac{\Delta}{2}$,

ולכן מס' הפאזות (מס' הערכים השונים של Δ) הוא $\leq \lceil log_2 c \rceil + 1$.

נותר להראות עוד, שבפאזה נתונה, בערך קבוע של Δ, יש לכל היותר $2m$ איטרציות.

הסיבה: בסוף "הפאזה ה־Δ", שטפה של הזרימה שקיבלנו רחוק מהאופטימום לכל היותר ב +$m\Delta$. (* הוכחה לכך אח"כ)

היות שאנו עובדים ב־$G_f(\Delta)$, ערך צוואר הבקבוק של כל מסילה שנמצא הוא $\leq \Delta$.

מסוף הפאזה הקודמת, יש לנו פער של $\leq m\Delta$ מהאופטימום, בכל פעם אנו מתקדמים בשטף לפחות כדי Δ (אולי זה $\frac{\Delta}{2}$? לי כתוב במחברת Δ אבל הגיוני שזה חצי. תודה ליצחק על התיקון!), ולכן יש לנו $\leq 2m$ איטרציות.

כעת נוכיח את * הנ"ל ע"י זה שנמצא חתך שקיבולו: (נשים לב, אי השויון הימני ביותר נכון תמיד...)

$$c(A,B) \leq \nu(f) + m\Delta \leq c(A,B) + m\Delta$$

טענה 3.18 תהיה f_Δ הזרימה בתום הפאזה ה־Δ. אזי,

$$\nu(f_\Delta) \geq \nu_{opt} - m\Delta$$

בעצם יש חתך (A,B) כך ש־ $c(A,B) \leq \nu(f_\Delta) + m\Delta$.

הוכחה: לטענה:

תהיה A קבוצת כל הקודקודים v שיש מסילה מכוונת ב־$G_f(\Delta)$ מ־s ל־v (בתום הפאזה ה־Δ). אז $t \notin A$, $s \in A$, $B := V \backslash A$.

($t \notin A$ כי בתום הפאזה ה־Δ אין מסילה בין s ל־t ב־ $G_f(\Delta)$). וכעת,

$$\nu(f_\Delta) = \sum_{e\,goes\,out\,from\,A} f(e) - \sum_{e\,goes\,into\,A} f(e) \geq \sum_{e\,goes\,out\,from\,A} (c(e) - \Delta) - \sum_{e\,goes\,into\,A} \Delta \geq c(A,B) - m\Delta$$

כאשר בביטוי הימני ביותר, $\sum_{e\,goes\,out\,from\,A} c(e)$ הוא שווה ל־$c(A,B)$, ו $m\Delta$ זה בגלל שהפסדנו לכל היותר Δ על כל צלע הנכנסת ל־A או יוצאת ממנה. מס' הצלעות האלה $\leq m$.

נסביר את **המעבר האמצעי:** באי שויון זה, נטענות שתי טענות:

1. אם e יוצאת מ־A, אז $f(e) \geq c(e) - \Delta$ $(*)$.
2. אם e נכנסת ל־A, אז $\Delta > f(e)$.

הוכחת א': נניח ש־ $u \in A$, $u,v \in G_f(\Delta)$, $v \notin A$ ו־ $\overrightarrow{e} = (u,v)$.

אילו אי השויון $(*)$ היה מופר, היתה גם $e \in E(G_f(\Delta))$ ואז היה ניתן גם להגיע מ־s ל־v (הולכים מ־s ל־u וממשיכים ל־v).

הוכחת ב': כנ"ל. ∎

3.2.2 אלגוריתם Edmonds-Karp לזרימה ברשת

זהו אלג' מטיפוס FF עם הכלל הבא:
מצא תמיד ב־G_f מסילה **קצרה ביותר** מ־s ל־t, והשתמש בה לשיפור הזרימה.

הגדרה 3.19 נאמר שברגע מסוים **מיצינו** את הצלע e אם הזרמנו בה אז בדיוק $c(e)$, או שהורדנו את הזרימה ב־e ל־0.

משפט 3.20 אלגוריתם EK פותר את בעיית הזרימה בזמן $O(m^2n)$ (m מס' הצלעות, n מס' הקודקודים).

המשפט נובע אם נוכל להראות שמס' האיטרציות של EK הוא $O(mn)$.
זה נובע משתי הטענות הבאות:

1. המרחק ב־G_f מ־s ל־t אינו יורד אף פעם במהלך הריצה של האלגוריתם.
2. בין שני מיצויים עוקבים של אותה הצלע e, המרחק ב־G_f מ־s ל־t **גדל ממש.**

מסקנה 3.21 (מ־1,2):
ל־EK יש $O(mn)$ איטרציות.
המרחק מ־s ל־t ב־G_f עשוי לגדול לכל היותר מ־1 ל־n (כמספר הקודקודים בגרף...). אחת ל־m צעדים לכל היותר אנו ממצים צלע, ולכן מדי m צעדים המרחק הנ"ל גדל ממש. (הצלע שהיא צוואר הבקבוק במסילה ההרחבה ־ ממוצה).

הגדרה 3.22 אנו נאמר ששתי מסילות P, Q **נמצאות בקונפליקט**, אם יש צלע כלשהי שהן עוברות בה שתיהן במגמות מנוגדות.

טענות 1,2 נובעות מהלמה הבאה:

למה 3.23 יהיו $r < r'$ שתי איטרציות של EK, ויהיו P, Q המסילות שבוחר בהן אלג' EK בזמנים אלה.
ונניח את ההנחות הבאות:
א) P, Q נמצאות בקונפליקט.
ב) אם R מסילה הנבחרת ע"י EK באיזשהו זמן ביניים r'', כלומר $r < r'' < r'$, אז R אינה נמצאת בקונפליקט לא עם P ולא עם Q.
אזי, $|P| < |Q|$.
(הגדרנו $|P|$=מס' הצלעות ב־P).

נראה שטענות 1,2 נובעות מהלמה:

1. נביט בזמנים $r, r' = r + 1$, ונראה שהמרחק לא יורד.

   יהיו P, Q המסילות הנבחרות בזמנים $r, r+1$.

   אם הן אינן בקונפליקט, משמע שגם Q עמדה כאפשרות בזמן r, אבל P נבחרה, ו־EK בוחר תמיד במסילה קצרה ביותר, ולכן $|P| \geq |Q|$.

   לכן המרחק לא ירד.

   ואילו אם P, Q נמצאות בקונפליקט, אז מתקיימות ההנחות של הלמה, ואפילו $|P| < |Q|$.

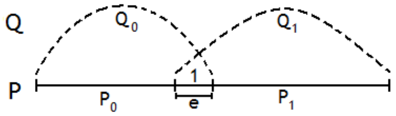
2. נביט בקטע הזמנים $t_1 < t_2$ שבו מוצתה e באופן עוקב.

   הקטע הזה מכיל תת־קטעים שבהם המסילה הראשונה והאחרונה מצויים בקונפליקט.

נביט בתת־קטע מינימלי של t_1, t_2, $[r_1, r_2] \subseteq [t_1, t_2]$, שבתחילתו ובסופו יש מסילות הנמצאות בקונפליקט.
עכשיו אנחנו בתנאי הלמה. (א' ־ ע"פ בחירה, ב' ־ בגלל המינימליות).
יוצא שהמרחק מ־s ל־t **גדל** (ממש) בקטע הזמן r_1, r_2, ולכן גם ב־t_1, t_2 כפי שרצינו. (הרי המרחק בין s ל־t לא יורד לעולם במהלך ריצת האלג').

הוכחת הלמה:

ראשית כל, אילוסטרציה:



Q לאו דווקא עמדה כאופציה לאלגוריתם בזמן r (שבו נבחרה המסילה P) מפני שהצלע $\overleftarrow{e}$ לא היתה ברשת השיורית.
אבל בהחלט עמדה לפני האלגוריתם האופציה למסילה Q_0, P_1 או המסילה P_0, Q_1.
היות ש־ EK בחר ב־P כמסילה קצרה ביותר, מתקיים

$|P| = |P_0| + |P_1| + 1 \leq |Q_0| + |P_1|$

$|P| = |P_0| + |P_1| + 1 \leq |P_0| + |Q_1|$

ולכן, נחבר:

$2(|P_0| + |P_1| + 1) \leq |Q_0| + |Q_1| + |P_0| + |P_1|$

$|P| + 1 = |P_0| + |P_1| + 2 \leq |Q_0| + |Q_1| = |Q| - 1$

ויוצא ש־ $|Q| \geq |P| + 2$ כלומר $|Q| > |P|$ כנדרש.

(נשים לב שדרך חלופית לחישוב הזה היא לראות ש־

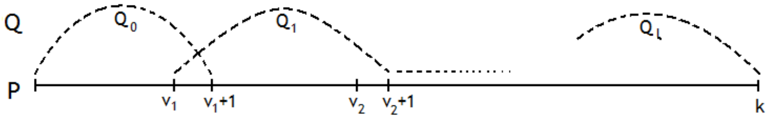
$|P_0| + 1 \leq |Q_0|$

$|P_1| + 1 \leq |Q_1|$

ולכן $|P| + 1 = |P_0| + |P_1| + 2 \leq |Q_0| + |Q_1| = |Q| - 1$)

הוכחה: למקרה הכללי:

יהיו $e_1, ..., e_l$ הצלעות שעליהן יש ל־P ול־Q קונפליקט.
נסמן את הקודקודים של P ב־ $1, ..., k$ (אז $|P| = k - 1$). ונאמר ש $e_j = (v_j, v_j + 1)$ $\forall j$.



בדיוק מאותם שיקולים כמו קודם:

$|Q_0| \geq v_1 + 1$

(המחיר ששילם P=) $|Q_1| \geq v_2 - v_1 + 1$ (=המחיר ששילם Q)

$|Q_2| \geq v_3 - v_2 + 1$

...

$|Q_l| \geq k - v_l$

כאשר נסכם את כל האי שוויונים האלה, יהיה "טלסקופ" (מלשון "טור טלסקופי") מצד ימין, ונקבל:

$$|Q| - l = \sum |Q_i| \geq k + l - 1$$

$$|Q| \geq k + 2l - 1 > |P| = k - 1$$

(כי $l \geq 1$). ∎

4 קירובים לבעיות קשות

4.0.3 הקדמה

החל מתחילת שנות ה־70, ידועות לנו אלפי בעיות אופטימיזציה והכרעה, שאין בידינו אלג' יעיל לפתרונן, אבל ידוע לנו שאם אפילו לאחת מהן יש אלגוריתם בעל זמן ריצה פולינומי, אז יש לכולן.
אוסף זה של בעיות נקרא "בעיות NP שלמות".

הגדרה 4.1 ניזכר במה שאמרנו על בעיות אופטימיזציה:
אנו מחפשים $max_{x\in D}f(x)$, $f > 0$.
אם אנו יודעים למצוא $y \in D$ כך שלכל $x \in D$ מתקיים $f(y) > \frac{f(x)}{A}$,
אז נוכל לומר שיש לנו A**-קירוב** לאופטימום.

לדוגמה:

4.0.4 בעיית הכיסוי הקודקודי ($vc =$ **vertex cover**)

הבעיה: הקלט - גרף $G = (V, E)$.
אומרים ש $S \subseteq V$ היא כיסוי קודקודי של G, אם לכל צלע $xy \in E$ מתקיים $\{x, y\} \cap S \neq \emptyset$.
הבעיה - למצוא כיסוי קודקודי ל־G מעצמה מזערית.

עובדה: בעיית ה־vc היא שלמה ב־NP.
נראה אלגוריתם פולינומי הנותן 2־קירוב לבעיית האופטימיזציה הזו.
ז"א, האלגוריתם מקבל כקלט גרף $G = (V, E)$, הוא מוצא בזמן פולינומי קבוצה $Z \subseteq V$ שהיא כיסוי קודקודי, כך שלכל כיסוי קודקודי אחר S (ובפרט לאופטימלי), מתקיים $|Z| \leq 2|S|$.

נזכיר: זיווג ($matching$) בגרף זהו אוסף של צלעות זרות בגרף.

הערה 4.2 אם M הוא זיווג ב־G, אז $vc(G)$ (כלומר הגודל הקטן ביותר של vc ב־G) מקיים $vc(G) \geq |M|$, כי לכל צלע ב־M חייב להיות לפחות קודקוד אחד בכיסוי הקודקודי, והיות שהצלעות זרות, אי אפשר שאותו קודקוד יהיה ליותר מצלע אחת.

האלגוריתם:
נמצא ב־G זיווג בלתי ניתן להרחבה, ז"א $M^* \subseteq E$ שהוא זיווג, כך שלכל צלע $e \in E$ יש לפחות קודקוד אחד משותף עם איזושהי צלע ב־M^*.
(קל למצוא M^* כנ"ל: בכל צעד מוסיפים איזושהי צלע שאינה פוגשת אף צלע שכבר נבחרה, עד שנעצרים).

מצד אחד, כפי שראינו, $|M^*| \leq vc(G)$.
והפתרון שלנו הוא:

$$S = \{x \in V \,|\, \text{There exists } e \in M^* \text{such that x is a vertex in e}\}$$

כמובן, $|S| = 2|M^*| \leq 2vc(G)$,
ונשים לב ש־S הוא כיסוי קודקודי, מפני שאילו היתה צלע $e = \{u, v\}$ כך ש־ $\{u, v\} \cap S = \emptyset$, היינו מצרפים את e ל־M^*, בניגוד לתכונה ש־M^* הוא זיווג שאינו ניתן להרחבה.

4.0.5 בעיית הכיסוי הקבוצתי (set-cover problem)

נתונה קבוצה X ("קבוצת הבסיס") ואוסף תת־קבוצות שלה $A_1, ..., A_m$ כך ש־ $\bigcup A_i = X$ (לא בהכרח איחוד זר!!).
רוצים למצוא תת משפחה קטנה ביותר $J \subseteq \{1, ..., m\}$ כך שגם $\bigcup_{j \in J} A_j = X$.
("קטנה ביותר" = $min\,|J|$).
זוהי בעיה NP־קשה.

בעיה אחרת, שקולה:
קבוצה פוגעת - Hitting Set.
שוב נתון לנו אוסף של קבוצות $B_1, ..., B_t$, תת־קב' של קבוצה סופית Y (ואף B_j אינה ריקה).
הפלט הרצוי: $Z \subseteq Y$ קטנה ביותר, כך שלכל j מתקיים $B_j \cap Z \neq \emptyset$.
למה הבעיות הן שקולות? נתבונן על זה ככה:
הקלט: נתונה מטריצה שאיבריה $0, 1$. אין בה שורת אפסים ואין בה עמודה של אפסים.
הפלט: מס' קטן ביותר של שורות ה"פוגשות" כל עמודה. ז"א, שבכל עמודה יש לפחות 1 אחד באחת השורות הנבחרות.

נשים לב שאם $S_1, ..., S_l$ הן תת־קבוצות של איזושהי קב' W אז ניתן לייצג זאת ע"י מטריצת החילה של האוסף. כלומר מטריצה M שבה מתקיים

$$M_{ij} = \begin{cases} 1 & i \in S_j \\ 0 & i \notin S_j \end{cases}$$

שתי הבעיות set cover, hitting set ניתנות לייצוג בתור הבעיה עם המטריצה, ולכן הן שקולות.
(למיטב הבנתי, לפחות...)

אלגוריתם קירוב לבעיית ה־SC (set cover):
אלגוריתם חמדן: בכל צעד, נצרך לאוסף שלנו קבוצה A_i המגדילה ככל האפשר את $|\bigcup A_j|$ (כאשר האיחוד הוא עבור כל ה־j שנבחרו עד עתה).

טענה 4.3 האלגוריתם החמדן נותן קירוב של $(1 + ln\,n)$ לפתרון, כאשר $n = |X|$. זאת אומרת,

$$opt \leq cost(greedy) \leq (1 + ln\,n) \cdot opt \quad \spadesuit$$

(אי השויון השמאלי ברור, הימני הוא הטענה).

הוכחה: נסמן ב־ $C(n, k)$ את המחיר המירבי שמשלם האלג' החמדן על פתרון של בעיית הכיסוי הקבוצתי במצב שבו גודל קבוצת הבסיס הוא n ($|X| = n$), והאלג' האופטימלי משלם $\leq k$. (התשלום הוא מס' הקבוצות בפתרון האופטימלי).
אז אנו רוצים להוכיח שמתקיים: $C(n, k) \leq (1 + ln\,n) \cdot k$.
כמובן שמתקיים $C(0, k) = 0$, וכן $C(x, y)$ היא פונקציה עולה בקואורדינטה הראשונה (תמיד אפשר לצרף איברי דמה). וגם עולה בקורד' השניה.
נרצה להוכיח את האי־שויון:

$$C(n, k) \leq C(\lfloor n(1 - \frac{1}{k}) \rfloor, k) + 1$$

נעשה זאת: היות שיש כיסוי ל־X ע"י k קבוצות (זה שניתן ע"י opt [הערה שלי: אם נתון כיסוי נמוך יותר ע"י opt אז ניתן להוסיף לו עוד איברים]), יש קבוצה הכוללת לפחות $\frac{n}{k}$ איברים, ולכן נבחר בקבוצה כזו. [הערה שלי: אם לא היתה קבוצה כזו, אז כל הקבוצות היו מכילות פחות, ולכן סה"כ לא היינו מקבלים כיסוי של כל n האיברים, בסכום של כל ה־k קבוצות...]
ולכן אי השויון לעיל הוא נכון - מפחיתים את הקבוצה הזו ומוסיפים 1 בשבילה (למיטב הבנתי, שוב).

ולמה בכל זאת הוא נכון?! (זה לא נאמר בהרצאה, ייתכנו טעויות).
נסמן ב־φ את גודלה של הקבוצה הגדולה ביותר מבין האפשריות (=את זאת שבה החמדן יבחר ראשון).
אז לפי מה שאמרנו, $C(n,k)=C(n-\varphi,k)+1$.
עוד לפי מה שאמרנו קודם, $\varphi \geq \lceil \frac{n}{k} \rceil$.
ולכן $n-\varphi \leq \lfloor n-\frac{n}{k} \rfloor$.
ומהמונוטוניות:

$$C(n,k)=C(n-\varphi,k)+1 \leq C(\lfloor n-\frac{n}{k} \rfloor,k)+1$$

נשתמש במה שהרגע הוכחנו ונרשום:

$$C(n,k) \leq 1+C(\lfloor n(1-\frac{1}{k}) \rfloor,k) \leq 2+C(\lfloor n(1-\frac{1}{k})^2 \rfloor,k) \leq ... \leq t+C(\lfloor n(1-\frac{1}{k})^t \rfloor,k) \leq$$

כעת, נזכור את האי־שויון הנומרי: $\forall x \in \mathbb{R} \quad 1+x \leq e^x$. נקבל:

$$\leq t+C(\lfloor n \cdot e^{-\frac{t}{k}} \rfloor,k)$$

כעת כל שנותר לחשב, הוא מהו ערכו של t שגורם לכך ש־ $n \cdot e^{-\frac{t}{k}} < 1$.
זאת אומרת:

$$n < e^{\frac{t}{k}}$$

$$ln\, n < \frac{t}{k}$$

$$k \cdot ln\, n < t$$

אז אם נבחר $t=\lceil k \cdot ln\, n+1 \rceil$ [המשך ההוכחה נכתב על־ידי, ייתכנו טעויות:],
נקבל שמתקיים שעבור t זה מתקבל $C(\lfloor n \cdot e^{-\frac{t}{k}} \rfloor,k)=C(0,k)=0$,
ובסה"כ מכל האי שויונים קיבלנו: $C(n,k) \leq t+C(0,k)=\lceil k \cdot ln\, n+1 \rceil$ וזה מספיק כדי להוכיח את ♠ (אם אינני טועה).

■

4.0.6 בעיית הסוכן הנוסע (Travelling Salesman Problem)

נודע גם בשם TSP. זוהי בעיה NP־קשה.
נתון גרף $G=(V,E)$ עם משקלות $0<w$ על הצלעות.
רוצים לעבור באופן מעגלי על כל קודקודי G, על כל קודקוד בדיוק פעם אחת, כך ש־ $\sum w(e)$ ע"פ הצלעות שבהן ביקרנו הוא מזערי.

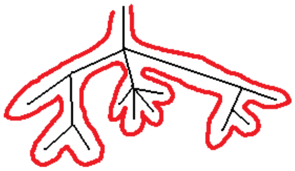
הערה 4.4 אם $P \neq NP$ (דבר שעדיין לא הוכח!), אז אין אף קבוע c כך שלבעיית ה־TSP יש c־קירוב הנמצא בזמן פולינומי.

אנו נצמצם את הדיון ל־ ΔTSP. זהו המקרה של TSP שבו מתקיים אי־שוויון המשולש: לכל 3 קודקודים $x, y, z \in V$,

$$w(x,y) + w(y,z) \geq w(x,z)$$

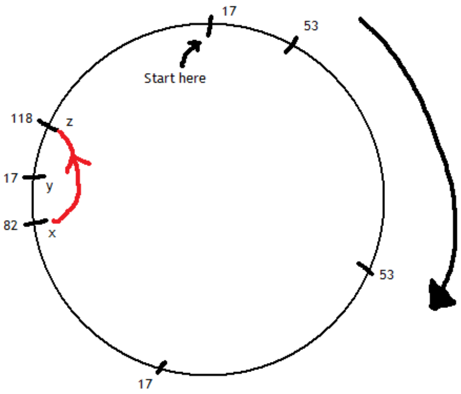
(אם הבנתי נכון, מעתה ואילך אנו מניחים שהגרף הוא שלם, כלומר כל הצלעות האפשריות בו קיימות ־ ואם "אין צלע" בגרף המקורי, נשים משקל ∞ על הצלע הזו).
נראה אלג' פולינומי המקרב את $opt(\Delta TSP)$ עד כדי 2:

ראשית כל נשים לב שמתקיים $opt(\Delta TSP, G) > MST(G)$, מכיוון שפתרון של TSP הוא עץ פורש בעצמו (+צלע), ובנוסף, מכיוון שהוא מעגל, תהיה בו צלע נוספת ש"סוגרת" את המעגל. ולכן אפילו אם קיבלנו עץ פורש מינימלי (+סגירת המעגל) מ־TSP, אי־השוויון הוא תמיד חזק כי עלינו להוסיף את המשקל של הצלע הסוגרת את המעגל.
כעת, אם נסרוק את העץ הפורש המינימלי ואז נחזור לשורש, אז מחיר המהלך הזה הוא $2 \cdot MST(G)$. (מתחילים מקודקוד מסוים, ואז על כל צלע שנבקר בה, ניאלץ לעבור בה שוב בדרך חזרה... הוכחנו את זה באחד התרגילים בדאסט שנה שעברה אם אינני טועה).



Travelling on an MST...

נרצה למצוא דרך יעילה לעבור מהמהלך שלנו על העץ (שעובר בכל קודקוד לאו דווקא פעם אחת), לפתרון מותר של TSP שמחירו $2 \cdot MST(G) \geq$,
וכך נשלם $2 \cdot opt \geq 2 \cdot MST \geq$.



"נפרוש" את העץ הזה למעגל כמו בציור לעיל, ונבצע פעולות של קיצור המסלול (מסומנת אחת לדוגמה באדום) ־ על כל קודקוד שכבר ביקרנו בו נפסח ככה,
ונקבל שהיות ש־ $w(x,z) \leq w(x,y) + w(y,z)$, אז פעולת הקיצור אינה יכולה להגדיל את מחיר המהלך.
כלומר מבצעים קיצורים על פני כל קודקוד שבו כבר ביקרנו, ובגלל א"ש המשולש, אנו איננו מגדילים את מחיר המהלך.

הערה 4.5 ראינו באופן מאוד כללי גם, רעיון לאלג' פולינומי שנותן קירוב פי $\frac{3}{2}$ ל-ΔTSP. לצערי לא ממש הצלחתי לעקוב כאן, ולמרבה המזל נתי גם הוסיף ואמר שזה חומר העשרה ("לא בחומר"), ולכן לא אכלול את הקטע הזה כאן. (הרצאה, 3.12.2009).

4.1 שימוש בהסתברות למציאת פתרונות מקורבים לבעיות קשות

4.1.1 הקדמה - רדוקציה

רדוקציה היא צמצום של בעיה (חישובית) אחת לבעיה אחרת.
אם הוכחנו שבעיה אחת היא קשה (כלומר NP-קשה), אז מכאן ואילך ניתן להראות ששאלה חישובית "חדשה" היא קשה, ע"י זה שמראים שהיא קשה לפחות כמו הבעיה הראשונה.
"בעיה א' קשה לפחות כמו בעיה ב'" - כלומר אילו היה לנו אלגוריתם יעיל לבעיה א', היה לנו גם אלג' יעיל לבעיה ב'. (ובמקרה של NP-קשה, אין כזה שידוע עליו).

4.1.2 3 דוגמאות לבעיות NP-קשות שניתנות לקירוב ע"י שימוש בהסתברות:

1. $max-cut$: הקלט הוא גרף משוקלל $G(V,E)$, $w: E \to \mathbb{R}^+$.
מחפשים את החלוקה $V = A \cup B$ (איחוד **זר**) כך שמתקבל $max \sum_{\substack{x\in A\\ y\in B}} w_{xy}$.

2. $max2lin$: נתונות לנו m משוואות ליניאריות מהצורה $\begin{cases} z_7+z_{19}+z_{24}=0 \\ z_5+z_8+z_{10}+z_{32}=1 \\ \dots \end{cases}$
במקדמי 0,1, ורוצים לפתור את המשוואות מודולו 2. המשתנים z_i מקבלים ערכים 0,1. ספציפית: רוצים לספק מס' מירבי של המשוואות.

3. $max3SAT$: ראשית כל, הסבר קצר על בעיית ה-SAT:

SAT: בעיית הספיקות.
יש נוסחה בצורת CNF $(conjunctive\, normal\, form)$ כלומר $f(x_1,...,x_n)=c_1\wedge c_2\wedge ...\wedge c_m$ כך שכל c_i הוא מהטיפוס, למשל: $c_5=x_7\vee x_{13}\vee \neg x_{28}\vee ...\vee x_{59}$.
בעיית SAT: נתונה נוסחה בצורת CNF, הבעיה היא האם היא ספיקה $(satisfiable)$.
ז"א האם ניתן לתת למשתנים $x_1,...,x_n$ ערכי $\mathbb{T}/\mathbb{F}$ כך ש-$f=\mathbb{T}$, ז"א, כל c_i יקיים $c_i=\mathbb{T}$.
כלומר בכל c_i, לפחות משתנה אחד המקבל ערך $\mathbb{T}$, למשל אם ב-c_i מופיע כאחד מאיברי האיווי: $\neg x_7$, וגם $x_7=\mathbb{F}$, או שמופיע שם x_8 וגם $x_8=\mathbb{T}$.
אם בכל פסוקית c_i יש בדיוק 3 משתנים, אומרים ש-f היא נוסחה ב-$3CNF$, ולבעיית הספיקות המתאימה קוראים $3SAT$.

בעיית $max3SAT$: נתונה נוסחה $f=c_1\wedge ...\wedge c_m$ בצורת $3CNF$, ז"א כל פסוקית c_i היא מהצורה $x_{i_1}^{\varepsilon_1}\vee x_{i_2}^{\varepsilon_2}\vee x_{i_3}^{\varepsilon_3}$, ו-$x_j$ $(1\le j\le n)$ הם המשתנים, וה-ε אומרים שהמשתנה יכול להיות בחיוב או בשלילה.
מחפשים הצבת ערכי אמת למשתנים כך שמס' **מירבי** של פסוקיות יסופקו.

4.1.3 אלג' הסתברותי לקירוב בעיית $max3SAT$

האלג' משיג קירוב של $\frac{7}{8}\cdot opt \le$.
בעצם, אם בפסוק $f=c_1\wedge ...\wedge c_m$ שבצורת $3CNF$ יש m פסוקיות,
אז תוחלת מס' הפסוקיות המסופקות ע"י האלג' שלנו היא $\frac{7m}{8}\le$.
האלגוריתם:
הגרל את ערכי המשתנים:

$$\forall i \qquad Pr(x_i \leftarrow \mathbb{T}) = Pr(x_i \leftarrow \mathbb{F}) = \frac{1}{2}$$

באופן בלתי תלוי על־פני המשתנים השונים.

נראה שזה נותן את הקירוב הרצוי:
המשתנים: $x_1, ..., x_m \in \{\mathbb{T}, \mathbb{F}\}$.
מרחב ההסתברות: $\Omega = \{\mathbb{T}, \mathbb{F}\}^n$, כלומר סדרות של $\mathbb{TTFFTF}...$ כך שכל x_i מקבל את הערך במקום ה־i בסדרה הזו.
ההס' של כל נקודה כזו במרחב: $(\frac{1}{2})^n$.
יש לנו כאן משתנים מקריים: $Y_1, ..., Y_m$ (כמספר הפסוקיות),

$$Y_i : \Omega \to \{0,1\} \qquad\qquad Y_i = (\mathbb{F}, \mathbb{F}, \mathbb{T}, \mathbb{T}, \mathbb{T}, ...) = \begin{cases} 1 & if\, c_i\, is\, satisfied\, under\, these\, \mathbb{T}/\mathbb{F}\, values \\ 0 & otherwise \end{cases}$$

מ"מ נוסף, Y:

$$Y = \sum_{i=1}^{m} Y_i$$

כלומר Y הוא מס' הפסוקיות המסופקות בהצבה זו. וכעת, נחשב:

$$E(Y) = E(\sum_{i=1}^{m} Y_i) = \sum_{i=1}^{m} E(Y_i) = \sum_{i=1}^{m} \frac{7}{8} = \frac{7}{8}m$$

כאשר המעבר השני הוא מליניאריות התוחלת, והמעבר השלישי הוא מכך ש־ $E(Y_i)$ הוא שווה להסתברות שהפסוקית c_i מסופקת ע"י הצבה מקרית.
למה זה $\frac{7}{8}$? למשל, אם $c_i = x_5 \vee \neg x_8 \vee x_{39}$, אז c_i לא תסופק אס"ם $x_5 = \mathbb{F},\ x_8 = \mathbb{T},\ x_{39} = \mathbb{F}$. ההס' שזה ייקרה היא בדיוק $\frac{1}{8}$, ואנו מחפשים את המאורע המשלים לכך, כלומר $\frac{7}{8}$.

טענה 4.6 ההסתברות לספק לפחות $\frac{7m}{8} - 1$ פסוקיות היא $< \frac{c}{m}$ (c קבוע).

הערה 4.7 כשעושים סדרה של ניסוים וסיכוי ההצלחה בניסיון בודד הוא q, אז תוחלת הזמן עד שמצליחים היא $\frac{1}{q}$. (הערה שלי: זו תוחלת של מ"מ בעל התפלגות גיאומטרית. ותודה למוטרו שהביאני עד הלום). לכן , תוחלת הזמן עד שנמצא הצבה המספקת לפחות $\frac{7m}{8} - 1$ פסוקיות היא $O(m)$.

הוכחה: (לטענה): תהיה P_j ההסתברות שסיפקנו בדיוק j פסוקיות. אז:

$$\frac{7m}{8} = E(Y) = \sum jP_j$$

נגדיר: $T = \lfloor \frac{7m}{8} - 1 \rfloor$ וגם נגדיר: ("פסוקית"="$clause$")

$$x := \sum_{T \le j \le m} P_j = Pr(at\, least\, T\, clauses\, are\, satisfied)$$

אז כדי להוכיח את הטענה עלינו להראות ש־ $x \geq \frac{c}{m}$. נחשב:

$$1 - x = \sum_{j=0}^{T-1} P_j$$

$$\frac{7m}{8} = \sum jP_j = \sum_{j<T} jP_j + \sum_{j\geq T} jP_j \leq (T-1)\sum_{j<T} P_j + m\sum_{j\geq T} P_j = (T-1)(1-x) + mx$$

כאשר השתמשנו בחישוב מהשורה אחת אחורה, ובהגדרת x, כדי לקבל את השויון האחרון. כלומר קיבלנו אחרי עוד קצת אלגברה:

$$\frac{7m}{8} \leq T - 1 + x(m - T + 1)$$

$$\frac{7m}{8} - T + 1 \leq x(m - T + 1) \qquad (1 \leq)$$

ומסיבה לא ברורה לי לגמרי, זה אומר ש־ $x \geq \frac{8}{m}$ ובזאת סיימנו. ■

מכיוון שהרבה אנשים (ואני ביניהם) לא מבינים את המעבר האחרון, אלחנן היה נחמד מספיק כדי לשלוח את ההוכחה הבאה שהוא כתב, אז תודה רבה לו (כמו תמיד בהוכחות סטודנטים תיתכן כאן טעות. לי זה נראה נכון, בכל אופן):

נגדיר: x ־ ההסתברות שסיפקנו לפחות $t = \lfloor \frac{7m}{8} - 1 \rfloor$ פסוקיות, P_j ־ ההסתברות שסיפקנו בדיוק j פסוקיות. אז $x = \sum_{j=t}^{m} P_j$.
עכשיו:

$$\frac{7m}{8} = E[Y] = \sum_{j=0}^{m} jP_j = \sum_{0\leq j<t} jP_j + \sum_{t\leq j\leq m} jP_j \leq t \cdot \sum_{0\leq j<t} P_j + m \cdot \sum_{t\leq j\leq m} P_j \overset{(1)}{\leq}$$

$$t \cdot \sum_{0\leq j<\boldsymbol{m}} P_j + m \cdot x \overset{(2)}{=} t + m \cdot x = \lfloor \frac{7m}{8} - 1 \rfloor + m \cdot x \leq \frac{7m}{8} - 1 + m \cdot x \Rightarrow$$

$$\frac{7m}{8} \leq \frac{7m}{8} - 1 + m \cdot x \Rightarrow \frac{1}{m} \leq x$$

(1) ־ הרחבנו את הסכימה בסכום השמאלי עד m וכך אנחנו רק מחברים עוד גורמים אי שליליים, בנוסף החלפנו את $\sum_{t\leq j\leq m} P_j$ ב x לפי ההגדרה.
(2) ־ $\sum_{0\leq j<m} P_j = 1$ בגלל שזו ההסתברות שמספר הפסוקיות המסופקות יהיה בין 0 ל m, וזה כמובן חייב לקרות.

הערה 4.8 אותה שיטת פעולה ואותן תוצאות חלות גם על בעיות ה־ $max - cut, max2lin$.
למשל, עבור $max - cut$, נפצל את הקודקודים לשני חלקים באקראי. ההס' של כל צלע להיות בחתך היא $\frac{1}{2} + O(\frac{1}{n})$, ולכן תוחלת משקל החתך $\leq \frac{|w|}{2}$. (מה זה w?)

5 טרנספורם פורייה

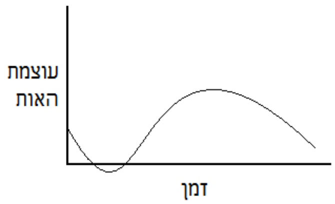
5.0.4 כפל של מטריצה בוקטור

באופן כללי, בעיה של כפל מטריצה בוקטור היא כזו:
הקלט: מטריצה ריבועית $A_{n\times n}$, ווקטור n־מימדי x, והפלט: הוקטור Ax.
ניתן לעשות זאת ב־n^2 פעולות כפל (ואכן, גם אי אפשר לקצר בזמן הזה).

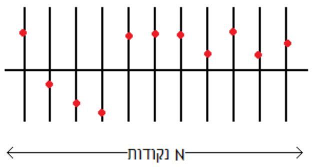
עם זאת, יש סיטואציות רבות שבהן הבעיה החשובה שונה קצת.
קובעים **מראש** מטריצה $A_{n\times n}$.
הקלט: וקטור x.
הפלט: Ax.
נתעניין במטריצות A כאלה שבשבילן יש דרך מהירה יותר לחישוב Ax.

5.0.5 טעימה מיסודות התחום של עיבוד אותות

רעיון חשוב בתחום, הוא לבטא אות שמע כללי כצירוף של סינוסים וקוסינוסים.



בעיה: נתון אות כלשהו, איך מפרקים אותו לצירוף של sin, cos?
נדגום את האות בנקודות מסוימות בזמן ("דיסקרטיזציה") כדי לקבל משהו בדיד, ולא רציף, לעבוד איתו.



(הערה: כמובן ששני האיורים לעיל הם לא קשורים זה לזה...)
אחרי הדיסקרטיזציה של הזמן, האות מתורגם לסדרה של מספרים ממשיים, ורוצים לבטא אותה באמצעות (הגירסה הדיסקרטית של) $sin(kx), cos(lx)$.
העולם שבו אנו עובדים הוא $\mathbb{R}^N$ (N הוא מס' נקודות הדגימה שלקחנו מהאות), והאות f הוא $f\in\mathbb{R}^N$.
גם על $sin(kx)$, $cos(lx)$ אנו חושבים כוקטורים N מימדיים.
הבעיה: לבטא את הוקטור f באמצעות הוקטורים המתאימים ל־ $sin(kx), cos(lx)$.
ע"י מודיפיקציה קלה של הבעיה, מגיעים לכך שהפונקציות $cos(lx), sin(kx)$ (כוקטורים ב־$\mathbb{R}^N$) הם בסיס למרחב ולכן יש לבעיה שלנו פתרון יחיד.
יתר על כן, זהו בסיס אורתונורמלי.
(תזכורת מלינארית 2: בסיס א"נ הוא כזה שכל הוקטורים בו מאונכים זה לזה, כלומר מכפלתם הפנימית היא 0, ובנוסף כל וקטור בו הוא בעל נורמה של 1.)
ויש לנו נוסחאות פשוטות למציאת הייצוג של וקטור נתון בבסיס אורתונורמלי.

אם $u_1, u_2, \ldots$ הם בסיס א"נ למרחב ליניארי V, ואם $z \in V$, אז ניתן לרשום $z = \sum \alpha_i u_i$, ומתקיים $\alpha_i = \langle z, u_i \rangle$.
(הוכחנו זאת באלגברה ליניארית 2....)
אז אנחנו בעצם מחפשים את ה־α־ים. נשים לב שמתקיים כתוצאה מהנ"ל:

$$\begin{pmatrix} - & u_1 & - \\ - & u_2 & - \\ & \vdots & \end{pmatrix} \begin{pmatrix} | \\ z \\ | \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \end{pmatrix}$$

בסופו של דבר, אנו נבטא את f :

$$f(x) \cong \sum_{k=1}^{N} a_k cos(kx) + \sum_{k=1}^{N} l_k sin(kx)$$

ואפשר לעבור לסיגנל אחר:

$$g(x) \cong \sum_{k=1}^{N} a'_k cos(kx) + \sum_{k=1}^{N} l'_k sin(kx)$$

וכו'.
הנקודה העיקרית לצרכינו בדיון:
בעיבוד אותות חשוב לנו מאוד לבצע בצורה יעילה את הפעולה $z \to Az$ כש־ z הוא וקטור הקלט (מתאר את האות שבו אנו עוסקים), ו־A מטריצה קבועה (המתאימה לתיאור הבסיס של sin, cos. נשים לב ש־A תלויה ב־N).

מי היא המטריצה $A = A_N$?
נסמן ב־

$$\omega = e^{\frac{2\pi i}{N}}$$

את שורש היחידה הפרימיטיבי מסדר N.
האיבר ה־ (p,q) במטריצה A הוא $\omega^{pq} = e^{\frac{2\pi ipq}{N}}$.
הפעולה $z \to Az$ נקראת טרנספורם פורייה דיסקרטי (DFT).
הנקודה העיקרית היא שאת הפעולה הזו, DFT, ניתן לבצע ב־ $O(N \cdot log(N))$, (וכמובן $N \cdot log(N) \lll N^2$), ע"י אלגוריתם הנקרא טרנספורם פורייה מהיר (FFT).

5.0.6 פולינומים ופעולות בפולינומים

עכשיו נראה שפעולת ה־DFT היא מועילה ביותר גם בטיפול בפולינומים, והעובדה שיש לנו אלג' מהיר FFT לביצועה מאפשרת לנו יעילות במניפולציה של פולינומים.
פולינום כללי ממעלה n: $P(x) = a_0 + a_1x + \ldots + a_nx^n$.
אם $Q(x) = b_0 + \ldots + b_nx^n$, אזי:
חיבור: $R(x) = c_0 + c_1x + \ldots + c_nx^n \qquad P + Q = R,$ כאשר $\forall i,\ c_i = a_i + b_i$.
כפל: $S(x) = d_0 + d_1x + \ldots + d_nx^n \qquad P \cdot Q = S,$ כאשר $\forall k,\ d_k = a_0b_k + a_1b_{k-1} + \ldots + a_kb_0$.
חיבור של פולינומים ממעלה n עולה $n+1$ פעולות חיבור.
כפל, עולה $\Omega(n^2)$ פעולות חשבון (כשמשתמשים בהגדרה הרגילה לכפל שכזה).

יש **דרך אלטרנטיבית לייצוג של פולינומים:**

אם P פולינום ממעלה $\geq n$, ואם $x_0, ..., x_n$ נקודות ממשיות **שונות**, אזי המספרים הממשיים $P(x_0), ..., P(x_n)$ קובעים את P ביחידות.

ז"א, הנתונים $(x_0, y_0), ..., (x_n, y_n)$ מגדירים פולינום אחד ויחיד ממעלה $\geq n$, שהוא P.

עובדה יסודית על פולינומים: לפולינום ממעלה d יש $\geq d$ שורשים.

$F(x) = a_0 + a_1 x + ... + a_d x^d$. אם t הוא שורש של F, ז"א $F(t) = 0$, אזי F מתחלק ללא שארית ב־ $x - t$:

$F(x) = (x - t) \cdot G(x)$.

מכאן יוצא בקלות שהנתונים $(x_0, y_0), ..., (x_n, y_n)$ אינם יכולים להתאים לשני פולינומים שונים ממעלה $\geq n$:

כי אם לכל i, $\begin{matrix} P(x_i) = y_i \\ Q(x_i) = y_i \end{matrix}$, אז $R(x) = P(x) - Q(x)$ הוא פולינום ממעלה $\geq n$, ומקיים $R(x_i) = 0$ לכל $0 \leq i \leq n$, ז"א יש לו לפחות $n + 1$ שורשים, ולכן $R \equiv 0$.

לפעולה שלוקחת את הנתונים $(x_0, y_0), ..., (x_n, y_n)$ ונותנת את הפולינום המתאים P קוראים **אינטרפולציה**, ויש לכך נוסחאות מפורשות.

כעת, אם שני פולינומים P, Q ידועים לנו בדרך כזו, ז"א יש נק' $z_1, ..., z_m$ ומה שידוע לנו זה $\begin{matrix} P(z_1), ..., P(z_m) \\ Q(z_1), ..., Q(z_m) \end{matrix}$, ואם נגדיר $P \cdot Q = R$, אז כמובן:

$$\forall j,\ R(z_j) = P(z_j) \cdot Q(z_j)$$

ועכשיו מצאנו את פולינום המכפלה (בייצוג של ערכיו בנק' נתונות) ב־n פעולות בלבד!

פעולת המעבר בין ייצוג ע"י מקדמים וייצוג ע"י ערכים היא בדיוק DFT.

5.0.7 DFT

DFT הוא טרנספורם פורייה בדיד ($discrete\, Fourier\, transform$). נסמן מטריצה:

$$A_N = A = \begin{pmatrix} & & \\ & \omega^{pq} & \\ & & \end{pmatrix}_{0 \leq p,q \leq N-1} \qquad \omega = e^{\frac{2\pi i}{N}}$$

וה־DFT הוא הפעולה $A\overrightarrow{a} \longmapsto \overrightarrow{a}$ (a וקטור).

סיבה ראשונה לחשיבותו של ה־DFT:

חושבים על שורות במטריצה A כוקטורים N־מימדיים מעל המרוכבים (ז"א וקטורים ב־$\mathbb{C}^N$).

ככאלה, הם בסיס אורתונורמלי* (אחרי תיקון קל).

אנחנו זוכרים שאם $u_1, ..., u_d$ הם בסיס א"נ למרחב וקטורי V, ואם $v \in V$ אזי ניתן להציג את v באופן יחיד כצירוף של ה־u_i, כלומר $v = \sum \alpha_i u_i$, ובמקרה המיוחד ש־ $\{u_i\}$ מהווים בסיס א"נ, ניתן לחשב את המקדמים α_i כך: $\alpha_i = \langle v, u_i \rangle$ (הוכחנו זאת באלגברה ליניארית 2). ולכן -

$$\begin{pmatrix} - & u_1 & - \\ & \vdots & \\ - & u_d & - \end{pmatrix} \begin{pmatrix} | \\ v \\ | \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{pmatrix}$$

*) הם יהיו בסיס אורתונורמלי אם ניקח $\frac{1}{\sqrt{N}}u_1, ..., \frac{1}{\sqrt{N}}u_d$)
כפי שנראה, השורות של A מקיימות:

$$\langle \text{The } i\text{-th row}, \text{The } j\text{-th row} \rangle = \begin{cases} 0 & i \neq j \\ N & i = j \end{cases}$$

בהקשר הזה, אנו חושבים על a כסדרת דגימות של איזשהו אות, ואת האות הזה אנחנו רוצים לפתח בבסיס של $\begin{matrix} sin\, kx \\ cos\, lx \end{matrix}$ ומוצאים את המקדמים בפיתוח ע"י חישוב של DFT.

נזכיר שבמרחבים וקטוריים מעל $\mathbb{C}$, מכפלה פנימית מוגדרת כך:

$$\langle (a_1, ..., a_n), (b_1, ..., b_n) \rangle := \sum_j a_j \overline{b_j} \qquad (a_j, b_k \in \mathbb{C})$$

נראה שכל שתי שורות במטריצה A ניצבות ולכל שורה u (=השורה ה־p) מתקיים $N = \langle u, u \rangle$:

$$\langle u, u \rangle = \sum_j u_j \overline{u_j} = \sum_{j=0}^{N-1} \omega^{jp} \omega^{-jp} = \sum_{j=0}^{N-1} 1 = N$$

עכשיו, נראה שלכל שתי שורות שונות, המכפלה הפנימית של השורה ה־p וה־r ניצבות. מקבלים:

$$\sum_{j=0}^{N-1} \omega^{jp} \omega^{-jr} = \sum_j \omega^{j(p-r)} = 1 + \omega^{p-r} + \omega^{2(p-r)} + ... + \omega^{(N-1)(p-r)} =$$

זהו טור גיאומטרי, שהאיבר הראשון שלו הוא 1, והמנה הקבועה היא ω^{p-r} (נזכור שהנוסחה לטור שכזה היא $\frac{1-q^N}{1-q}$). לכן,

$$= \frac{1 - (\omega^{p-r})^N}{1 - \omega^{p-r}} = \frac{1 - (\omega^N)^{p-r}}{1 - \omega^{p-r}} = \frac{1 - 1^{p-r}}{1 - \omega^{p-r}} = 0$$

כאשר השתמשנו בחישוב: $\omega^N = (e^{\frac{2\pi i}{N}})^N = e^{2\pi i} = 1$.

- החישוב הזה מלמד אותנו גם מהי המטריצה ההופכית למטריצת DFT. מתקיים:

$$\begin{pmatrix} & & \\ & \omega^{pq} & \\ & & \end{pmatrix}_{0 \leq p,q \leq N-1} \cdot \begin{pmatrix} & & \\ & \omega^{-pq} & \\ & & \end{pmatrix}_{0 \leq p,q \leq N-1} = N \cdot I$$

(I היא מטריצת היחידה ו־N הוא מספר, כמובן!).
זה נכון פשוט מהחישוב שעשינו (או חישוב דומה לו), אם $i \neq j$, אז השורה ה־i במטריצה השמאלית ניצבת לעמודה ה־j במטריצה הימנית, ואם $i = j$ אז מכפלתם הפנימית היא N.

ולכן $\begin{pmatrix} & & \\ & \omega^{pq} & \\ & & \end{pmatrix}^{-1} = \frac{1}{N} \cdot \begin{pmatrix} & & \\ & \omega^{-pq} & \\ & & \end{pmatrix}$.

- DFT^{-1}: נתון הוקטור $\overrightarrow{a} \cdot \begin{pmatrix} & \omega^{pq} & \end{pmatrix} = \overrightarrow{b}$, ורוצים לחשב את a, כלומר: $\overrightarrow{b} \cdot \begin{pmatrix} & \omega^{-pq} & \end{pmatrix} = \overrightarrow{a}$
- אם $A\overrightarrow{a} \mapsto \overrightarrow{a}$ הוא הטרנספורם, אז נהוג לדבר על $\overrightarrow{a}$ כעל האות ב"תחום הזמן" ($time\, domain$), ועל $A\overrightarrow{a}$ (לאחר DFT) בתור האות בתחום התדר ($frequency\, domain$).

5.0.8 עוד על טיפול בפולינומים

ניתן להציג פולינומים בשתי צורות שונות.

ע"י מקדמים		ע"י הערכה ($evaluation$) של P ב־$n+1$ נק'
$P(x) = a_0 + a_1x + ... + a_nx^n$	**אינטרפולציה** $\longrightarrow$	$P(x_0), ..., P(x_n)$
חיבור: זול, $O(n)$	**הצבת ערכים** $\longleftarrow$	כפל, חיבור: שניהם זולים, $\Theta(n)$
כפל: יקר, $\Theta(n^2)$		

להלן טבלה שימושית המסכמת מה אפשר לעשות מכל מצב, לאן אפשר ללכת ובאיזו עלות, וכו':

אם $R(x) = P(x) \cdot Q(x)$, אז:

פלט:		קלט:
R נתון ע"י מקדמים	$\Theta(n^2)$, directly $\longrightarrow$	P, Q נתונים ע"י מקדמים
$\uparrow$ $(O(NlogN),\, via\, FFT)$		$(O(NlogN),\, via\, FFT)$ $\downarrow$
R נתון ע"י ערכו בשורשי היחידה	$\longrightarrow$ $O(n)$	P, Q נתונים ע"י ערכיהם בשורשי היחידה

הערה 5.1 אם $\deg P = d_1$, $\deg Q = d_2$, ו־$R = P \cdot Q$, אז $\deg R = d_1 + d_2$.

על מנת לקבוע את מקדמי R, עלינו לדעת את $R(x)$ ב־ $d_1 + d_2 + 1$ ערכים שונים של x. לשם כך נצטרך לדעת גם את $P(x), Q(x)$ באותן $d_1 + d_2 + 1$ נקודות.

הערכה של פולינומים ו־DFT

נניח ש־P פולינום ממעלה n, $P(x) = a_0 + a_1x + ... + a_nx^n$ (נתון כוקטור $\begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix}$), ויהיו $x_0, ..., x_n$ $n+1$ מספרים ממשיים שונים.

כעת, אנו מתעניינים בלדעת את $P(x_0), ..., P(x_n)$.

ניתן לבטא את ההערכות של P בנק' הנ"ל, ז"א את $P(x_0), ..., P(x_n)$ כדלקמן - נביט במטריצה M כך ש:

$$M \cdot \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} 1 & x_0 & x_0^2 & ... & x_0^n \\ \vdots & & & & \vdots \\ 1 & x_n & x_n^2 & ... & x_n^n \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} a_0 + a_1x_0 + a_2x_0^2 + ... + a_nx_0^n \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} P(x_0) \\ \vdots \\ P(x_n) \end{pmatrix}$$

נשים לב שאם $x_j = \omega^j$, אז המטריצה הנ"ל היא המטריצה של DFT, ובמקרה זה המעבר מייצוג במקדמים לייצוג בערכים איננו אלא DFT.

- הפעולה ההפוכה: בהינתן $(x_0,y_0),...,(x_n,y_n)$, למצוא את הפולינום (היחידי) P ממעלה $\geq n$, כך שלכל i, $P(x_i)=y_i$. (=אינטרפולציה).

ראשית, המטריצה $\begin{pmatrix} & & \\ & x_p^q & \\ & & \end{pmatrix}$, $0\leq p,q\leq n$, היא מטריצת **ון־דר־מונדה**, שהיא לא סינגולרית, ולכן ניתן לבצע גם את פעולת האינטרפולציה ע"י כפל במטריצה (ההופכית של הנ"ל).

- במקרה של שורשי יחידה, ראינו ש־ $\begin{pmatrix} & & \\ & \omega^{pq} & \\ & & \end{pmatrix}^{-1}=\frac{1}{N}\cdot\begin{pmatrix} & & \\ & \omega^{-pq} & \\ & & \end{pmatrix}$ (כאשר $0\leq p,q\leq N-1$).

הערה 5.2 כדאי לציין שיש נוסחה מפורשת (נוסחת האינטרפולציה של לגראנג') הפותרת בעיה זו:

$$P(x)=\sum_k y_k\prod_{j\neq k}\frac{x-x_j}{x_k-x_j}$$

5.0.9 קונבולוציה

הגדרה 5.3 יהיו $F=(F_0,...,F_n)$, $G=(G_0,...,G_m)$ שני וקטורים.
הקונבולוציה שלהם $H=F*G$ גם היא וקטור המוגדר כך:

$$H_k=\sum_j F_j\cdot G_{k-j}$$

(הסכימה על j רצה בכל מקום שבו j מוגדר...)

מקובל לסמן $\hat{f}=Af$ "טרנספורם פורייה של f".
יש פעולה בסיסית ביותר בתחום של עיבוד אותות הנקראת **קונבולוציה**.
דרך פשוטה ביותר להחליש את האפקט של רעש המתווסף לאות שבו אנו מעוניינים זה ע"י פעולת מיצוע פשוטה.
אם אנו מסתכלים על האות $...,f_{-2},f_{-1},f_0,f_1,f_2,...$, אז נרצה לעבור לסדרה $...,h_{-2},h_{-1},h_0,h_1,h_2,...$ המוגדרת כך:

$$h_n:=\frac{1}{4}f_{n-2}+\frac{1}{2}f_{n-1}+\frac{1}{4}f_n$$

השאלה: האם יש דרך נוחה לחשב את h_n מתוך f, גם אם מה שבידינו הוא $\hat{f}$. למען הדיוק: איך ניתן למצוא את $\hat{h}_n$ בהינתן $\hat{f}$.
נביט בדברים כך: $h=f*g$, כאשר $g_0=\frac{1}{4}$, $g_1=\frac{1}{2}$, $g_2=\frac{1}{4}$, וכל שאר ה־g_j הם 0.

השאלה המתבקשת היא, אם כן:
אם ידועים לנו $\hat{f},\hat{g}$, איך נראה טרנספורם פורייה של הקונבולוציה שלהם?
תשובה ־ משפט הקונבולוציה \ זהות פרסוול ־ $\widehat{f*g}=\hat{f}\cdot\hat{g}$.
(f,g וקטורים, $\hat{}$ מציין טרנספורם פורייה, $*$ זה קונבולוציה, וב־$\hat{f}\cdot\hat{g}$ אנו מסמנים וקטור שהקואורדינטה ה־i שלו היא $\hat{f}_i\cdot\hat{g}_i$).
(ללא הוכחה בהרצאה).

	עיבוד אותות	פולינומים
$f\to Af$	DFT. $f\to\hat{f}$. מעבר מתחום הזמן לתחום התדר.	הערכת הפולינום P בשורשי היחידה.
$g\to A^{-1}g$	DFT^{-1}. $\hat{f}\to f$. מעבר מתחום התדר לתחום הזמן.	אינטרפולציה ע"י הערכים בשורשי היחידה.
	קונבולוציה	כפל פולינומים

מה הקשר בין קונבולוציה לכפל פולינומים?

אם $\begin{array}{l} P(x) = a_0 + a_1x + ... + a_nx^n \\ Q(x) = b_0 + b_1x + ... + b_mx^m \end{array}$, ו־ $R(x) = P(x)Q(x) = c_0 + c_1x + ... + c_{m+n}x^{m+n}$,
נשים לב שע"פ ההגדרה של כפל פולינומים נקבל:

$$c_k = \sum a_j b_{k-j} \Longleftrightarrow c = a * b$$

(צד ימין הוא בכתיב וקטורי, כלומר של וקטורי המקדמים).

5.0.10 FFT

נראה עכשיו איך ניתן לחשב את DFT על N שורשי היחידה בזמן $O(NlogN)$.
בדיון שלהלן, $N = 2^t$.
נעשה את החישוב תוך שימוש בנקודת המבט של פולינומים. ז"א, יהיה נתון לנו פולינום $A(x)$ ממעלה $N-1$, ונרצה לחשב את $A(\omega^j)$, כש־ω שורש יחידה N־י, לכל $0 \leq j \leq N-1$.

:DFT
(1) בהינתן קלט x, חשב את Ax
$\Updownarrow$
(2) בהינתן פולינום P חשב את $P(\omega^j)$ לכל $0 \leq j \leq N-1$.

FFT הוא אלגוריתם לביצוע DFT בגירסה הפולינומית (2), בזמן $O(NlogN)$.
האלגוריתם ייתן את משוואת הרקורסיה הבאה, מבחינת סיבוכיות:

$$T(N) \leq c \cdot N + 2T(\frac{N}{2}) \leq c \cdot N + 2(\frac{c \cdot N}{2} + 2T(\frac{N}{4})) = 2c \cdot N + 4T(\frac{N}{4}) = ... = jc \cdot N + 2^jT(\frac{N}{2^j})$$

וכאשר $j = t = log_2N$, מגיעים לכך ש־ $T(N) \leq O(NlogN)$.
הרקורסיה שרשמנו רומזת על כיוון הפעולה הבא:
נפתור שתי בעיות עם פולינומים ממעלה $\frac{N}{2}$, ומשני הפתרונות נשיג פתרון לבעייתינו במחיר נוסף של $O(N)$.
נרצה ליצור מ־P שני פולינומים E, D ־ משמעות השמות היא E עבור זוגיים ($even$), D עבור אי־זוגיים (odd).
אם $P(x) = a_0 + a_1x + a_2x^2 + ... + a_{N-1}x^{N-1}$, אז ניתן לרשום:

$$P(x) = a_0 + a_2x^2 + ... \qquad + \qquad x(a_1 + a_3x^2 + a_5x^4 + ...)$$

אז נגדיר:

$$\begin{array}{l} E(y) = a_0 + a_2y + a_4y^2 + ... \\ D(y) = a_1 + a_3y + a_5y^2 + ... \end{array}$$

וקל לראות שמתקיימת הזהות הפשוטה:

$$P(x) = E(x^2) + x \cdot D(x^2)$$

(כי $E(x^2) = a_0 + a_2x^2 + a_4y^4 + ...$ ובנוסף $D(x^2) = a_1 + a_3x^2 + a_5x^4 + ...$).

נגדיר: $\omega_{r,s}=e^{\frac{2\pi i\cdot r}{s}}$
אז מתקיים $\omega_{r,s}=\omega_{2r,2s}$ (הרחבת שבר פשוטה).
אנחנו רוצים לדעת את $P(\omega_{j,N})$ לכל $0\le j\le N-1$.
לכן עלינו לדעת את $E(\omega_{j,N}^2)=E(\omega_{2j,N})=E(\omega_{j,\frac{N}{2}})$ ואותו דבר לגבי D.
ז"א עלינו לדעת את ערכי E בכל שורשי היחידה מסדר $\frac{N}{2}$.
אז החישוב עולה:

$$P(x)=E(x^2)+x\cdot D(x^2)$$

כאשר $E(x^2),\ D(x^2)$ הם כל אחד $T(\frac{N}{2})$ ומדובר בהערכה של פולי' ממעלה $\frac{N}{2}$ בשורשי היחידה מסדר $\frac{N}{2}$.

הסבר שלי, בשפה שונה, לחישוב שעשינו עם הסימון של $\omega_{r,s}$:
לפי הסימון שבד"כ אנו עובדים איתו, מתקיים: $\omega_{r,s}=\omega_s^r$.
אז החישוב שעשינו הוא זה:

$$\omega_s^r=e^{\frac{2\pi ir}{s}}=e^{\frac{2\cdot\pi ir}{2\cdot s}}=\omega_{2s}^{2r}$$

כעת, המשכנו וקיבלנו:

$$\left(\omega_N^j\right)^2=\omega_N^{2j}=\omega_{\frac{N}{2}}^j$$

5.0.11 שיעור החזרה מה־7.1.2010: עוד על FFT.

ניתן לחשוב על DFT כפונקציה שמקבלת וקטור מקדמים $\overrightarrow{x}\in\mathbb{C}^n$ ומחזירה את $\left(P(1),P(\omega_n),P(\omega_n^2),...,P(\omega_n^{n-1})\right)$ כאשר $P(t)=\sum_{i=0}^{n-1}x_it^i$.

תזכורת: כאשר לוקחים את $1,\omega_n,\omega_n^2,...,\omega_n^{n-1}$ ומעלים אותם בריבוע, מקבלים:

$$1,\omega_{\frac{n}{2}},\omega_{\frac{n}{2}}^2,...,\omega_{\frac{n}{2}}^{\frac{n}{2}-1},1,\omega_{\frac{n}{2}},\omega_{\frac{n}{2}}^2,...,\omega_{\frac{n}{2}}^{\frac{n}{2}-1}$$

(כאשר כמובן $\omega_n=e^{\frac{2\pi i}{n}}$)

חישוב מהיר:
$n=2^k$ ומניחים $t\in\{1,\omega_n,...,\omega_n^{n-1}\}$:

$$P(t)=\sum_{i=0}^{n-1}x_it^i=\sum_{j=0}^{\frac{n}{2}-1}x_{2j}t^{2j}+\sum_{j=0}^{\frac{n}{2}-1}x_{2j+1}t^{2j+1}=\sum_{j=0}^{\frac{n}{2}-1}x_{2j}(t^2)^j+t\cdot\sum_{j=0}^{\frac{n}{2}-1}x_{2j+1}(t^2)^j=P_{even}(t^2)+t\cdot P_{odd}(t^2)$$

כאשר P_{even} הוא הפולינום מדרגה $\frac{n}{2}-1$ עם המקדמים הזוגיים של P, ו־P_{odd} כנ"ל עבור האי־זוגיים.

$$\begin{pmatrix} P(1) \\ P(\omega_n) \\ \vdots \\ P(\omega_n^{n-1}) \end{pmatrix} = \begin{pmatrix} P_{even}(1) \\ P_{even}(\omega_{\frac{n}{2}}) \\ \vdots \\ P_{even}(\omega_{\frac{n}{2}}^{\frac{n}{2}-1}) \\ \hline P_{even}(1) \\ P_{even}(\omega_{\frac{n}{2}}) \\ \vdots \\ P_{even}(\omega_{\frac{n}{2}}^{\frac{n}{2}-1}) \end{pmatrix} + \begin{pmatrix} P_{odd}(1) \\ P_{odd}(\omega_{\frac{n}{2}}) \\ \vdots \\ P_{odd}(\omega_{\frac{n}{2}}^{\frac{n}{2}-1}) \\ \hline P_{odd}(1) \\ P_{odd}(\omega_{\frac{n}{2}}) \\ \vdots \\ P_{odd}(\omega_{\frac{n}{2}}^{\frac{n}{2}-1}) \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ \omega_n \\ \omega_n^2 \\ \vdots \\ \omega_n^{n-1} \end{pmatrix}$$

כאשר $\divideontimes$ פירושו **כפל נקודה־נקודה**. אז קיבלנו,

$$DFT(P) = \begin{pmatrix} DFT(P_{even}) \\ DFT(P_{even}) \end{pmatrix} + \begin{pmatrix} DFT(P_{odd}) \\ DFT(P_{odd}) \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ \omega_n \\ \vdots \\ \omega_n^{n-1} \end{pmatrix}$$

כלומר ב־ $O(n)$ פעולות אפשר לעבור מ־ $DFT(P_{even})$, $DFT(P_{odd})$ ל־ $DFT(P)$.
ולכן, $T(n) = 2T(\frac{n}{2}) + O(n) \Longrightarrow T(n) = O(nlogn)$.

לדוגמה:

$$DFT\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} DFT\begin{pmatrix} 1 \\ 3 \end{pmatrix} \\ DFT\begin{pmatrix} 1 \\ 3 \end{pmatrix} \end{pmatrix} + \begin{pmatrix} DFT\begin{pmatrix} 2 \\ 4 \end{pmatrix} \\ DFT\begin{pmatrix} 2 \\ 4 \end{pmatrix} \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ i \\ -1 \\ -i \end{pmatrix} = \bigstar$$

נחשב:

$$\begin{cases} DFT\begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} DFT(1) \\ DFT(1) \end{pmatrix} + \begin{pmatrix} DFT(3) \\ DFT(3) \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 4 \\ -2 \end{pmatrix} \\ DFT\begin{pmatrix} 2 \\ 4 \end{pmatrix} = \begin{pmatrix} DFT(2) \\ DFT(2) \end{pmatrix} + \begin{pmatrix} DFT(4) \\ DFT(4) \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 6 \\ -2 \end{pmatrix} \end{cases}$$

ולכן,

$$\bigstar = \begin{pmatrix} 4 \\ -2 \\ 4 \\ -2 \end{pmatrix} + \begin{pmatrix} 6 \\ -2 \\ 6 \\ -2 \end{pmatrix} \divideontimes \begin{pmatrix} 1 \\ i \\ -1 \\ -i \end{pmatrix} = \begin{pmatrix} 10 \\ -2-2i \\ -2 \\ -2+2i \end{pmatrix}$$

משפט 5.4 ("משפט הקונבולוציה")
עבור $a,b\in\mathbb{C}^n$ מתקיים:

$$DFT(a*b)=DFT(\overrightarrow{a})\circledast DFT(\overrightarrow{b})$$

כאשר $a*b$ היא הקונבולוציה הציקלית של a,b, ו־$\circledast$ הוא כפל קואורדינטה־קואורדינטה.
ההוכחה היתה בתרגיל...

6 אלגוריתמים בתורת המספרים ובקריפטוגרפיה

6.0.12 הקדמה

מספר טבעי נקרא ראשוני אם הוא מתחלק רק בעצמו וב־1. אחרת הוא נקרא פריק.
עובדה: לכל מספר טבעי יש ייצוג אחד ויחיד כמכפלה של חזקות של ראשוניים. למשל, $1080=2^3\cdot 3^3\cdot 5^1$.

כבר ההגדרות הבסיסיות האלה מעוררות מספר שאלות אלגוריתמיות טבעיות:

- הכרעת ראשוניות: בהנתן מס' טבעי n, להכריע האם n ראשוני או פריק (יש אלג' בזמן ריצה פולינומי לכך. אנחנו נראה אלג' הסתברותי).
- פירוק לגורמים: בהנתן n, הצג אותו כ־ $n=\prod p_i^{e_i}$ כאשר p_i ראשוניים (**לא ידוע** אלגוריתם פולינומי לבעיה זו!).

נזכור שאת המספר n אנו מציגים ב־ $\lceil log_2 n\rceil$ ביטים, ולכן זמן ריצה פולינומי כאן פירושו זמן ריצה $\geq (log\, n)^c$, כאשר $c>0$ קבוע כלשהו.

6.0.13 gcd, lcm

הגדרה 6.1 המכנה המשותף המקסימלי של שני מספרים טבעיים a,b זהו המס' הטבעי הגדול ביותר המחלק הן את a והן את b.
סימונים: אם x,y טבעיים, $x|y$ מסמן "x מחלק את y",
והמחלק המשותף המקסימלי של a,b מסומן $\gcd(a,b)$ (*greatest common divisor*).
באופן דומה, **כפולה משותפת מינימלית:** $\mathbf{lcm}(a,b)=\min\{x\in\mathbb{N}\mid a|x,\, b|x\}$.

יהיו a,b טבעיים, ונרשום $a=\prod p_i e^i$, $b=\prod p_i^{f_i}$ (אנו מרשים שיהיו אפסים בחזקות, אז זה מטפל בבעייתיות של הצגה עם אותם p_i...).

נניח $x=\prod p_i^{\alpha_i}$, $y=\prod p_i^{\beta_i}$. אז,

$$\forall i\, \alpha_i\leq\beta_i \;\Leftrightarrow\; x|y$$

והמנה:

$$\frac{y}{x}=\prod p_i^{\beta_i-\alpha_i}$$

אז,
אם $c=\gcd(a,b)$, $c=\prod p_i^{t_i}$, אז לפי מה שאמרנו, על מנת ש־ c יחלק גם את a וגם את b, צריך להתקיים לכל i, $t_i\leq e_i$ כדי ש־ $c|a$, וגם $t_i\leq f_i$ כדי ש־ $c|b$.
ואנחנו גם רוצים ש־ c יהיה גדול ככל האפשר בנסיבות אלה, ז"א ש־ t_i יהיה מירבי.

יוצא ש־ $t_i = \min(e_i, f_i)$.
כלומר:

$$\gcd(a,b) = \prod p_i^{\min(e_i,f_i)}$$

ובאופן דומה,

$$\text{lcm}(a,b) = \prod p_i^{\max(e_i,f_i)}$$

משפט 6.2 $\gcd(a,b)$ זהו המס' החיובי הקטן ביותר s מהצורה $s = ax + by$, כאשר x, y שלמים.

מכאן נעבור לדיון באלגוריתם אוקלידס ל־gcd. (בתרגול...)

6.1 שדות, חבורות (Fields, Groups)

הגדרה 6.3 שדה: זוהי קבוצה $\mathbb{F}$ עם שתי פעולות: $\circ$ ("כפל") ו־ $+$("חיבור"), כך שמתקיימות התכונות הבאות:

1. חיבור וכפל הם אסוציאטיביים וקומוטטיביים: $x \circ y = y \circ x$ (קומוטטיביות לכפל), $(x+y)+z = x+(y+z)$ (אסוצ' לחיבור), וכו'.
2. קיום של איבר יחידה כפלי ואיבר יחידה חיבורי. יש איבר $1 \in \mathbb{F}$ ואיבר $0 \in \mathbb{F}$ המקיימים:

$$\forall x \in \mathbb{F} \qquad 1 \circ x = x, \qquad 0 + x = x$$

3. הופכי: לכל $x \in \mathbb{F}$ יש y (המסומן $y = -x$) כך ש־ $x + y = 0$, ולכל $x \neq 0$ יש $z \in \mathbb{F}$ (המסומן $z = x^{-1}$) כך ש־ $x \circ z = 1$.
4. החוק הדיסטריביוטיבי: $x \circ (y+z) = x \circ y + x \circ z \qquad \forall x, y, z \in \mathbb{F}$

דוגמאות:
$\mathbb{F} = \mathbb{Q}$ המס' הרציונליים
$\mathbb{R}$ הממשיים
$\mathbb{C}$ המרוכבים
$\mathbb{F} = \mathbb{F}_p$ המספרים מוד' p, כאשר p ראשוני.

שאלה: האם יש שדות סופיים מלבד $\mathbb{F}_p$?
התשובה ניתנה ע"י $Galois$: לכל p ראשוני ולכל $n \geq 1$ טבעי, יש שדה אחד ויחיד ובו p^n איברים. מקובל לסמן את השדה הזה ב־ $GF(p^n)$. אין עוד שדות סופיים מלבד אלה.

הגדרה 6.4 חבורה: זו קב' G עם פעולה $\circ$ המקיימת:

1. הפעולה אסוציאטיבית: $(x \circ y) \circ z = x \circ (y \circ z)$
2. יש איבר יחידה e, המקיים $e \circ x = x \qquad \forall x \in G$
3. קיום הופכי: לכל $x \in G$ יש איבר y (המסומן $y = x^{-1}$) כך ש־ $x \circ x^{-1} = e$.

נתחיל בכמה דוגמאות מוכרות:

- חיבור מוד' n. $G=\{0,...,n-1\}$ והפעולה היא חיבור מודולו n. (נסמן את הפעולה ב־"+"). שימו לב שזוהי חבורה קומוטטיבית, ז"א כאן (אבל לא בהכרח בכל חבורה) מתקיים $x+y=y+x \quad \forall x,y\in G$.

נציין שחבורות קומוטטיביות סופיות ניתנות לתיאור מאוד מספק: "משפט המבנה של חבורות קומוטטיביות סופיות".

- דוגמא לחבורות לא קומוטטיביות:

  S_n - החבורה הסימטרית מסדר n.

  תמורה (permutation) על $[n]=\{1,...,n\}$ זו העתקה חח"ע ועל $\pi:[n]\to[n]$.

  כקבוצה, S_n כוללת את כל התמורות של $[n]$. הפעולה: **הרכבה** של העתקות.

  למשל, עבור $n=3$, אם נתבונן בשתי תמורות, $\pi:\begin{cases}1\mapsto 2\\ 2\mapsto 1\\ 3\mapsto 3\end{cases}$ $\tau:\begin{cases}1\mapsto & 3\\ 2\mapsto & 2\\ 3\mapsto & 1\end{cases}$ אז $\tau\circ\pi:\begin{cases}1\mapsto & 2\\ 2\mapsto & 3\\ 3\mapsto & 1\end{cases}$ אבל

  $\pi\circ\tau:\begin{cases}1\mapsto & 3\\ 2\mapsto & 1\\ 3\mapsto & 2\end{cases}$ כלומר זה לא קומוטטיבי!

  איבר היחידה זו תמורת הזהות: $Id:\begin{cases}1\mapsto & 1\\ 2\mapsto & 2\\ 3\mapsto & 3\end{cases}$.

  החבורה S_3 אינה קומוטטיבית.

  לגבי אסוציאטיביות - ברור שהיא מתקיימת.

  ההופכי של π זה π^{-1}.

- ראינו שכאשר p ראשוני, אז המספרים $mod\, p$ מהוים שדה (כל זאת ועוד ראינו והוכחנו לעומק באלגברה ליניארית 1).
- ראינו שלכל n, המספרים מוד' n הם חבורה קומוטטיבית יחסית ל<u>חיבור</u>.

שאלה: מה בדבר המספרים $mod\, n$ (כש־ n פריק) יחסית לפעולת <u>הכפל</u>?

כאשר n פריק ורוצים למצוא מבנה של חבורה יחסית לפעולת הכפל במספרים $1,...,n-1$ (ודאי שיש לנפות את 0), מתעורר הקושי הבא: ייתכן בהחלט ש־ $a\cdot b\equiv 0\,(mod\, n)$.

אף כי $\begin{matrix} a\not\equiv 0\,(mod\, n)\\ b\not\equiv 0\,(mod\, n)\end{matrix}$, יוצא שהקבוצה הנ"ל אינה סגורה לכפל (כי את 0 כבר ניפינו).

6.1.1 החבורה $\mathbb{Z}_n^*$

הגדרה 6.5 אומרים שהמספרים הטבעיים a,b הם **זרים** (Relatively prime) אם $\gcd(a,b)=1$.
ז"א המס' הטבעי היחיד המחלק הן את a והן את b הוא 1.
בהינתן $2\le n\in\mathbb{N}$, נסמן ב־ $\mathbb{Z}_n^*$ את אוסף המספרים $1\le a\le n-1$ **שזרים** ל־n.
הקב' $\mathbb{Z}_n^*$ עם פעולת הכפל מוד' n היא חבורה.
זו חבורה קומוטטיבית, ואיבר היחידה הוא 1.

לדוגמה, עבור $n=30$, $\mathbb{Z}_{30}^{*}=\{1,7,11,13,17,19,23,29\}$.
נקבל את הטבלה הנחמדה הבאה:

	1	7	11	13	17	19	23	29
1	1	7	11	13	17	19	23	29
7	7	19	17	1	29	13	11	23
11	11	17	1	23	7	29	13	19
13	13	1	23	19	11	7	29	17
17	17	29	7	11	19	23	1	13
19	19	13	29	7	23	1	17	11
23	23	11	13	29	1	17	19	7
29	29	23	19	17	13	11	7	1

טענה 6.6 לכל שני מספרים טבעיים a,b יש x,y שלמים כך ש- $\gcd(a,b)=ax+by$. (ללא הוכחה כרגע)

טענה 6.7 $\mathbb{Z}_n^*$ היא אכן חבורה.

הוכחה: נשים לב שאם $1\le a<n$ זר ל-n, וגם $1\le b<n$ זר ל-n, אז גם ab זר ל-n. ולכן גם $ab_{mod\,n}$ זר ל-n, כלומר יש סגירות לפעולה.

- החלק השני הוא ברור: אם $x>n$ זר ל-n אז גם $x_{mod\,n}$ זר ל-n (מפני ש- $x\,mod\,n=x-kn$ ל-k מתאים ולכן אילו היה מחלק משותף $1<$ ל-n,x אז הוא מחלק גם את $x-kn$).
- הוכחה שאם a,b זרים ל-n אז גם ab זר ל-n:

מתי שני מספרים u,v הם זרים: אם רושמים $\begin{matrix}u=\prod p_i^{e_i}\\ v=\prod p_i^{f_i}\end{matrix}$ אז u,v זרים $\Leftrightarrow$ לכל i, או ש- $e_i=0$ או ש- $f_i=0$.

$\Leftarrow$ ולכן, אם n זר ל-a,b ואם $\begin{matrix}a=\prod p_i^{\alpha_i}\\ n=\prod p_i^{\beta_i}\\ b=\prod p_i^{\gamma_i}\end{matrix}$ כלומר $ab=\prod p_i^{\alpha_i+\gamma_i}$,

אז $\begin{cases}\alpha_i\beta_i=0 & \forall i\\ \beta_i\gamma_i=0 & \forall i\end{cases}\Leftarrow\quad \beta_i(\alpha_i+\gamma_i)=0\quad\forall i$ ולכן ab ו-n זרים כנדרש (כי מתקיים התנאי).

קיום של הופכי:
אם a,n זרים אז יש x,y שלמים (הוכחה בהמשך) כך ש- $ax+ny=1$ ($\Leftrightarrow ax\equiv 1_{(mod\,n)}$).
ולכן $x\,mod\,n$ הוא ההופכי של a מוד' n.

- מדוע $x\,mod\,n\in\mathbb{Z}_n^*$ $\Leftrightarrow$ ז"א מדוע $x\,mod\,n$ זר ל-n $\Leftrightarrow$ מדוע x זר ל-n?

כי אחרת, אם קיים t כך ש- $t|n$, $t|x$ וגם $t>1$ (כלומר אם נניח בשלילה שהם לא זרים), אזי $t|(ax+ny)$ ומאידך גיסא $ax+ny=1$, לכן זה לא ייתכן (כי $t>1$ לפי הנחת השלילה).

לכן זוהי אכן חבורה.

■

טענה 6.8 יהיו $a,b \geq 1$ מספרים טבעיים, ונביט בקב'

$$S = \{xa + yb \mid x, y \in \mathbb{Z},\ xa + yb \geq 1\}$$

אזי, $\gcd(a,b) = \min S$.
ובפרט, אם a,b זרים אז יש $x,y \in \mathbb{Z}$ כך ש־ $xa + yb = 1$.

הוכחה: נסמן: $\gcd(a,b) = t$. אז נשים לב ש־t מחלק כל איבר ב־S ולכן $t \leq \min S$.
נותר להוכיח את האי שויון ההפוך, ונסיים.
נסמן ב־z את $\min S$.
אם $\begin{matrix} z|a \\ z|b \end{matrix}$ אז $z|\gcd(a,b)$ ובפרט $z \leq t$. (ובמקרה זה גמרנו).
נניח לכן, למשל, ש־ $z \nmid a$. ואז נביט במספר $z > (a \bmod z) > 0$.
אבל, המספר $a \bmod z$ שייך ל־S גם הוא:
כי $a \bmod z = a - kz$ לאיזשהו k טבעי, אבל $z = ax + by$ ולכן

$$a \bmod z = a - k(ax + by) = a \cdot (1 - kx) - b \cdot ky = x'a + y'b \in S$$

קיבלנו אם כן, איבר ב־S שהוא קטן מ־z בסתירה למינימליות z.
(הערה שלי: נשים לב שלא היינו יכולים לבצע את אותו מהלך במקרה הראשון שבו $\begin{matrix} z|a \\ z|b \end{matrix}$, כי שם היינו מקבלים למשל ש־ $a \bmod z = 0$, ולכן $a \bmod z \notin S$, כי $\forall i \in S,\ i > 0$ לפי הגדרת S). ■

- **מהו גודלה של החבורה $\mathbb{Z}_n^*$?** כלומר, כמה מספרים $1 \leq a \leq n-1$ הם זרים ל־n?
  התשובה ניתנת ע"י **פונקציית אוילר** φ, וזה הביטוי:

$$\varphi(n) = n \cdot \prod_{\substack{p_i \text{ is a prime} \\ p_i | n}} (1 - \frac{1}{p_i})$$

לדוגמה: $n = 30$. אז $\mathbb{Z}_{30}^* = \{1,7,11,13,17,19,23,29\}$, $|\mathbb{Z}_{30}^*| = 8$.

$$8 = 30 \cdot \left((1 - \frac{1}{2})(1 - \frac{1}{3})(1 - \frac{1}{5}) \right)$$

ההוכחה של נוסחת אוילר היא מקרה פרטי של נוסלת ההכלה וההדחה (שלומדים בקורס מתמטיקה דיסקרטית).

נוסחת ההכלה וההדחה:
אם $A_1, ..., A_k \subseteq X$ תת־קבוצות, אז:

$$|X \setminus (\bigcup A_i)| = |X| - \sum_i |A_i| + \sum_{i<j} |A_i \cap A_j| - \sum_{i<j<l} |A_i \cap A_j \cap A_l| \pm ....$$

כאן: $X=\{1,...,n\}$, $A_i=\{x\in X\,|\,p_i|x\}$, ו־ p_i הוא הראשוני ה־i המחלק את n.
וכעת,

$$\prod_{i=1}^{k}(1-t_i)=(1-t_1)(1-t_2)...(1-t_k)=1-\sum_i t_i+\sum_{i<j}t_it_j-\sum_{i<j<l}t_it_jt_l\pm...$$

ואם נחזור למקודם:

$$\varphi(n)=n\cdot\prod_{\substack{p_i \text{ is a prime}\\ p_i|n}}(1-\frac{1}{p_i})=n\cdot\left(1-\sum_i\frac{1}{p_i}+\sum_{i<j}\frac{1}{p_ip_j}-\sum_{i<j<l}\frac{1}{p_ip_jp_l}\pm...\right)$$

וזה מסתדר איכשהו. יש הוכחה מלאה בספר של נתי, שלמדנו איתו לקורס בדיסקרטית, אם מישהו מעוניין.

6.1.2 תת־חבורות

הגדרה 6.9 תהיה $(G,\circ)$ חבורה סופית.
אומרים ש־$(H,\circ)$ היא **תת־חבורה** של G אם $H\subseteq G$ ו־ $(H,\circ)$ כשלעצמה היא חבורה.

דוגמאות:

- חיבור מוד' n $(G,+_{mod\,n})$ כאשר n פריק. נאמר $n=14$. אם נביט ב־ $G=\{0,1,...,13\}$ אז $\{\{0,2,4,6,8,10,12\},+_{mod\,14}\}$ תת־חבורה.
- ראינו שאוסף התמורות של $\{1,...,n\}$ עם פעולת ההרכבה זו חבורה (הנקראת S_n ־ החבורה הסימטרית מסדר n). אז $H=\{\pi\in S_n\,|\,\pi(n)=n\}$ זוהי חבורה.

  H היא תת־חבורה מכיוון ש:

  - סגירות להופכי: אם $\pi\in H$ אז גם $\pi^{-1}\in H$ ($\pi^{-1}(\pi(n))=n\Leftrightarrow$).
  - סגירות לפעולה: אם $\pi,\sigma\in H$ אז גם $\pi\circ\sigma\in H$ ז"א $\pi(\sigma(n))=n$.
- נחזור שוב ל־ $G=\mathbb{Z}_{30}^*$. נחפש תת חבורה $H\subseteq G$ הכוללת את 7.

  $$1,7,7^2=19,7^3=13,7^4=1$$

  $\{1,7,13,19\}$ היא תת־חבורה של $\mathbb{Z}_{30}^*$, "התת־חבורה הנוצרת ע"י 7".

זו תופעה כללית:
אם $(G,\circ)$ חבורה סופית, ו־$a\in G$, ניתן לדבר על התת חבורה הנוצרת ע"י a:

$$H=\{a\circ...\circ a=a^k\ |\ k\in\mathbb{Z}\}$$

זו תת־חבורה: $a^k\circ a^l=a^{k+l}$ וגם $(a^k)^{-1}=(a^{-1})^k$.

הגדרה 6.10 קוסט ($coset$):
תהי G חבורה סופית. אם $H \subseteq G$ תת־חבורה ואם $a \in G$, מגדירים:

$$aH = \{a \circ h \mid h \in H\}$$

משפט 6.11 משפט לגראנז':
אם G חבורה סופית, H תת־חבורה שלה, אז $|H|$ מחלק את $|G|$.

הוכחה: נראה שכל הקוסטים הם זהים או זרים ובסה"כ כל הקוסטים השונים של H נותנים חלוקה של G והמסקנה נובעת.
קונקרטית, אנו טוענים:

1. לכל $a, b \in G$, או ש־ $aH = bH$, או ש־ $aH \cap bH = \emptyset$.
2. כל איבר ב־G שייך (בדיוק) לאחד הקוסטים.

ברור שמטענות אלה נובע המשפט ("וכמובן הגודל של כל קוסט הוא H" ־ הוכחה עוד רגע), כי:
$|G| = |H| \cdot$ (מס' הקוסטים השונים זה מזה)

- מדוע $|aH| = |H|$? כי נתאים באופן חח"ע $h \longleftrightarrow ah$. עלינו רק להראות שלא ייתכן כי $ah_1 = ah_2$ לאיזשהו $h_1 \neq h_2 \in H$. ואכן, אם נכפיל משמאל ב־a^{-1} את $ah_1 = ah_2$, נקבל $h_1 = h_2$ כנדרש.

הוכחת 1:
טענה זו אומרת שאם $aH \cap bH \neq \emptyset$ אז $aH = bH$.
נניח ש־ $x \in aH \cap bH$, ז"א $x = ah_1 = bh_2$ כאשר $h_1, h_2 \in H$. קיבלנו: $\Leftarrow$ $b = ah_1h_2^{-1}$.
ונרצה להראות: $bH \subseteq aH$. (הכיוון השני סימטרי, כנראה).
איך נראה איבר כללי ב־bH? תשובה: $b \cdot h$, כלומר $\left(ah_1h_2^{-1}\right) \cdot h$ (לפי מה שקיבלנו לעיל).
ואם נסמן $h_3 = h_1h_2^{-1}h$ אז האיבר הכללי ב־bH שלנו הוא $a \cdot h_3$, כלומר $b \cdot h = a \cdot h_3 \subseteq aH$ ולכן הראינו את הנדרש.
הוכחת 2:
בהנתן $x \in G$ עלינו למצוא איזשהו $a \in G$ כך ש־ $x \in aH$. ז"א בהינתן x עלינו למצוא $a \in G$, $h \in H$ כך ש־ $x = a \cdot h$.
נבחר $h \in H$ כרצונינו ונגדיר: $a = x \cdot h^{-1}$. ואז כמובן $x = a \cdot h$ כפי שרצינו.

■

נחזור שוב לתת־חבורה הנוצרת ע"י איבר יחיד $a \in G$: $1, a, a^2, \ldots$.
יש איזשהו אינדקס k מזערי כך ש־ $a^k = 1$.

הגדרה 6.12 ל־k הזה קוראים **הסדר של** a ($ord(a)$) בחבורה.

גודל התת־חבורה הנוצרת ע"י a הוא בדיוק $ord(a)$.
ממשפט לגראנז' נובע:

מסקנה 6.13 מסקנה א': תהיה G חבורה ו־a איבר בה. אזי $ord(a)$ מחלק את $|G|$.

הוכחה: (הוכחה שלי, תיתכן טעות): אמרנו שאם H החבורה הנוצרת ע"י a אז $|H| = ord(a)$. ולפי משפט לגראנז', $|H|||G|$. אז קיבלנו את הנדרש: $ord(a)||G|$. ■

מסקנה 6.14 מסקנה ב': **המשפט הקטן של פרמה ($Fermat$)**
לכל p ראשוני ולכל $1 \leq a \leq p-1$ מתקיים $a^{p-1} \equiv 1_{mod\,p}$.

הוכחה: (למסקנה ב'): היות ש- p ראשוני, אז $\mathbb{Z}_p^* = \{1, ..., p-1\}$.
ז"א גודלה של החבורה הכפלית $\mathbb{Z}_p^*$ הוא $p-1$.
לכל $a \in \mathbb{Z}_p^*$, מתקיים $a^{ord(a)} = 1_{mod\,p}$ (ע"פ ההגדרה של סדר).
וע"פ מסקנה א', $ord(a)|p-1$ ז"א ניתן לרשום $p-1 = t \cdot ord(a)$.
ניקח את השויון $a^{ord(a)} \equiv 1_{mod\,p}$ ונעלה אותו בחזקת t, ונקבל:

$$a^{t \cdot ord(a)} = a^{p-1} = 1^t = 1_{mod\,p}$$

■

לדוגמה: $a = 2$, $p = 17$. אז $2^{16} = (2^4)^4 = (16)^4 \equiv (-1_{mod\,17})^4 = 1_{mod\,17}$.

6.2 RSA ומציאת ראשוניים

6.2.1 כמה מילים על מספרים ראשוניים

כשמביטים בסדרה של המס' הראשוניים, קשה לראות תבניות ברורות. הקושי הזה הוביל למחקר ארוך ולבעיות פתוחות רבות.
למשל (בעיה אסימפטוטית): כש-n הוא גדול, כמה מס' ראשוניים יש בין 1 ל-n?
או: כאשר $n \gg \Delta \gg 1$, כמה ראשוניים יש בין n ל-$n+\Delta$?

6.2.2 משפט המספרים הראשוניים (PNT)

משפט 6.15 נסמן ב-$\pi(N)$ את מספר המספרים הראשוניים בין 1 ל-N.
אזי $\pi(N) = (1 + o(1)) \frac{N}{ln(N)}$.

אחת הבעיות הפתוחות הידועות ביותר במתמטיקה: **השערת רימן:** לכל $\varepsilon > 0$ ולכל N גדול דיו, יש בהכרח מספר ראשוני בקטע $[N, N + N^{\frac{1}{2}+\varepsilon}]$.

6.2.3 RSA

(הערה: אין כאן באמת תיאור של RSA. כדאי לקרוא על האלגוריתם עצמו בספר של קורמן).

בוחרים שני מס' ראשוניים גדולים p, q.
מפרסמים את $n = p \cdot q$ ושומרים את p, q לעצמנו.

מתעוררת השאלה: איך ניתן למצוא מס' ראשוניים גדולים מאוד (בני מאות ספרות)?
ממשפט המספרים הראשוניים PNT - די אם יהיה בידינו אלג' יעיל להכרעת ראשוניות (="מבחן ראשוניות").
קלט: מס' n.
פלט: האם n ראשוני או פריק.
PNT אומר (?) שכשבוחרים באקראי מס' בעל k ספרות, ההסתברות לכך שיהיה ראשוני היא $\Theta(\frac{1}{k})$.
לכן הבעיה שבה נתעסק לעומק היא מבחן ראשוניות.
נציג אלג' של $Rabin - Miller$ שהוא אחד האלג' ההסתברותיים הראשונים שפותחו.

6.2.4 תזכורת קצרה לגבי מה שעשינו עד כה (ועוד קצת דברים חדשים):

ראינו מה זה חבורה, תת־חבורה, קוסט. הוכחנו את משפט לגראנז': אם G חבורה סופית ו־H תת־חבורה שלה, אז $|H|$ מחלק את $|G|$.

חבורה שבה התעסקנו לא מעט היא $\mathbb{Z}_n^*$, החבורה הכפלית של המס' הזרים ל־n. ראינו גם שגודלה הוא:

$$|\mathbb{Z}_n^*| = \varphi(n) = n \cdot \prod_{\substack{p \text{ is a prime} \\ p|n}} (1-\frac{1}{p})$$

מבחן הראשוניות שאותו נציג בהמשך הוא פולינומי, כלומר, זמן הריצה שלו על קלט n הוא $(log(n))^c$ עבור קבוע כלשהו c.

בנוסף: אם G חבורה, $a \in G$, אז **הסדר** של a ($ord(a)$) זהו ה־k הטבעי הקטן ביותר כך ש־ $a^k = id$ (כאשר id הוא איבר היחידה של החבורה).

למה יש כזה? כי האיברים $id, a, a^2, ...$ מהווים תת־חבורה של G, וגודלה הוא k. (תת־החבורה נקראת תת החבורה הציקלית של G הנוצרת ע"י a: $\langle a \rangle_G$).

מסקנה ממשפט לגראנז': לכל חבורה סופית G ולכל $a \in G$, מתקיים ש־ $ord(a)$ מחלק את $|G|$.

מסקנה 6.16 לכל n טבעי ולכל a שזר ל־n מתקיים $a^{\varphi(n)} \equiv 1_{mod\, n}$.

הוכחה: יהיה $k = ord(a)$ (ז"א הסדר של a כאיבר בחבורה $\mathbb{Z}_n^*$). ע"פ משפט לגראנז', $k||\mathbb{Z}_n^*|$, כלומר $k|\varphi(n)$. מהגדרת הסדר, $a^k \equiv 1 (mod\, n)$. נעלה את שני האגפים בחזקה (השלמה) $\frac{\varphi(n)}{k}$ ונקבל $a^{\varphi(n)} \equiv 1_{mod\, n}$. ■

דוגמה:

$a = 5$, $n = 48$.

$\varphi(48) = 48 \cdot (1-\frac{1}{2}) \cdot (1-\frac{1}{3}) = 48 \cdot \frac{1}{2} \cdot \frac{2}{3} = 16$ ולכן,

$5^{16} \equiv 1 (mod\, 48)$

$5^4 = 625 = 13 * 48 + 1$

$5^4 = 1 (mod\, 48)$

נעלה ברביעית:

$5^{16} = 1 (mod\, 48)$

מקרה פרטי ידוע: המשפט הקטן של פרמה: לכל $1 \le a \le p-1$ כש־p ראשוני, מתקיים $a^{p-1} \equiv 1_{mod\, p}$.

- ידוע לנו שאם h פולינום ממעלה d מעל שדה כלשהו, אז למשוואה $h(x) = 0$ יש לכל היותר d שורשים בשדה.

טענה 6.17 אם $n = p^e$ כש־p ראשוני אי־זוגי, ו־$e \ge 1$, אז למשוואה $x^2 \equiv 1\, mod\, n$ אין פתרון מלבד $x = \pm 1\, mod\, n$.

הוכחה: את המשוואה הזו $x^2 \equiv 1\, mod\, n$ נרשום כ־

$p^e | (x^2 - 1)$

$p^e | (x+1)(x-1)$

לכן או ש־ $p^e | x+1$ (ואז $x \equiv -1\, mod\, p^e$), או ש־ $p^e | x-1$ (ואז $x \equiv 1\, mod\, p^e$), אבל לא שניהם כי ההפרש בין $x+1$ ל־$x-1$ הוא 2.

(לא ממש ברור לי למה זה אומר שהוא מחלק את אחד מהם לפחות, ייתכן שצריך פה איזשהו נימוק?)

(הסבר ללמה הוא לא מחלק את שניהם: p ראשוני אי־זוגי. לכן $p > 2$. אם נניח בשלילה שהוא מחלק את שניהם, אזי נקבל $p \cdot k_1 = x+1$ וגם $p \cdot k_2 = x-1$. נחסיר המשוואות ונקבל $p(k_1 - k_2) = (x+1)-(x-1) = 2$. כלומר ההפרש בין $x-1$ ו־$x+1$, שהוא 2, הוא כפולה של p. אבל $p > 2$ לפי ההנחה, ולכן הדבר לא ייתכן, וקיבלנו סתירה. תודה רבה לנעם על התיקון+ההסבר). ■

משפט 6.18 לכל חזקת ראשוני $n = p^e$ (p ראשוני, $e \geq 1$) מתקיים שהחבורה $\mathbb{Z}_n^*$ היא ציקלית. (לא נוכיח את המשפט).

החבורה הציקלית מוד' m זו החבורה של חיבור מודולו m. לחילופין, זו חבורה הנוצרת ע"י איבר יחיד.
G היא חבורה ציקלית $\Leftrightarrow$ יש איבר $g \in G$ ("יוצר") כך שלכל $h \in G$ יש k טבעי כך ש־ $h = g^k$.

במקרה הרגיל של חיבור מוד' m, אפשר לקחת למשל את $g = 1$, או כל מספר אחר שזר ל־m. למשל, $m = 12$, ניקח $a = 7$, אז לכל $0 \leq h \leq 11$ יש k כך ש־ $7k = h \,(mod\, 12)$.

6.2.5 אלגוריתם $Miller - Rabin$, הקדמה

הרעיון הבסיסי של אלגוריתם MR הוא זה:
יש לנו שני תנאים מספיקים לכך שהמספר n הוא פריק:
0. מצאנו a כך ש־ $\gcd(a, n) > 1$)
1. הפרת המשפט הקטן של פרמה: ז"א מציאת $2 \leq a \leq n - 1$ כך ש־ $a^{n-1} \not\equiv 1 \,(mod\, n)$
2. מציאת "שורש ריבועי לא טריויאלי מוד' n", זאת אומרת מוצאים $x \not\equiv \pm 1 \,(mod\, n)$ ואף על פי כן $x^2 \equiv 1 \,(mod\, n)$.

אם מוצאים a כמו ב־1 או x כמו ב־2, יש לנו עד ($witness$) לפריקות של n.

- הרעיון הבסיסי: אנחנו נראה (בערך) שאם n פריק, אז לפחות חצי מהמספרים בין $2, ..., n - 1$ הם עדים לפריקותו. האלג' יבחר a כזה באקראי. אם הוא מתברר כעד, אנו **בטוחים** ש־n פריק. אם חוזרים על הניסוי מספר גדול של פעמים ואף פעם לא מתגלה לנו עד, האלגוריתם עונה "n כנראה ראשוני". במקרה זה, האלג' עלול לטעות, אבל אפשר להקטין את הסתברות הטעות כרצונינו.

מתברר, (וזו כבר עובדה קשה), ש"כמעט תמיד" כבר המבחן האם $2^{n-1} \not\equiv 1 \,(mod\, n)$ מגלה פריקות.
(ז"א מספר המספרים האי־זוגיים הפריקים n בין 1 ל־N שבשבילם $2^{n-1} \equiv 1 \,(mod\, n)$ הוא $o(N)$).
אבל מתברר שיש גם חריגים. יש מספרים טבעיים הנקראים מספרי $Carmichael$ כך שאם n מס' כזה ו־ $1 \leq a \leq n - 1$ זר ל־n אז $a^{n-1} \equiv 1 \,(mod\, n)$.
ל־n־ים כאלה, משפט פרמה לא יופר אף פעם, ולכן אין לנו עדים מטיפוס כזה לפריקות של n.

הגדרה 6.19 אומרים ש־n הוא **מספר Carmichael** אם הוא פריק, ולכל $2 \leq a \leq n - 1$ שזר ל־n מתקיים $a^{n-1} \equiv 1 \,(mod\, n)$.
(הגדרה שקולה (אולי? לא בטוח): $\forall a \quad a^n = a \,(mod\, n)$).

טענה 6.20 מס' $Carmichael$ הקטן ביותר הוא $561 = 3 \cdot 11 \cdot 17$.

הוכחה: (לא ממש הוכחה, אלא הסבר):
עלינו להראות שלכל $1 \leq a \leq 560$ כך ש־a זר ל־561, (ז"א $3 \nmid a$, $11 \nmid a$, $17 \nmid a$) מתקיים $a^{560} \equiv 1 \,(mod\, 561)$.
משפט השאריות הסיני (CRT): אם $t_1, ..., t_d$ מס' טבעיים זרים בזוגות, אז ידיעת הערכים של $x \, mod \, t_i$ ל־$i = 1, ..., d$ מגדירה באופן יחיד את $x \, mod \prod t$.
מ־CRT נובע שעל מנת להוכיח כי $a^{560} \equiv 1 \,(mod\, 561)$ הכרחי ומספיק להוכיח כי:
$a^{560} \equiv 1 \,(mod\, 3)$
$a^{560} \equiv 1 \,(mod\, 11)$
$a^{560} \equiv 1 \,(mod\, 17)$
אנחנו יודעים ש־ $a^{16} \equiv 1 \,(mod\, 17)$ לכל $17 \nmid a$ (מהמשפט של פרמה).
היות ש־ $560 = 16 \cdot 35$, אפשר להעלות בחזקה: $a^{560} = 1 \,(mod\, 17)$. וכך באופן דומה בשני המקרים האחרים (כנראה).
לסיכום: 561 הוא מס' $Carmichael$ כי הוא מכפלה של ראשוניים אי־זוגיים שונים, $3 \cdot 11 \cdot 17$, וכמו כן:
$(17 - 1) | (561 - 1)$
$(11 - 1) | (561 - 1)$

$(3-1)|(561-1)$ ■

בעצם, מספר n הוא מספר $Carmichael$ ⇔הוא מכפלה של מספרים ראשוניים שונים $(square\ free)$ $n=p_1\cdot\ldots\cdot p_k$ כך שלכל i, $(p_i-1)|(n-1)$.
רק ב־1994 הוכח שיש אינסוף מספרים כאלה.

6.2.6 מבחן הראשוניות של $Miller-Rabin$

קלט: מספר n טבעי (אי־זוגי).
פלט: האם n ראשוני\פריק?

האלגוריתם:
בוחרים באקראי $2\le a\le n-1$.
צעד 0: בודקים אם $\gcd(a,n)>1$. אם כן, "n פריק" ומסיימים.
כעת, רושמים $n-1=u\cdot 2^t$ כש־ u מספר אי־זוגי.
מחשבים את $a^u\ (mod\, n)$ ואז את הסדרה:

$$(a^u)^2=a^{2u},\ (a^{2u})^2=a^{4u},\ \ldots,\ (a^{2^{t-1}u})^2=a^{2^tu}=a^{n-1}\qquad (\mathbf{mod\, n})$$

אם $a^{n-1}\not\equiv 1\ (mod\, n)$ אז "n פריק, בוודאות". (כי n מפר את משפט פרמה)
אחרת, מביטים בסדרת המספרים $(mod\, n)$: $a^u, a^{2u}, a^{4u}, \ldots, a^{n-1}$. הסדרה הזו מסתיימת ב־1, מההנחה שלנו לגבי המקרה הנוכחי (אחרת היינו מסיימים בשורה הקודמת!).
מביטים במקום האחרון בסדרה שבו היה ערך $\ne 1\ (mod\, n)$.
אם במקום הזה מופיע ערך שהוא $\ne -1\ (mod\, n)$ אז "n פריק, בוודאות". (כי מצאנו פתרון $\ne\pm 1$ למשוואה $x^2\equiv 1\, mod\, n$).
בכל מקרה אחר, "n **כנראה** ראשוני".

משפט 6.21 אם n אי־זוגי הוא פריק אז לפחות $\frac{n-1}{2}$ מהמספרים a שניתן לבחור יראו זאת.
לכן ההסתברות לטעות בריצה בודדת של האלג' (ז"א n פריק אבל האלג' אומר "כנראה ראשוני") היא $<\frac{1}{2}$. (ולכן אם נחזור על זה s פעמים אז ההס' $<\frac{1}{2^s}$).

דיון נוסף באלגוריתם:
נביט בסדרת המספרים הזו: $a^u, a^{2u}, a^{4u}, \ldots, a^{n-1}$. יש כמה מקרים:

- $1,\ldots,1$ הסדרה היא כולה אחדים. במקרה זה, האלגוריתם יאמר "כנראה ראשוני".
- $\ldots,(-1),1,1,\ldots,1$ כלומר מסתיים באחדים, והמעבר היה מ־(-1). המס' n עמד בשני המבחנים, גם $a^{n-1}\equiv 1\ (mod\, n)$ וגם במבחן של $x^2\equiv 1$ (כאן $x=-1$).
- האפשרות המעניינת יותר: $\ldots,x,1,1,\ldots,1$ כאשר $x\ne 1$. כאן נחשפה פריקותו של n ע"פ הקריטריון השני.

הערה 6.22 אם הקלט n ראשוני אז הפלט יהיה בוודאות "כנראה ראשוני".
אם n פריק, אז בהסתברות $\ge\frac{1}{2}$ הפלט יהיה "פריק",
ובהסתברות $<\frac{1}{2}$ הפלט יהיה "כנראה ראשוני". (וכאן אנו נכשלים! כי בחרנו a שאיננו עד).
מטרתינו להראות שאם n הוא פריק אז ההסתברות שנבחר אי־עד היא קטנה ⇔מס' האי־עדים הוא $\le\frac{n-1}{2}$.

נראה זאת ע"י כך שנוכיח כי קבוצת האי־עדים תת־חבורה ממש, של $\mathbb{Z}_n^*$. נקרא לה H.
$|H|\,|\,|\mathbb{Z}_n^*|$ ולכן $|H|\le\frac{1}{2}|\mathbb{Z}_n^*|\le\frac{n-1}{2}$.

מקרה 1: n איננו מספר $Carmichael$. נגדיר:

$$B = \{a \in \mathbb{Z}_n^* \,:\, a^{n-1} \equiv 1\,(mod\,n)\}$$

B היא חלקית ממש ל־$\mathbb{Z}_n^*$
(זה בדיוק פירוש הדבר ש־n איננו מספר $Carmichael$, ז"א יש $a \in \mathbb{Z}_n^*$ המקיימים $a^{n-1} \not\equiv 1\,(mod\,n)$).
מדוע B תת־חבורה:
אם $a, b \in B$ צ"ל שגם $ab \in B$. מתקיים: $\begin{matrix} a^{n-1} \equiv 1\,(mod\,n) \\ b^{n-1} \equiv 1\,(mod\,n) \end{matrix}$ ⇐נכפיל: $(ab)^{n-1} \equiv 1\,(mod\,n)$.
הופכי: $(a^{n-1})^{-1} = (a^{-1})^{n-1} \equiv 1\,(mod\,n)$.
מקרה 2: n הוא מספר $Carmichael$.
אז לכל $a \in \mathbb{Z}_n^*$ מתקיים $a^{n-1} \equiv 1\,(mod\,n)$.

נראה תחילה שלא ייתכן ש־n הוא חזקת ראשוני: $n = p^e$ (כאשר p ראשוני, $e \geq 2$).
כזכור, אם $n = p^e$ אז $\mathbb{Z}_n^*$ חבורה ציקלית. יהיה g יוצר שלה. בפרט, $ord(g) = |\mathbb{Z}_n^*|$.
נשים לב (לפי פונקציית אוילר):

$$|\mathbb{Z}_n^*| = \varphi(n) = \varphi(p^e) = p^e \cdot (1 - \frac{1}{p}) = p^e - p^{e-1}$$

ולכן $ord(g) = p^e - p^{e-1}$.
מצד שני, $g^{n-1} \equiv 1\,(mod\,n)$ כלומר $g^{p^e-1} \equiv 1\,(mod\,n)$
(n הוא מס' $Carmichael$. $g \in \mathbb{Z}_n^*$ ולכן זר ל־n. לכן לפי הגדרת n כמספר $Carmichael$ מתקיים $g^{n-1} \equiv 1_{mod\,n}$).
וממה שראינו על הסדר של g, נקבל $g^{p^e-p^{e-1}} \equiv 1\,(mod\,n)$,
ולכן (כשמחלקים) נקבל

$$g^{p^{e-1}-1} \equiv 1\,(mod\,n)$$

בניגוד לכך ש־ $p^e - p^{e-1}$ הוא הסדר של g.
ולכן הראינו שלא ייתכן ש־n הוא חזקת ראשוני.

במקרה בו n פריק ואינו חזקת ראשוני, שוב נמצא תת־חבורה ממש $B \subseteq \mathbb{Z}_n^*$ הכוללת את כל האי־עדים.
כאמור, אם n מס' $Carmichael$ אז הוא אינו חזקת ראשוני. נגדיר:

$$B = \{x \in \mathbb{Z}_n^* \mid x^{2^j u} \equiv \pm 1\,(mod\,n)\}$$

יש לבחור את האינדקס j כך ש -

- B תת־חבורה
- יש $w \in \mathbb{Z}_n^* \backslash B$ (ולכן B תת־חבורה ממש).

היות ש־n אי־זוגי ואיננו חזקת ראשוני, ניתן לרשום $n = n_1 \cdot n_2$ כאשר n_1, n_2 זרים זה לזה.
בחירת j: ידוע לנו שלכל $a \in \mathbb{Z}_n^*$ מקיים $a^{2^t u} \equiv 1\,(mod\,n)$.
נגדיר את j כאינדקס הגדול ביותר כך שיש ν המקיים $\nu^{2^j u} \equiv (-1)\,(mod\,n)$.
(למה יש בכלל כאלה? ניקח $\nu \equiv (-1)\,(mod\,n)$).

מכאן נובע ש־B היא תת־חבורה: מכפלה וגם הופכי של איברים ב־B גם הם ב־B.
כעת, מחפשים $w\in\mathbb{Z}_n^*\backslash B$.

ע"פ משפט השאריות הסיני, יש w המקיים $\begin{cases} w\equiv\nu\,(mod\,n_1)\\ w\equiv 1\,(mod\,n_2)\end{cases}$ ★.

לכן, $w^{2^ju}\equiv 1\,(mod\,n_2)$ וגם $w^{2^ju}\equiv\nu^{2^ju}\equiv(-1)\,(mod\,n_1)$.

- נשים לב:
  - $w\in\mathbb{Z}_n^*$
  - $w^{2^ju}\neq\pm1\,(mod\,n)\Leftrightarrow w\notin B$

כעת, אם $w^{2^ju}\equiv 1\,(mod\,n)$, $\Leftarrow$אז $w^{2^ju}\equiv 1\,(mod\,n_1)$ (בסתירה לכך ש־ $w^{2^ju}\equiv(-1)\,(mod\,n_1)$).
ואם $w^{2^ju}\equiv(-1)\,(mod\,n)$, $\Leftarrow$ אז $w^{2^ju}\equiv(-1)\,(mod\,n_2)$ (בסתירה לכך ש־ $w^{2^ju}\equiv(+1)\,(mod\,n_2)$).

לכן מצאנו איבר שאינו ב־B.
למה $w\in\mathbb{Z}_n^*$? אילו היה ל־w גורם משותף עם n, היה לו גם גורם משותף עם n_1 או עם n_2, בסתירה ל־★.

7 אלגברה ליניארית

7.0.7 אלגוריתם $Strassen$ להכפלת מטריצות

בעבר, ציינו את העובדה (ללא הוכחה) שלבעיה הבאה יש סיבוכיות ריבועית (נדרשות לפחות n^2 פעולות כפל):
קלט: מטריצה (ממשית) $A_{n\times n}$ ווקטור $x\in\mathbb{R}^n$.
פלט: הוקטור Ax.

אבל מה בדבר הבעיה החישובית הבאה:
קלט: שתי מטריצות A,B ממשיות, $n\times n$.
פלט: המטריצה AB.
לאלגוריתם שאותו לומדים בקורס באלגברה ליניארית יש סיבוכיות $O(n^3)$.
להפתעת רבים, אלג' יעיל יותר נמצא ע"י $V\ Strassen$ עם זמן ריצה $O(n^{log_2 7})=O(n^{2.83})$.
איננו יודעים מהו המעריך הנמוך ביותר ω כך שניתן להכפיל 2 מטריצות $n\times n$ בזמן $O(n^\omega)$. אחרי עבודתו של $Strassen$ נמצאו שיפורים נוספים ־ האלג' הטוב ביותר המוכר לנו משיג $\omega=2.36...$. רבים מהתחום הזה משערים שהאמת היא $\omega=2+\varepsilon$ לכל $\varepsilon>0$.

תיאור כללי של אלגוריתם $Strassen$**:**
ניתן להכפיל שתי מטריצות 2×2 תוך שימוש ב**שבע** (ולא שמונה) פעולות כפל, וזאת גם אם איברי המטריצות אינם מתחלפים כפלית זה בזה *.
רוצים לחשב:

$$\begin{pmatrix} a_{11} & a_{12}\\ a_{21} & a_{22}\end{pmatrix}\begin{pmatrix} b_{11} & b_{12}\\ b_{21} & b_{22}\end{pmatrix}=\begin{pmatrix} c_{11} & c_{12}\\ c_{21} & c_{22}\end{pmatrix}$$

כאשר ה־ a_{ij} וה־ b_{kl} אינם מתחלפים אלו באלו (= לא ניתן להניח $xy=yx$).
("העשרה": פורמלית אנו עובדים בחוג $\mathbb{R}[x,y]$ כש־ x,y משתנים שאינם מתחלפים).
מדוע העובדה הזו (*) מאפשרת לנו לחשב כפל מטריצות בצורה יעילה יותר מהדרך הסטנדרטית?
$A=\begin{pmatrix} A_{11} & A_{12}\\ A_{21} & A_{22}\end{pmatrix}$ כאשר כל A_{ij} היא מטריצה ממשית של $\frac{n}{2}\times\frac{n}{2}$.

אז את הפעולה $AB = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$ נבצע באמצעות האלגוריתם המתואר ב-*.
נקבל את החסם: $T(n) \leq 7 \cdot T(\frac{n}{2}) + cn^2$ כאשר $T(n)$ מס' הפעולות הנדרש להכפלת שתי מטריצות ממשיות $n \times n$ זו בזו, וה-cn^2 הוא מה שמשלמים על "הנהלת חשבונות" ופעולות עזר.
מכאן לא קשה להסיק ש- $T(n) \leq O(n^{log_2 7})$. (חישבנו את זה בהרצאה...)
כמו כן ראינו בהרצאה תיאור חלקי ביותר של האלגוריתם להכפלת מטריצות 2×2. זה לא היה הכי ברור אז זה לא מופיע בסיכום זה...

7.0.8 על פתרון מערכות משוואות ליניאריות

פתרון מערכות משוואות ליניאריות $Ax = b$ זו אחת הפעולות החשובות ביותר בכל יישום מתמטי. בסיטואציות מעשיות, די רגיל ש: איברי A ידועים לנו רק בערך, וכנ"ל לאיברי b.

- ניתוח רגישות: עד כמה רגיש הפתרון שנמצא x לסטיות קטנות בערכים שב- A, b?
- מהם המאפיינים של הקלט A, b שגורמים לכך ש:
  - שינויים קטנים בערכים שב- A, b גורמים לשינוי קטן בערך הנכון של x (מצב רצוי).
  - השינוי המתקבל ב-x עלול להיות גדול (מצב לא רצוי).

די שכיח לנסות להתמודד עם הבעיה ע"י "הוספה של מדידות".
$x \in \mathbb{R}^n$. בעולם המושלם של אלגברה ליניארית עיונית, ניתן למצוא את x באופן יחיד מתוך מערכת משוואות $Ax = b$ כש-A מטריצה לא-סינגולרית $n \times n$.
(כל שורה ב-A מתאימה ל"איסוף מידע" על x).
ניתן להגיע למצב שבו ב-A יש $m > n$ שורות, ורוצים לפתור $Ax \approx b$.
$A_{m \times n}$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$.

בכל הנאמר כאן נניח $m \geq n$ וגם $rank(A) = n$ (= העמודות בת"ל)

מה זה '$\approx$'?
ראשית, איברי A, b ידועים לנו רק בקירוב.
במערכת הליניארית $Ax = b$ יש יותר משוואות מנעלמים, וכרגיל אין למערכת כזו פתרון.
נפרש, אם כך, את $Ax \approx b$ באופן הבא:

מצא x כך שהוקטור Ax "קרוב ל-b"$\Leftrightarrow$הוקטור $Ax - b$ הוא קטן.

על מנת להפוך את הדיון הזה למשמעותי, אנו נזקקים לדרך שתאפשר לייחס "גודל" לוקטורים.
איך מכמתים את העובדה שהמס' הממשיים u, v "קרובים": ע"י הסתכלות ב- $|u - v|$.
ז"א, הגודל של המס' הממשי t הוא $|t|$.
המטרה היא אם כן, למצוא וקטור x כך של-$Ax - b$ יש **נורמה** קטנה.

נסמן $x = (x_1, ..., x_n)$. הנורמות החשובות ביותר בשימושים הן:

- $\parallel x \parallel_2 = \sqrt{x_1^2 + ... + x_n^2}$ (נורמת l_2, נורמה אוקלידית)
- $\parallel x \parallel_1 = \sum |x_i|$
- $\parallel x \parallel_\infty = \max_i |x_i|$ (נורמת ∞, נורמת - מקס)

נורמה וקטורית:
זוהי העתקה $\mathbb{R}^n \to \mathbb{R}_+$ המסומנת $x \mapsto \parallel x \parallel$ ומקיימת:

1. $\| x \| \geq 0$ $\forall x \in \mathbb{R}^n$ ושוויון אם"ם $x = 0$ (אי־שליליות)

2. $\| \alpha x \| = |\alpha| \cdot \| x \|$ $\forall x \in \mathbb{R}^n, \ \forall \alpha \in \mathbb{R}$ (ליניאריות)

3. $\| x + y \| \leq \| x \| + \| y \|$ $\forall x, y \in \mathbb{R}^n$ (א"ש המשולש)

תזכורת: דרגה $(rank)$ של מטריצה. אם A מטריצה, אז דרגת השורות שלה זהו המספר המירבי של שורות בת"ל ב־A. באותו אופן מגדירים את דרגת העמודות של A. זהו משפט יסודי באלגברה ליניארית שהשניים שווים, ולערך המשותף קוראים הדרגה של A, ומסמנים $rank(A)$. עובדה נוספת מאלגברה ליניארית: אומרים על מטריצה A שהיא מדרגה 1 אם יש וקטורים u, v כך ש־ $a_{ij} = u_i \cdot v_j$. ואז הדרגה של מטריצה כללית M זהו המספר הקטן ביותר של מטריצות מדרגה 1 שסכומן הוא M. $\begin{array}{c\|ccc} & v_1 & \cdots & v_n \\ \hline u_1 & & & \\ \vdots & & & \\ u_n & & & \end{array}$.

בעיה

בהנתן $A_{m \times n}$ מטריצה ממשית $(m \geq n)$ ו־ $rank(A) = n$, וקטור $b \in \mathbb{R}^m$, מצא את x כך ש־ $\min \| Ax - b \|_2$.

- נקודת מבט גיאומטרית על השאלה: האוסף $\{Ax\}_{x \in \mathbb{R}}$ הוא תת־מרחב n מימדי של $\mathbb{R}^n$. נקרא לו W. לכן, השאלה שלנו היא זו: בהנתן W תת־מרחב n־מימדי של $\mathbb{R}^n$, ונקודה $b \in \mathbb{R}^n$, מצאו נקודה ב־W שקרובה ביותר ל־b (במובן של נורמה אוקלידית).

(הערה שלי: כנראה שבפסקה האחרונה הכוונה היא ל־m, לא n).

מנקודת מבט זו, ברור מה עלינו לעשות: למצוא את ההיטל הניצב של b על W, נאמר y, ואז לבטא $y = Ax$ (יש x אחד ויחיד כזה).
אופייני ש־l_2 נוחה לעבודה, כי כלים מאנליזה מתמטית מטפלים בה היטב. לעומת זאת, ב־ l_1, l_∞ נוח לטפל לרוב בכלים של תכנון ליניארי.
נפתור את הבעיה $\min_x \| Ax - b \|_2$ ע"י גזירה ואיפוס הנגזרת.
על מנת למזער את $\| Ax - b \|_2$ הכרחי ומספיק למזער את $\| Ax - b \|^2$.
אבל, $\| y \|_2^2 = \langle y, y \rangle = y^T y$,

$$\min_x \| Ax - b \|_2^2 = \min_x \langle Ax - b, Ax - b \rangle = \min \left(\left(x^T A^T - b^T \right) (Ax - b) \right) =$$

$$= \min \left(x^T A^T Ax - x^T A^T b - b^T Ax + b^T b \right) =$$

נשים לב ש־ $x^T A^T b = b^T Ax$ (שניהם מספרים והם שווים), ולכן

$$= \min \left(x^T A^T Ax - 2b^T Ax + \| b \|_2^2 \right)$$

נסמן $Z = A^T A$. Z לא סינגולרית, כי לפי ההנחה $rank(A) = n$, ובנוסף Z סימטרית: $Z_{\alpha\beta} = Z_{\beta\alpha}$. אז מחפשים:

$$\min_x \left(x^T Z x - 2b^T A + \| b \|_2^2 \right) = \min_x \left(\sum_{i,j} Z_{ij} x_i x_j - 2 \sum_j (A^T b)_j x_j + \| b \|_2^2 \right) = \bigstar$$

נשים לב ש־ $uMv = \sum_{i,j} u_i v_j M_{ij}$.
זה נכון כי $\langle\alpha,\beta\rangle = \sum \alpha_i\beta_i$. אז,

$$\langle uM, v\rangle = \sum (uM)_j v_j = \sum_j v_j (uM)_j = \sum_j v_j \sum_i u_i M_{ij} = \sum_{i,j} u_i v_j M_{ij}$$

עכשיו נרצה למזער את ★:
נגזור ע"פ x_k ונשווה את הנגזרת ל־0.

$$\forall k \qquad 2\sum_t Z_{kt}x_t - 2(A^T b)_k = 0$$

(יש 2 בצד שמאל, כי יש פעם אחת ל־i ופעם אחת ל־j, ומהסימטריות של Z. זה יוצא $\sum_i Z_{ik}x_i + \sum_j Z_{kj}x_j$). נמשיך:

$$\forall k \qquad \sum_t Z_{kt}x_t = (A^T b)_k$$

אבל $(Zx)_k = \sum_t Z_{kt}x_t$ ולכן,

$$Zx = A^T b \ \Rightarrow \ x = Z^{-1}A^T b$$

$$x = (A^T A)^{-1} A^T b$$

(יש הופכי כי Z לא סינגולרית).
הערה: למטריצה $(A^TA)^{-1}A^T$ קוראים **ההפיך המוכלל** של A ($Moore-Penrose\,inverse$).
לסיכום: חיפשנו את $\min \parallel Ax-b \parallel_2$, מצאנו את $\min \parallel Ax-b \parallel_2^2 = \langle Ax-b, Ax-b\rangle$. זהו פולינום ממעלה שניה ב־$x_1,...,x_n$.
גזרנו אותו ומצאנו את התשובה.

7.0.9 קירוב בריבועים פחותים

הזכרנו בעבר את המושג של אינטרפולציה:
נתונות $n+1$ נקודות במישור: $(x_0,y_0),...,(x_n,y_n)$ (כאשר ה־x_i שונים זה מזה).
אז יש פולינום יחיד ממעלה $\geq n$, P, כך ש־ $\forall i \;\; P(x_i) = y_i$.
מה ניתן לומר אם אנו מעוניינים בפולינום Q ממעלה $\geq k$ כש־ $n > k$?
כמובן, **אין** Q ממעלה $\geq k$ המקיים $\forall i \;\; Q(x_i) = y_i$.
ניתן לבקש שוב קירוב טוב ביותר במובן של l_2 ("ריבועים פחותים"), ז"א למצוא Q ממעלה $\geq k$ כך ש־

$$\min_{Q\, is\, a\, polynomial\, of\ \deg \leq k} \left(\sum_{i=0}^{n} (y_i - Q(x_i))^2 \right)$$

לא קשה לראות שזה מקרה פרטי של הבעיה שכבר פתרנו:

$$A = \begin{pmatrix} 1 & x_0 & \cdots & x_0^k \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & \cdots & x_n^k \end{pmatrix} \qquad b = \begin{pmatrix} y_0 \\ \vdots \\ y_n \end{pmatrix} \qquad \alpha = (\alpha_0, \alpha_1, ..., \alpha_k)$$

$$(A\alpha - b)_i = Q(x_i) - y_i$$

$$Q(t) = \alpha_0 + \alpha_1 t + ... + \alpha_k t^k$$

ומקבלים שתחת הפרמטרים האלו, זו בדיוק הבעיה שאנחנו כבר יודעים לפתור, של $\min_x \| Ax - b \|_2$.
זה נכון כי מתקיים:

$$A\alpha = \begin{pmatrix} 1 & x_0 & \cdots & x_0^k \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & \cdots & x_n^k \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_k \end{pmatrix} = \begin{pmatrix} \alpha_0 + \alpha_1 x_0 + \alpha_2 x_0^2 + ... + \alpha_k x_0^k \\ \alpha_0 + \alpha_1 x_1 + \alpha_2 x_1^2 + ... + \alpha_k x_1^k \\ \vdots \\ \alpha_0 + \alpha_1 x_n + \alpha_2 x_n^2 + ... + \alpha_k x_n^k \end{pmatrix} \begin{matrix} \Leftarrow Q(x_0) \\ \Leftarrow Q(x_1) \\ \\ \Leftarrow Q(x_n) \end{matrix}$$

7.0.10 הנורמה האופרטורית

ראינו כמה דרכים מעניינות לייחס "אורך" או "גודל" לוקטורים ע"י המושג של נורמה.
עכשיו נרצה להגדיר גם נורמות של מטריצות.
איך עושים זאת?

- ניתן להתעלם מהעובדה שבמטריצה האיברים מופיעים במלבן, ולהתייחס אליהם כאל אוסף של מספרים.
  למשל, A מטריצה, אז $\sqrt{\sum a_{ij}^2} = \| A \|_2$ נורמת $Frobenius$ ($Hillbert - Schmidt$).
- נזכור שמטריצה היא לא רק מערך של מספרים, אלא מייצגת גם העתקה ליניארית.

יהיו X, Y מרחבים ליניאריים, שכל אחד מהם מצוייד בנורמה משלו: $\|\|_Y$, $\|\|_X$. ותהינה $A: X \to Y$ העתקה ליניארית.

הגדרה 7.1 נביט ב**נורמה האופרטורית של** A:

$$\| A \|_{op} = \max_{v \in X} \frac{\| Av \|_Y}{\| v \|_X}$$

במילים: $\| A \|_{op}$ הנורמה האופרטורית של A זהו יחס המתיחה המירבי בין האורך של תמונתו של וקטור לאורכו המקורי.

כעת,
נביט שוב במצב שבו עלינו לפתור מערכת $Ax = b$.
מה יקרה אם b אינו ידוע לנו במדויק?
מה אם אגף ימין האמיתי (שאינו ידוע לנו) איננו b אלא $b + \Delta b$?
הפתרון האמיתי הוא אם כן $x + \Delta x$.

$$A(x + \Delta x) = b + \Delta b$$

$$Ax + A\Delta x = b + \Delta b$$

אבל $Ax = b$ אז אפשר לצמצם:

$$A\Delta x = \Delta b$$

$$\Delta x = A^{-1}\Delta b$$

החשש הוא שאף כי $\frac{\|\Delta b\|}{\|b\|}$ קטן, (זו הטעות היחסית ברישום אגף ימין), הטעות היחסית שלנו בפתרון: $\frac{\|\Delta x\|}{\|x\|}$ היא גדולה.
כשזה קורה, אומרים שהבעיה ("פתור את $Ax = b$") היא $ill - posed$.
לדוגמה:

$$A = \frac{1}{2}\begin{pmatrix} 1 & 1 \\ 1+\varepsilon & 1-\varepsilon \end{pmatrix} \qquad A^{-1} = \frac{1}{\varepsilon}\begin{pmatrix} -1+\varepsilon & 1 \\ 1+\varepsilon & -1 \end{pmatrix}$$

נשים לב שהמספרים במטריצה A^{-1} יהיו מאוד גדולים, ככל ש־ ε קטן.
היות ש־ $\Delta x = A^{-1}\Delta b$, יוצא שטעות קטנה ברישום של b מתבטאת בטעות גדולה מאוד ב־ Δx.
הקושי מתבטא בכך שלמטריצה A^{-1} יש נורמה אופרטורית גדולה.

7.0.11 ערכים עצמיים לעומת ערכים סינגולריים

אנו פותרים מערכת משוואות ליניאריות $Ax = b$ ($A_{n\times n}$ לא סינגולרית), ובפועל אגף ימין היה אמור להיות $b + \Delta b$ כאשר Δb הוא וקטור לא ידוע של "טעויות מדידה". ראינו כבר ש־ $\Delta x = A^{-1}\Delta b$.
השאלה המעניינת אותנו היא האם בהנחה ששגיאת המידה שלנו Δb קטנה, גם הטעות בפתרון Δx תהיה קטנה.
יש מטריצות A שבשבילן התשובה היא חיובית, ואז אומרים שהמערכת $Ax = b$ מוצגת היטב ($well - posed$), אבל יש גם A שבשבילן המערכת אינה מיצגת היטב ($ill - posed$).
יותר בדיוק, רוב העניין שלנו הוא בטעויות יחסיות, ולכן השאלה שבה נתעניין ממש היא האם בהנתן ש־ $\frac{\|\Delta b\|}{\|b\|}$ קטן (כאן $\|\|$ היא נורמה שבחרנו ואיתה נעבוד), גם הטעות היחסית בתשובה, $\frac{\|\Delta x\|}{\|x\|}$ קטנה.
(כלומר האם טעות יחסית קטנה בקלט גוררת טעות יחסית קטנה בתשובה שלנו).
נזכור שהגדרנו:

$$\| A \|_{op} = \max_{v\in X} \frac{\| Av \|_Y}{\| v \|_x}$$

ולכן לכל $u \in X$ מתקיים $\| Au \| \leq \| A \| \| u \|$. לכן,

$$\| A \| \| x \| \geq \| Ax \| = \| b \|$$

$$\| A^{-1} \| \| \Delta b \| \geq \| A^{-1} \Delta b \| = \| \Delta x \|$$

נכפיל את המשוואות:

$$\| A \| \| A^{-1} \| \| x \| \| \Delta b \| \geq \| b \| \| \Delta x \|$$

$$\| A \| \| A^{-1} \| \frac{\| \Delta b \|}{\| b \|} \geq \frac{\| \Delta x \|}{\| x \|}$$

ונסמן $\kappa(A) = \| A \| \| A^{-1} \|$ ("$condition\, number$").
יוצא שאם $\kappa(A)$ קטן, הבעיה "פתור $Ax = b$" מוצגת היטב.
(בעצם גם ההיפך נכון, אבל לא ניכנס לזה).
קל לראות שלכל A מתקיים $\kappa(A) = \| A \| \| A^{-1} \| \geq 1$:

הוכחה: לפי מה שראינו קודם, $\| x \| = \| AA^{-1}x \| \leq \| A \| \| A^{-1} \| \| x \| = \kappa(A) \cdot \| x \| \qquad \forall x$.

- עוד אי שוויון שימושי: אם A, B מטריצות $n \times n$, אז $\| AB \| \leq \| A \| \| B \|$.

<u>איך מוצאים את $\| A \|_{op}$?</u>
כאן $\| A \|_{op} = \| A \|_{2 \to 2}$ ז"א הנורמות הוקטוריות המופיעות כאן הן נורמת l_2.
(אז $\| A \|_{op} = \max_x \frac{\| Ax \|_2}{\| x \|_2}$).
קל לראות שזה לא מושפע מגודל הוקטור, ולכן:

$$\| A \|_{op_{2 \to 2}} = \max_x \frac{\| Ax \|_2}{\| x \|_2} = \max_{\| x \|_2 = 1} \| Ax \|_2$$

כלומר, השאלה שעלינו לפתור על מנת למצוא את $\| A \|_{op}$ היא מהו

$$\| A \|_{op}^2 = \max_{\| x \| = 1} \langle Ax, Ax \rangle = \max_{\| x \| = 1} x^T A^T A x = \text{The biggest eigenvalue of } AA^T$$

ההגדרה הווריאציונית של ע"ע אומרת (מנת ריילי):

$$\text{The biggest eigenvalue of a symmetric real matrix M} = \max_{\| u \| = 1} uMu^T$$

מצאנו ש־ $\| A \|_{op_{2 \to 2}}$ שווה בדיוק לשורש של הע"ע הגדול ביותר של המטריצה AA^T (המטריצה AA^T היא מוגדרת אי־שלילית ולכן כל הע"ע שלה ≤ 0).

AA^T אכן מוגדרת אי־שלילית, בעקבות המשפט הבא:
התנאים הבאים שקולים לגבי מטריצה ממשית סימטרית M:

1. M מוגדרת אי־שלילית.
2. ניתן לרשום את M בצורה $M = SS^T$.

מינוח:
לגודל שעליו דיברנו ($\| A \|_{op}$): השורש של הע"ע הגדול ביותר של AA^T, קוראים גם **הערך הסינגולרי** הגדול ביותר של A.
באופן כללי, הערכים הסינגולרים של A = שורשי הערכים העצמיים של AA^T.

משפט ה־SVD:
(SVD = פירוק ערכים סינגולריים = $Singular\,Value\,Decomposition$)
תהיה $A_{m\times n}$ מטריצה ממשית ($m \geq n$).
אז ניתן להציג את A בצורה $A = UDV^T$, כאשר
$U_{m\times m}$ אורתוגונלית, $V_{n\times n}$ אורתוגונלית,
D "אלכסונית" עם m שורות, n עמודות: $\begin{pmatrix} \sigma_1 & & 0 \\ & \ddots & \\ & & \sigma_n \\ 0 & & \end{pmatrix}$ ($\sigma_1 \geq ... \geq \sigma_n \geq 0$).
למספרים σ_i קוראים הערכים הסינגולריים של A (למשל, $\sigma_1 = \| A \|_{op_{2\to 2}}$).
אומרים שמטריצה $M_{k\times k}$ היא אורתוגונלית אם:
$MM^T = I$ ⇔ השורות של M הן בסיס אורתונורמלי למרחב ⇔ העמודות של M הן בסיס אורתונורמלי למרחב.
בעיה:
בהנתן מטריצה A, מצא לה קירוב טוב ביותר ע"י מטריצה מדרגה $\leq k$, שנסמנה B.
עד כמה טוב הקירוב המוצע B?

$$\min_{rank(B)\leq k} \| A - B \|$$

שתי הבחירות החשובות ביותר (לבחירת הנורמה הנ"ל):

1. $\|\|_{op}$
2. $\|\|_2$ (כשחושבים על מטריצה כוקטור).

משפט 7.2 תהיה $A_{m\times n}$ ($m \geq n$), $A = UDV^T$ כבמשפט ה־SVD, ויהיה $n \geq k$.
אז הקירוב הטוב ביותר ל־A ע"י מטריצה מדרגה $\leq k$ (הן במובן של נורמה אופרטורית והן במובן של l_2) מתקבל כך:

$$D^{(k)} = \begin{pmatrix} \sigma_1 & & & 0 \\ & \ddots & & \\ & & \sigma_k & \\ & & & 0 \\ 0 & & & \end{pmatrix} \qquad D = \begin{pmatrix} \sigma_1 & & & 0 \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \\ 0 & & & \end{pmatrix} \qquad B = UD^{(k)}V^T$$

בהצלחה במבחן!!

8 נספח: נוסחאות \ דברים שכדאי לזכור למבחן

כאן אשים דברים שנראה לי לנכון לזכור לקראת המבחן. כל האמור כאן זו המלצה\דעתי האישית, כמובן...

8.0.12 משפט השאריות הסיני

זהו הסבר שלי - תרגום של ויקיפדיה מאנגלית פחות או יותר. עשיתי אותו כדי לזכור ולהבין, אבל אולי מישהו ימצא את זה שימושי.

המשפט:

אם $n_1, ..., n_k$ הם מספרים טבעיים זרים בזוגות, ואם $a_1, ..., a_k$ מספרים **כלשהם**, אזי קיים x הפותר את המערכת הבאה:

$$x \equiv a_1 \quad (mod\, n_1)$$
$$x \equiv a_2 \quad (mod\, n_2)$$
$$\vdots$$
$$x \equiv a_k \quad (mod\, a_k)$$

ובנוסף, כל פתרון x למערכת הזו מקיים:

$$\forall i\ x \equiv y\,(mod\, n_i) \Leftrightarrow x \equiv y\,(mod\, N = \prod_i n_i)$$

אלגוריתם לפתרון מערכת המשוואות:

נגדיר $N = n_1 \cdot n_2 \cdot ... \cdot n_k$. ונשים לב שלכל i, מתקיים ש- n_i ו- $\frac{N}{n_i}$ הם זרים (כי אנחנו מניחים שכל ה-n_j זרים בזוגות).

לכן, $\gcd(\frac{N}{n_i}, n_i) = 1 \quad \forall i$. ולכן קיימים r_i, s_i שלמים כך ש- $r_i n_i + s_i \frac{N}{n_i} = 1$ (ניתן למצוא כאלה בעזרת $Extended - Euclid$).

נסמן $e_i = s_i \frac{N}{n_i}$. קיבלנו: $r_i n_i + e_i = 1 \quad \forall i$. ולכן:

$$\forall i \qquad e_i \equiv 1\,(mod\, n_i)$$

כמו כן, מכיוון ש-$e_i = s_i \frac{N}{n_i} = s_i \cdot n_1 \cdot ... \cdot n_{i-1} \cdot n_{i+1} \cdot ... n_k$, אנו יודעים שהוא כפולה של כל n_j ($j \neq i$). כלומר:

$$\forall j \neq i \qquad e_i \equiv 0\,(mod\, n_j)$$

ולכן מחוקי חשבון מודולארי, אנחנו מקבלים שפתרון אפשרי למערכת המשוואות שלנו הוא:

$$x = \sum_{i=1}^{k} a_i e_i$$

(למה? כי כאשר נבצע $mod\, n_i$ נקבל שכל מי שמכיל את e_j עבור $j \neq i$ יתאפס, ואילו האיבר $a_i e_i$ יקבל את הערך $a_i \cdot 1\,(mod\, n_i) = a_i$ כנדרש).

כעת, כל פתרון מהטיפוס $x + Nk$ עבור $k \in \mathbb{Z}$ יהיה חוקי עבור המערכת הנ"ל.

(האם אלו כל הפתרונות? אני לא בטוח).

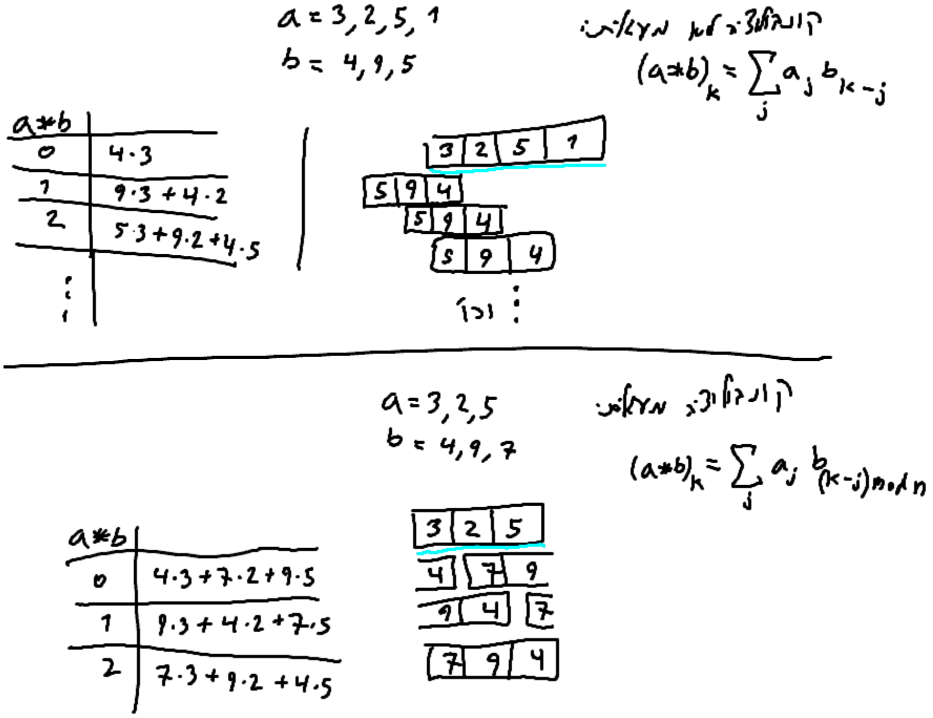
8.0.13 פירוק LUP

המטרה: לפרק מטריצה A למטריצה משולשית עליונה U, משולשית תחתונה L, ומטריצת פרמוטציה P. ויתקיים: $PA = LU$.
אלגוריתם לפירוק $LUP(A)$: ערך ההחזרה הוא שלשה של מטריצות $\{P, L, U\}$.

- אם A בגודל 1×1, מחזירים $LUP(A) = \{(1), (1), A\}$.
- אחרת:
  - נמצא שורה k שעבורה $a_{k1} \neq 0$, ונגדיר מטריצת פרמוטציה Q שמחליפה בין השורות 1 ו־k.
  - נסמן: $QA = \begin{pmatrix} a_{k1} & - & w^T & - \\ | & & & \\ v & & A' & \\ | & & & \end{pmatrix}$. (כנראה! תיתכן כאן טעות).
  - נחשב: $B = A' - \frac{v \cdot w^T}{a_{k1}}$.
  - $\{P', L', U'\} \leftarrow LUP(B)\}$
  - $U = \begin{pmatrix} a_{k1} & w^T \\ 0 & U' \end{pmatrix}$, $L = \begin{pmatrix} 1 & 0 \\ P' \cdot \frac{v}{a_{k1}} & L' \end{pmatrix}$, $P = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & P' & \\ 0 & & & \end{pmatrix} \cdot Q$
  - החזר את $\{P, L, U\}$.
- זמן ריצה: $O(n^2)$ לכל קריאה רקורסיבית. סה"כ $O(n^3)$.

8.0.14 קונבולוציה

- חישוב קונבולוציה: הופכים את הוקטור השני, ואז מריצים אותו מתחת לראשון וכל פעם מחברים... הטבלה בצד שמאל מציינת את $a * b$ במקום ה־i (כן כן, אני יודע שזה ציור יפה, לא צריך להחמיא):



איור 1: חישוב קונבולוציה, מעגלית ולא מעגלית