25.5.21 יום שלישי

מקרא:
- הוכחה
- דוגמה
- משפט
- טענה
- הגדרה

# אלגוריתמים
# שבוע 9 - תרגול 9

## אלגוריתמי קירוב

**הגדרה:** יהי $X$ מרחב הפתרונות החוקיים לבעיית אופטימיזציה. עבור פונקציה $f: X \to \mathbb{R}^+$ והקלט מסוים לבעיה.

נסמן ב-$X^*$ את הפתרון האופטימלי. כלומר $f(X^*) \leq f(X)$ $\forall x \in X$

יהי $C \geq 1$. נאמר שאלגוריתם הוא **C-מקרב** עבור הבעיה: $\min_{x \in X} f(X)$

אם לכל קלט, האלגוריתם מחזיר פתרון חוקי $X \in X$ המקיים: $f(X) \leq f(X^*) \cdot C$

עבור מקסימיזציה $f(X) \geq \frac{1}{C} f(X^*)$.

**הערות:** 1. לרוב נקצר ונסמן ב-OPT את ערך הפתרון האופטימלי.

2. באלג' קירוב, הפתרון שלנו תמיד יהיה גדול (במינימיזציה) וקטן (במקסימיזציה) מהפתרון האופטימלי.

## בעיית ה-3-SAT

**בעיית הכרעה:** האם קיימת לנוסחת 3-CNF השמה מספקת.

**הגדרת הבעיה:** * משתנים בוליאנים $X_i \in \{T, F\}$

* ליטרלים - משתנה $X_i$ או שלילתו $\sim X_i$

* פסוקיות בנוסחת 3-CNF: אוסף של 3 ליטרלים ואיחוד ביניהם

למשל "או" (איווי) ביניהם. למשל: $(X_1 \vee \sim X_2 \vee X_4)$.

**נוסחת 3-CNF:** גימום של פסוקיות.

למשל: $(X_1 \vee \sim X_2 \vee X_4) \wedge (\sim X_1 \vee \sim X_2 \vee X_3)$

**קלט:** נוסחת 3-CNF בעלת $m$ פסוקיות: $C_1,...,C_m$ מעל $n$ המשתנים $X_1,...,X_n$.
כמו כן, נניח $n \le 3m$.

<u>הנחה:</u> המשתנים בכל פסוקית שונים זה מזה.

**פלט:** האם קיימת השמה ל-$X_1,...,X_n$ כך שערך הנוסחה הוא $T$.

* בעיה זו היא NP-קשה, לכן סביר שלא נמצא פתרון יעיל עבורה.
לכן נתפשר על פתרון מקורב לבעיית אופטימיזציה דומה:

**פלט אופטימיזציה:** השמה ל-$X_1,...,X_n$ שממקסמת את מספר הפסוקיות שמקבלות $T$.

<u>לדוגמא:</u> $(X_1 \vee \sim X_3 \vee \sim X_4) \wedge (\sim X_1 \vee \sim X_3 \vee X_4)$

ההשמה: $(X_1,X_2,X_3,X_4)=(T,F,T,F)$ מספקת פסוקית 1.

וההשמה: $(X_1,X_2,X_3,X_4)=(F,F,F,T)$ מספקת את שתיהן.

<u>הסבר:</u> "יתכן שנמצא השמה של $n-1$ פסוקיות או $\frac{n}{2}$ פסוקיות ומצאנו כן
יש השמה של $n$ פסוקיות ואנחנו לא מסוגלים למצוא אותה.

## אלגוריתם 2-מקורב לבעיית ה-3-SAT

1) נבדוק כמה פסוקיות מסופקות ע"י ההשמה $\vec{X_F}=(F,F,...,F)$ נסמן ב-$f$.

2) נבדוק כמה פסוקיות מסופקות ע"י ההשמה $\vec{X_T}=(T,T,...,T)$ נסמן ב-$t$.

3) נחזיר $\vec{X_F}$ אם $f>t$ ו-$\vec{X_T}$ אחרת.

**הוכחת נכונות:** לכל $C_i$: או ש-$\vec{X_F}$ מספקת אותה או ש-$\vec{X_T}$ מספקת אותה או שתיהן. מכיוון ש-$t+f \ge m$ ומכיוון ש-$\max\{t,f\} \ge \frac{m}{2}$ ולכן מחזירים

השמה שמספקת לפחות חצי מ-$C_1,\dots,C_m$ מכיוון ש-$OPT \le m$ קיבלנו 2-קירוב.

הערות: 1. הטענה היה עובד עבור כל השמה $\vec{x}$ ושלילתה.
2. הטענה לא משתמש בכך שיש 3 ליטרלים בכל פסוקית.
לכן הוא עובד על נוסחת CNF.

לכן ופותר כל בעיית X-SAT

## התרמיל השלם:

נתונים $n$ פריטים עם נפחים $V_1,\dots,V_n$ ומשקלים $W_1,\dots,W_n$.
נתונה הגבלת נפח $V$, כלומר $\sum_i V_i \le V$ ו-$V_i \le V$.
מטרה: למצוא תת הקבוצה של פריטים עם סכום משקלים מקסימלי וסכום נפחים שאינו עולה על $V$.

שאלה ממבחן (1) הראו שהטענה החמדן, אשר בוחר פריטים לפי ערכם הסגולי - $p_i = \frac{w_i}{v_i}$ אינו משיג 2-קירוב.

פתרון: בהינתן $c$ נבנה קלט עם פתרון אופטי' $x^*$ כך שהטענה החמדן משיג פתרון $x$ עם $f(x) < \frac{1}{c}\cdot f(x^*)$.
בהינתן $c$, נבחר $V$ המקיים $V-1 > c$.

דוגמא לקלט שהטענה לא עובד עליו, על כל $c$

נסתכל על הקלט הבא:

| | w | v | p |
|---|---|---|---|
| $x_1$ | 1 | 1 | 1 |
| $x_2$ | $V-1$ | $V$ | $1-\frac{1}{V}$ |

האלג' החמדן יבחר את $X_1$ למרות שהפתרון האופטימלי הוא $X_2$.

לכן $(V-1)\cdot f(x)^{=1} = V-1 = f(x^*)$. כלומר, $f(x) = \frac{1}{V-1} f(x^*) < \frac{1}{c} f(x^*)$.

2) הציעו תיקון לאלג' החמדן, כך שנשיג 2-מקרב.

פתרון: 1. מיין את הפריטים לפי הסדר הסגולי $p_i = \frac{w_i}{v_i}$

2. מצא את ה-k המינימלי $\sum_{i=1}^{k} v_i > V$

3. מבין $X_1 = (\underbrace{1,\dots,1}_{k-1},0,\dots,0)$ ו- $X_2 = (\underbrace{0,\dots,0}_{k-1},\underset{\overset{\uparrow}{k}}{1},\dots,0)$ ונחזיר את הפתרון שממקסם את $f(x) = \langle w,x \rangle$.

חוקיות: הפתרון הוא בחירה בין שני פתרונות חוקיים. $X_1$ חוקי מהמינימליות של k, ו-$X_2$ חוקי בגלל ההנחה על הקלט $v_i \le V$. לכן הפתרון כולו חוקי.

יהי $X^*$ פתרון אופטימלי לבעיה.

טענה: הפתרון $X$ שמוחזר ע"י האלג' מקיים $f(x) \ge \frac{1}{2} f(x^*)$.

הוכחה: נסמן ב-$Z^*$ את הפתרון האופטימלי לבעיית התרמיל השברי באותו קלט. מכיוון שכל פתרון לבעיית התרמיל השלם הוא גם פתרון לשברי, אז $f(x^*) \le f(z^*)$

נזכר כי $z^* = (\underbrace{1,1,\dots,1}_{k-1},\alpha,0,\dots,0)$ (לפי האלג' של הפתרון השברי).

מכאן:

$$f(x^*) \le f(z^*) = \underbrace{\sum_{i=1}^{k-1} w_i}_{f(x_1)} + \underbrace{\alpha w_k}_{\alpha \cdot f(x_2)} \le f(x_1) + f(x_2) \le$$

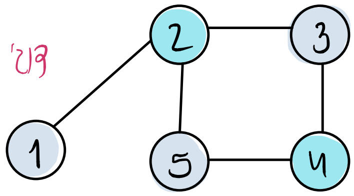
$$2\max\{f(x_1), f(x_2)\} = 2f(x) \implies f(x) \ge \frac{1}{2} f(x^*)$$

זמן ריצה: $O(n\log n)$

## בעיית חתך מקסימום Max-Cut

יהי $G=(V,E)$ גרף לא מכוון. חתך ב-$A,B$ ב-$G$ הוא אוסף הצלעות: $\{(u,v) \mid u\in A, v\in B\}$
כאשר $A,B$ חלוקה של $V$ כך ש: $A\cup B=V$.

דוג' לחתך מקסימלי




**קלט:** גרף $G=(V,E)$ לא מכוון.

**פלט:** חתך $(A,B)$ בו מס' הצלעות מקסימלי!

מדובר שוב בבעיה NP קשה ולכן נמצא אלג' 2-מקרב.

### האלגוריתם

נמספר את הקודקודים $V=(v_1,..,v_n)$

1. נתחיל בחלוקה הטריוויאלית $A=V, B=\emptyset$
2. לכל קודקוד (לפי סדר המספור) נבדוק:

אם מס' השכנים שלו בקבוצה שלו גדול ממש ממספר השכנים שלו בקב' השניה,
נעביר אותו קבוצה.

**הערה:** לא בהכרח מדובר על מעבר מ-A ל-B, לפעמים נעביר בכיוון השני!

3. נחזור על שלב 2, עד שלא "ישארו קודקודים להעביר.

**הוכחת נכונות:** מספר הצלעות בחתך חסום מלמעלה ע"י $|E|$.
בכל העברה של קודקוד, מס' הצלעות בחתך עולה ב-1 לפחות.
לכן האלג' בהכרח עוצר.

**חוקיות:** מתחילים מחתך חוקי $(V,\emptyset)$ ובכל איטרציה מעבירים קודקוד בודד בין הקבוצות. לכן בכל איטרציה נשארים עם חתך חוקי!

נסמן ב- $C(A,B)$ את החתך שמחזיר האלג'.

לכל $v \in V$ נסמן את הדרגה של $v$ ב- $d(v)$ ו-$V_c$ הצלעות שנוגעות ב-$v$ וחוצות $C$.

**טענה:** $|C| \geq \frac{1}{2} OPT$

כל קודקוד נספר פעמיים

אם כך לא היה נכון האלג' לא היה מפסיק

סכום הדרגות שווה $2|E|$

**הוכחה:** $$|C| = \frac{1}{2}\sum_{v\in V} V_c \geq \frac{1}{2}\sum_{v\in V}\frac{1}{2}d(v) = \frac{1}{4}\sum_{v\in V} d(v) = \frac{1}{4}\cdot 2\cdot|E| = \frac{1}{2}|E| \geq \frac{1}{2} OPT$$

**זמן ריצה:** בשלב 2 - האלג' עושה לכל היותר $|E|$ איטרציות.

בכל איטרציה, נעבור על כל קודקוד וכל הצלעות, לכן זמן הריצה הוא $O(|E|(|E|+|V|))$.

יום שלישי 1.6.21

מקרא:
דוגמה - הוכחה -
טענה - משפט -
הגדרה -

# אלגוריתמים
# שבוע 10 - תרגול 10

## Online Learning

בעיות סיווג - Classification

בבעיות אלו המטרה היא לחלק קבוצה של פריטים לתתי-קבוצות.

דוגמה: לסווג תפוחים לקבוצה של תפוחים בשלים וקבוצה של תפוחים לא בשלים.

בתרגול נתמקד בסיווג בינארי, ונסכם מסויים של בעית סיווג - למידת online מערכת מומחים.

הגדרה: ישנם T סיבובים, $t=1,..,T$. בכל סיבוב מקבלים פריט $x_t\in X$.

לכל פריט יש סיווג $y_t\in\{-1,1\}$ שאותו נרצה למצוא. בנוסף, יש לנו קבוצה של n מומחים.

נסמן את המומחים ב- $f_1,..,f_n$ כאשר כל מומחה נותן את דעתו על הסיווג של $x_t$.

באופן פורמלי: $f_i:X\to\{-1,+1\}$, פונ' שמקבלת $x_t$ ומחזירה סיווג.

אנו לא יודעים מראש כמה המומחים טובים.

### נשחק "משחק"

1. נאתחל את מס' הטעויות שלנו, mistakes =0
2. עבור כל אחד מהסיבובים $t=1,..,T$:

א. מקבלים אובייקט מהסביבה $x_t$ ורשימה של N סיווגים מהמומחים: $f_1^t,..,f_N^t$

ב. יוצרים סיווג משלנו $\hat{y}_t$ כפונ' של סיווגי מומחים

ג. מקבלים מהסביבה את הסיווג האמיתי $y_t$

ד. אם $\hat{y}_t\neq y_t$ מעלים ב-1 את mistakes.

הַמטרה: לזכור סיווגים $\hat{y}_1,\dots,\hat{y}_T$ כך שמס' הטעויות הכולל יהיה נמוך כמה שניתן.

## אלגוריתם Halving

בכל סיווג נשתמש רק במומחים שצדקו בכל הסיבובים הקודמים.
מבין הסיווגים האלו, נבחר את מה שהרוב מסכימים עליו.

פורמלית: נגדיר את $experts_t = \{f_i \mid f_i^d = y_d \ \forall d=1,\dots,t-1\}$

כל המומחים שצדקו בכל הסיווגים עד עכשיו.

נחזיר את הסיווג:

$$\hat{y}_t = \begin{cases} 1 & |\{f_i \in experts \mid f_i^t = 1\}| \geq |\{f_i \in experts \mid f_i^t = -1\}| \\ -1 & else \end{cases}$$

טענה: אם יש מומחה מושלם $f_i$ שצודק תמיד, אז מתקיים: $mistakes \leq \log(N)$

הערה: האלג' מניח קיום מומחה שלא טועה. אחרת, נוכל להשאר עם קבוצה ריקה של מומחים.

הוכחה: בכל סיבוב $t$ בו עשינו טעות, רוב המומחים ב-$experts_t$ טעו.
לכן $|experts_{t+1}| \leq \frac{1}{2}|experts_t|$.
מכיוון ש-$|experts_1| = N$, אחרי לכל היותר $\log(N)$ טעויות, נשאר עם קבוצה בגודל 1. מכיוון שהמומחה שנותר הוא מושלם, לא נבצע יותר טעויות.
נשים לב שקיבלנו חסם עליון על כמות הטעויות שהאלג' יכול לעשות.
כמו כן, חסם זה אינו תלוי במס' הסיבובים.
נשים לב שחסם זה הוא הדוק. אם נבחר $N=2^n$ (חבקה של 2 לשהי)

ונחליט שרק אחד הוא משלם. נסדר את הדוגמאות ככה שבכל סיבוב בדיוק חצי יטעו ויתנו דדקו. בהרצה זו, האלג' יעשה בדיוק $\log(N)$ טעויות לפני שנשאר עם קבוצה בגודל 1.

לאמיתה, ניתן להוכיח שזהו שגיאה המינימלית לכל אלג' סוג, תחת ההנחה של מומחה משלם.

לכן אלג' halving משיג את התוצאה הטובה ביותר האפשרית.

אבל - הנחנו הנחה חזקה מאוד והיא שקיים מומחה משלם, וברוב המקרים לא יהיה ככה ולכן נצטרך לפנות לאלג' אחרים.

## אלגוריתם רוב ממושקל

לכל מומחה, $f_i$, ניצור משקולת - $w_i$, שתייצג כמה חשיבות אנו מייחסים לדעתו של המומחה $f_i$. נאתחל את כל המשקולות $w_i=1$ (לכל $i\in[N]$).

בכל סיבוב, נחזיר את הניחוש $\hat{y}_t=\text{sign}\left(\sum_{i=1}^{N}w_if_i^t\right)$

בכל סיבוב, נעדכן את המשקולת של מומחה שטעה להיות: $w_i^{t+1}=\frac{1}{2}w_i^t$

נסמן ב-$\text{mistakes}_i$ את מספר הטעויות של המומחה ה-$f_i$.

הוכחנו מכיתה כי:

$$\forall i\in[N]\quad \underbrace{\text{mistakes}}_{\text{טעויות האלג'}}\leq\frac{1}{\log\left(\frac{4}{3}\right)}\cdot\left(\text{mistakes}_i+\log(N)\right)$$

נשים לב שאם קיים מומחה משלם, מס' הטעויות חסום ע"י

$$\text{mistakes}\leq\frac{1}{\log\left(\frac{4}{3}\right)}\cdot\left(0+\log(N)\right)\approx 2.41\cdot\log(N)$$

כלומר, טוב במצב פחות טוב מטעמי halving אבל קיבלנו ערך שעדיין טוב גם אם אין מומחה מושלם.

הערה: נשים לב שמכיוון ש-$\hat{y}_t \in \{-1,+1\}$ צריך לשנות את הסוף sign ככה שיתפזר 1 או -1 במקרה של sign(0), לכן נבחר שרירותית במקרה כזה.

## משקולות כפליים

נכליל מעט את הבעיה - במקום לספור לכל מומחה כמה טעויות הוא ביצע, נגדיר סוג של הסיווג של כל מומחה בכל סיבוב. $(Punishment)_i^t \in [-1,+1]$. מומחה $i$, סיבוב $t$.
נשים לב שניתן לתת קנס שלילי, וזהו פרס, על תשובה נכונה.
נשים לב שזוהי הכללה של הבעיה הקודמת. אם נגדיר:

$$(Punishment)_i^t = \begin{cases} 1 & f_i^t \neq y_t \\ 0 & \text{else} \end{cases}$$

נחזור לבעיה המקורית.

כעת נמדוד את עצמנו לעומת המומחה שקיבל את הקנס הכי נמוך:

$$\min_i \sum_{t=1}^{T} (Punishment)_i^t = \sum_{t=1}^{T} (Punishment)_{OPT}^t$$

נרצה להגדיר כמה קנס אנחנו נשלם עבור הסיווגים שלנו.
כדי לעשות זאת, במקום לבחור סיווג על פי רוב המומחים, נעבור להקשיב למומחה יחיד בכל סיבוב. בכל סיבוב, נקבל את הקנס של המומחה שלו הקשבנו.
במשחק - במקום הפונ' של סיווג המומחים, $\hat{y}_t$ נייצר אחרת.

## האלגוריתם:

לכל מומחה נחזיק משקולת $w_1,\dots,w_N$. נאתחל $w_i=1$ לכל $i\in[N]$.

בתשובה לבחור מומחה להקשיב לו בסיבוב $t$, נבחר מומחה מקבוצת המומחים לפי ההתפלגות: $p_i^t=\frac{w_i^t}{\sum_{j=1}^{T}w_j^t}$ (ככה נקבל הסתברות יותר גבוהה לבחור במומחה ש-$w$ שלו גדול)

תרגיל: נשתכנע ש- $\sum_{i=1}^{N}p_i^t=1$.

בגלל שבכל סיבוב אנו מגרילים מומחה שלו נקשיב, אנו יכולים לקבל קנסות שונים כתלות באופן שבו דגמנו.

לכן, במקום להסתכל על הקנס הספציפי שקיבלנו בהרצה מסוימת, נתעניין בתוחלת הקנס על פני $T$ הסיבובים.

$$\underset{\text{(our punishment)}}{OP}=\sum_{t=1}^{T}\mathbb{E}_{p^t}[OP^t]=\sum_{t=1}^{T}\sum_{i=1}^{N}\underbrace{p_i^t\cdot(\text{Punishment})_i^t}_{\text{ההסתברות לקבל את הערך x הקנס}}$$

נסמן את הקנס בסיבוב ה-$t$ של המומחה הטוב ביותר ב-$(\text{Punishment})_{OPT}^t$.

אנחנו רוצים ש-$OP$ יהיה קרוב ככל הניתן ל-$\sum_{t=1}^{T}(\text{Punishment})_{OPT}^t$.

בכיתה הגדרנו את כלל העדכון הבא עבור המשקולות:

$$\underset{\text{המשקולות של המומחה ה-}i\text{ בסיבוב ה-}t+1}{}\rightarrow w_i^{t+1}=(1-\varepsilon\cdot(\text{Punishment})_i^t)\cdot w_i^t \qquad \left(0<\varepsilon\le\tfrac{1}{2}\right)$$

אם קיבלתי קנס חיובי (עונש), אז המשקולת תקטן, נתייחס פחות למומחה הנ"ל.

אם קיבלתי קנס שלילי (פרס), אז המשקולת תגדל, נתייחס יותר למומחה הנ"ל.

בכיתה הוכחנו שבעזרת כלל עדכון זה ניתן לחסום את OP ע"י:

$$\boxed{OP \le \sum_{t=1}^{T} (\text{Punishment}_t)_{OPT}^{t} + \varepsilon \cdot \sum_{t=1}^{T} |(\text{Punishment}_t)_{OPT}^{t}| + \frac{1}{\varepsilon} \cdot \log(n)}$$

נשים לב - אם נחזור לטבלה הקודמת (שבה) $(\text{Punishment})_i^t = \mathbb{1}[f_i^t \neq y_t]$

נוכל לחסום את מס' השגיאות ע"י:

$$OP \le (1+\varepsilon) \cdot \text{Mistakes}_{OPT} + \frac{1}{\varepsilon} \log(N)$$

כלומר, שאם קיים מומחה מושלם ו- $\varepsilon = \frac{1}{2}$ נקבל ש:

$$OP \le \frac{1}{\frac{1}{2}\log(N)} = 2 \cdot \log(N)$$

שזה טוב יותר מהאלגוריתם הרוב-הממושקל.

<u>מסקנה:</u> הוספת הבחירה ההסתברותית גרמה לנו לפעול באופן טוב יותר מתוחלת.

יום שלישי 8.6.21

מקרא:
הוכחה - ● דוגמה - ●
משפט - ● טענה - ●
הגדרה - ●

# אלגוריתמים
# שבוע 11 - תרגול 11

## אלגוריתמים הסתברותיים

הגדרה: אלגוריתם הסתברותי הוא אלגוריתם אשר במהלך ריצתו משתמש ב-"הטלות מטבע". כלומר, מבצע הגרלות במהלך הריצה.

## בעיית צבצוע - חיפוש מספרים במערך

קלט: מערך $A=[a_1,\dots,a_n]$ אשר חצי מהערכים בו הם 0 וחצי 1.

פלט: אינדקס i כלשהו עבורו $a_i=1$

פתרון נאיבי: נעבור על המערך לפי הסדר עד שנמצא אינדקס עם $a_i=1$ ונחזיר אותו. זמן ריצה - במקרה הגרוע ייקח לנו $O(n)$.

נראה כעת שניתן לשפר את זמן הריצה בעזרת הגרלות.

## סוגי אלגוריתמים הסתברותיים

נבחין בין 2 סוגים עיקריים של אלג' הסתברותיים:

### 1. אלגוריתמי לאס-וגאס

אלגוריתמים שבהם הטלות המטבע אינן משפיעות על נכונות הפלט.

פתרון לאס-וגאס לבעיית הצבצוע:

1. נגריל באקראי אינדקס $1\le i\le n$
2. אם $a_i=1$ נחזיר i, אחרת נחזור לשלב 1 (לא מחזירים את i ל-"ערימה").

* נשים לב שהאלגוריתם משתמש בהגרלות ולכן הוא הסתברותי, אך תמיד יחזיר פלט חוקי (נכון).

**זמן ריצה**- במקרה הגרוע, הגרלנו ב-$\frac{n}{2}$ ההגרלות הראשונות אינדקס $i$ כך ש- $a_i=0$. לכן זמן הריצה במקרה הגרוע ביותר הוא עדיין $O(n)$. עם זאת, מכיוון שמגרילים אינדקסים לגמרי באקראיות, הקלט הנתון אינו משנה. הסיכוי להצליח בכל הגרלה בודדת הוא לכל הפחות $\frac{1}{2}$ בתוחלת, ונצטרך לכל היותר 2 הגרלות בלבד. על כן, זמן הריצה של האלג' הוא $O(1)$ בתוחלת (או amortized running time). זהו שיפור משמעותי על פני האלגוריתם הנאיבי.

**הערה:** בקורסים קודמים ראיתם דוגמאות לאלג' מסוג זה (las vegas), למשל quick-sort. בקורס לא נתמקד באלג' כאלו.

## 2. אלגוריתמי מונטה-קרלו

אלגוריתמים שבהם הטלות המטבע משפיעות על נכונות הפלט. בפרט, ריצות שונות על הקלט יכולות להסתיים בתוצאה שונה גם בהיבט הנכונות. כלומר, ייתכן והאלגוריתם "יכשל".

**פתרון מונטה-קרלו לבעיית הדוגמה:**

1. נגריל באקראי אינדקס $1\le i\le n$.
2. אם $a_i=1$ נחזיר $i$, אחרת נחזיר fail

**הערה:** בניגוד לאלג' שראינו עד כה בקורס, בהם דרשנו שהפתרון יהיה חוקי תמיד, האלג' הבא מחזיר לעתים פתרון חוקי ולעתים לא. במקרה שלנו

האלג' החזיר fail ולא כן אנו יודעים לא להשתמש בתוצאה. אם במקרה אחר הוא עלול להחזיר כל דבר וזו עלולה להיות שגויה.

זמן ריצה- מבצעים הגרלה אחת ולכן זמן הריצה הוא $O(1)$.
האלג' רץ בזמן טוב משמעותית מהאלג' הנאיבי עבור קלט, אם כה מה לא מחיר: מדי פעם האלג' כושל ולא מחזיר את התשובה הרצויה.

נתמודד עם כה באופן הבא: נדרוש תמיד כי אלג' הסתברותיים יחזירו תשובה נכונה עבור קלט בהסתברות "גדולה כרצוננו".
דרך מקובלת להגדיר "גדולה כרצוננו" היא שהאלג' צודק בהסתברות של לפחות $1-\frac{1}{e^k}$ עבור $k\in\mathbb{N}$ נתון. באופן כה, אנחנו משאירים למשתמש לבחור כמה חשובה לו חוקיות התשובה. ככל שירצה דיוק גדול יותר, בד"כ יצטרך לשלם בזמן ריצה גדול יותר.

הערה: אלג' מונטה-קרלו נפוצים במיוחד בקרב בדיקות ראשוניות (Primary Test) המאפשרים לבדוק ראשוניות (בהס' גדולה) בזמן לוגריתמי. אלג' אלה נמצאים בשימוש נרחב עבור הצפנת מידע (אלג' מילר-רבין).

## ניפוח אלגוריתמים הסתברותיים

אחת החוזקות של אלג' הסתברותיים היא שניתן בקלות יחסית "לנפח" את ההסתברות לקבלת תשובה נכונה ע"י הרצה חוזרת, כדי להמחיש זאת נסתכל שוב על הדוגמה ממקודם:

## פתרון עם הסתברות גדולה כרצוננו לבעיית הצלעות:

בהינתן $K\in\mathbb{N}$, נריץ את האלג' הקודם ♡ $\log_2(e)\cdot K$ פעמים.
אם קיבלנו לפחות פעם אחת אינדקס $i$ מתאים, נחזיר אותו.
אחרת, נחזיר fail.

**טענה:** לכל קלט, האלג' טועה בהסתברות של לכל היותר $\frac{1}{e^K}$.

**הוכחה:** במערך ישנם $\frac{n}{2}$ אפסים. ההסתברות לקבל אינדקס $i$ עם $a_i=0$ היא $\frac{1}{2}$. מכיוון שכל ההגרלות ב"ת ההסתברות שכל הדגימות יהיו עם $a_i=0$ היא:

$$Pr\left(\text{כל הפעמים האלג' יחזירו fail}\right)=\prod_{i=1}^{\log(e)\cdot K}\left(\frac{1}{2}\right)=\left(\frac{1}{2}\right)^{\log_2(e)\cdot K}=\frac{1}{2^{\log_2(e)\cdot K}}=\frac{1}{e^K}$$

זמן ריצה - אנחנו מריצים $O(K)$ פעמים אלג' שרץ בזמן $O(1)$. לכן, זמן הריצה הכולל הוא $O(K)$. נשים לב, שגם עבור $n$ גדול מאוד, מספיק $K$ קטן כדי לקבל הסתברות שהיא פרקטית 1.

## אלגוריתמי קירוב הסתברותיים

* אלג' מקרב-C: הוא אלג' שלכל קלט מחזיר פתרון חוקי ו-C מקרב לפתרון האופטימלי.
* אלג' הסתברותי: הוא אלג' שלכל קלט, בזמן פולינומיאלי מחזיר בהסתברות גדולה כרצוננו פתרון חוקי אופטימלי, ובהסתברות נמוכה נכשל.

* <u>אלג' קירוב הסתברותי</u> הוא אלגוריתם שלכל קלט, בזמן פולינומיאלי מחזיר פלט חוקי ו-C-מקרב לפתרון האופטימלי בהסתברות גדולה כרצוננו, והסתברות נמוכה נכשל.

## בעיית ה-MAX 3-SAT

<u>תזכורת:</u>

* <u>פסוקית של נוסחת 3-CNF</u> - אוסף של 3 פסוקיות ליטרלים ואיווי ביניהם.
  דוגמא - $(x_1 \vee \sim x_2 \vee x_4)$
* <u>נוסחת 3-CNF</u> - גימום של פסוקיות.
  דוגמא: $(x_1 \vee \sim x_2 \vee x_4) \wedge (x_3 \vee \sim x_2 \vee \sim x_5)$
* <u>קלט:</u> נוסחת 3-CNF בעלת m פסוקיות: $C_1, \dots, C_m$ מעל n המשתנים $x_1, \dots, x_n$.
* <u>פלט:</u> האם קיימת השמה ל-$x_1, \dots, x_n$ שממקסמת את מס' הפסוקיות שמסתפקות (מקבלות ערך אמת).

<u>הנחה:</u> המשתנים בכל פסוקית שונים זה מזה.

בתרגול הקודם ראינו אלג' נאיבי 2-מקרב לבעיה.
היום נבנה אלג' הסתברותי שנותן $\frac{8}{7}$-קירוב.

## אלגוריתם הסתברותי $\frac{8}{7}$-מקרב ל-MAX 3-SAT

בהינתן $k \in \mathbb{N}$ נבנה אלגוריתם שיצליח בהסתברות $1-\frac{1}{e^k}$.
בנוסף, זמן הריצה יהיה פולינומי ב-K.

## אלגוריתם בסיסי:

1. לכל משתנה $x_i$ נטיל מטבע הוגן
   - אם יצא פלי, נגדיר $x_i = T$
   - אם יצא פלי, נגדיר $x_i = F$
2. אם ההשמה שהגרלנו מספקת לפחות $\frac{7}{8}m$ מהפסוקיות - נחזיר אותה.
   אחרת, נחזיר fail.

__זמן ריצה:__ עלינו להגריל n משתנים באופן הסתברותי ולאחר ההגרלות לבדוק כמה פסוקיות מקבלות ערך אמת ע"י ההשמה שהוגרלה. סה"כ $O(n+m) = O(m)$ (כי הנחנו $n \leq 3m$).

## אלגוריתם שומתי:

נריץ את האלג' הבסיסי $k(m+1)$ פעמים באופן בלתי תלוי. אם באחת ההרצות הייתה הצלחה, נחזיר את ההשמה שהתקבלה ואחרת נחזיר fail.

__זמן ריצה:__ האלג' הבסיסי לוקח ב-$O(m)$. לכן זמן הריצה הכולל הוא:
$O(k(m+1)\cdot m) = O(k\cdot m^2)$

__נכונות:__ צ"ל שהאלג' הבסיסי מצליח בהסתברות מספיק גבוהה, ושבמידה והוא מצליח, הוא גם $\frac{8}{7}$-מקרב.

__קירוב:__ אם האלג' מצליח, אז הוא מחזיר השמה שמספקת $\frac{7}{8}m$ פסוקיות.
מכוון נובע שהוא באמת $\frac{8}{7}$-מקרב כי $m \geq OPT$ ולכן $\frac{7}{8}m \geq \frac{7}{8}OPT$.

לכן נותר להראות את ⊛, לשם כך נוכיח את הטענה הבאה:

**טענה:** סיכויי ההצלחה של האלג' הבסיסי $\geq \frac{1}{m+1}$.

**הוכחה:** במהלך ההוכחה נעבור 2 שלבים ונעזר בלמה.

**שלב ראשון:** נראה שתוחלת מס' הפסוקיות המסתפקות בהשמה המקורית הוא $\frac{7}{8}m$. (בהמשך נראה שניתן לתרגם את הבטחות על התוחלת לסיכויי הצלחה)

נגדיר מ"מ $X$ כאשר $X(w)$ הוא מספר הפסוקיות שמסתפקות ע"י ההשמה $w$.

נגדיר לשם כך את $X_i$ מ"מ אינדיקטור:

$$X_i(w)=\begin{cases}1 & w(C_i)=T\\ 0 & w(C_i)=F\end{cases}$$

נחשב את $E[X]$. לשם כך נבטא את $X=\sum_{i=1}^{m}X_i$.

$$E(X_i)\overset{\text{תוחלת של אינדיקטור}}{=}0\cdot P(X_i=0)+1\cdot P(X_i=1)=P(X_i=1)=1-P(X_i=0)\overset{\text{המשתנים שונים ומוגרלים ב"ת}}{=}1-\left(\frac{1}{2}\right)^3=\frac{7}{8}$$

ומכאן ש:

$$E[X]=E\left[\sum_{i=1}^{m}X_i\right]\overset{\text{לינאריות התוחלת}}{=}\sum_{i=1}^{m}E(X_i)=\sum_{i=1}^{m}\frac{7}{8}=\frac{7}{8}m$$

**הערה:** מחשבה כזו, נובע שתמיד קיימת השמה המספקת לפחות $\frac{7}{8}$ מהפסוקיות.

**שלב שני:** נראה שסיכויי ההצלחה של האלג' הבסיסי לכל הפחות $\frac{1}{m+1}$.

לשם כך נשתמש בטוש מרקום.

<u>נזכיר באוש מרקוב</u>: יהי $X$ מ"מ אי-שלילי ויהי $a>0$ אזי $P(X\geq a)\leq\frac{E[X]}{a}$

נגדיר מ"מ חדש, $Y$, כאשר $Y(\omega)$ הוא מס הפסוקיות <u>שלא</u> הסתפקו ע' $\omega$. נשים לב כי מתקיים $Y=m-X$ ולכן: $E[Y]=\frac{1}{8}m$

<u>למה</u>: יהי $Y$ מ"מ המקבל ערכים שלמים. אזי לכל $c,m\in\mathbb{N}$ מתקיים:

$$Y>\frac{1}{c}\cdot m\iff Y\geq\frac{1}{c}m+\frac{1}{c}$$

<u>מוטיבציה</u>: שימוש מועיל באוש מרקוב

<u>הוכחה</u>:

$$Y>\frac{1}{c}\cdot m\iff c\cdot Y>m\iff c\cdot Y\geq m+1\iff Y\geq\frac{1}{c}\cdot m+\frac{1}{c}$$

מהלמה מתקיים:

נוח לשימוש במרקוב

$$P(\text{האלגוריתם נכשל})=P\left(Y>\frac{1}{8}m\right)\overset{\text{למה}}{=}P\left(Y\geq\frac{1}{8}m+\frac{1}{8}\right)=$$

$$P\Big(\overset{\geq0}{Y}\geq\underbrace{\tfrac{1}{8}m\left(1+\tfrac{1}{m}\right)}_{a}\Big)\overset{\text{מרקוב}}{\leq}\frac{E[Y]}{\frac{1}{8}m\left(1+\frac{1}{m}\right)}\overset{\text{הראינו בשלב הקודם}}{=}\frac{\cancel{\frac{1}{8}}m}{\cancel{\frac{1}{8}}m\left(1+\frac{1}{m}\right)}=\frac{m}{m+1}$$

ומכאן, כאשר נסתכל על המשלים:

$$P(\text{האלגוריתם מצליח})=1-P(\text{האלגוריתם נכשל})\geq1-\frac{m}{m+1}=\frac{1}{m+1}$$

<u>מסקנה</u>: האלג' הכללי מצליח בהסתברות של הפחות $1-\frac{1}{e^{k}}$

**הוכחה:** מהטענה שהוכחנו מתקיים

$$P\left(\text{האלגוריתם הכללי נכשל}\right) = P\left(\text{האלגוריתם הבסיסי נכשל } k(m+1) \text{ פעמים}\right) = P\left(\text{האלגוריתם הבסיסי נכשל}\right)^{(m+1)k}$$

$$\leq \left(1-\frac{1}{m+1}\right)^{(m+1)\cdot k} = \left(\underbrace{\left(1-\frac{1}{m+1}\right)^{m+1}}_{\text{שואפת ל-}\frac{1}{e}}\right)^{k} < \left(\frac{1}{e}\right)^{k} = \frac{1}{e^{k}}$$

לכן האלגוריתם הכללי מצליח בהסתברות של לפחות $1-\frac{1}{e^k}$, כנדרש.

15.6.21 יום שלישי

מקרא:

הוכחה - ; דוגמה - ; משפט - ; טענה - ; הגדרה -

# אלגוריתמים

## שבוע 12 - תרגול 12

### בעית פתרון מערכת משוואות מודולו 2- Max 2-Lin

**קלט:** מטריצה $A$ מסדר $m\times n$ מעל השדה $\mathbb{F}_2$, ווקטור $b$ באורך $m$ מעל השדה $\mathbb{F}_2$ המייצגים מערכת משוואות לינארית $Ax=b$ עם $x=(x_1,..,x_n)^T$ וקטור משתנים.

**פלט:** השמה למשתנים $x_1,...,x_n$ המספקת מס' מקסימלי של משוואות.

**דוגמה:** $m=4$, $n=2$

$$\underbrace{\begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}}_{x} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}}_{b}$$

**נבחין:** למערכת אין פתרון משום ש-

$$\left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = 0$$

$$\left\langle \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = 0$$

$$\left\langle \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = 1$$

$$1 = \left\langle \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = \left\langle \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = \left\langle \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle + \left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \right\rangle = 0+0$$

ההשמה $x_1=0, x_2=0$ מספקת 3 משוואות וזהו המס' המקסימלי.

בעיה זו היא NP-קשה ונלמד אלג' הסתברותי 2-קירוב.

**תזכורת:** אלג' קירוב הסתברותי צריך לכל קלט, בהסתברות גדולה

כבר ראינו ובזמן פולינומי, להחזיר פתרון חוקי ו-C-מקרב לבעיה ובהסתברות קטנה עלול להכשל (להחזיר פתרון לא חוקי / לא C-מקרב..)

## אלג' 2-מקרב ל-Max 2Lin

### אלג' בסיסי:

1. נגריל השמה מקרית $S \in \mathbb{F}_2^n$
(לכל משתנה $x_i$ נגריל $x_i = 1$ בהסתברות $\frac{1}{2}$ ו-$x_i = 0$ בהסתברות $\frac{1}{2}$).
2. אם $S$ מספקת $\frac{1}{2}$ מהמשוואות החזר אותה, אחרת החזר fail.

נראה שהאלג' נכשל בהסתברות קטנה שווה מ-$\frac{m}{m+1}$.

בשלב הראשון נחשב את תוחלת מס' המשוואות המסתפקות ע"י השמה מקרית.

נגדיר מס' סימונים שיעזרו לנו: * נסמן ב-$r_i$ את השורה ה-i של מטריצה $A$.
* נסמן ב-$b_i$ את הקאור' ה-i של הוקטור $b$.

נאמר שהמשוואה ה-i מסופקת ע"י השמה $S$ אמ"מ $\langle r_i, S \rangle = b_i$.

* נסמן ב-$X$ את המ"מ שסופר את מס' המשוואות המסופקות ע"י $S$.
* נגדיר $X_i$ מ"מ אינדיקטור לאירוע המשוואה ה-i סופקה ע"י $S$.

נחשב את תוחלת $X_i$:

נניח בה"כ כי $b_i = 1$ עבור השורה ה-i ($r_i$) נסמן ב-$d$ את הקואור' הגדולה ביותר ששונה מ-0.

$$\mathbb{E}(X_i) = \mathbb{P}(X_i = 1) = \mathbb{P}(\langle r_i, S \rangle = b_i) =$$

מאורעות בת"ל

$$\mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 0 \wedge v_{id}s_d = 1\right) + \mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 1 \wedge v_{id}s_d = 0\right) =$$

$$\mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 0\right)\cdot \mathbb{P}(v_{id}s_d = 1) + \mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 1\right)\cdot \mathbb{P}(v_{id}s_d = 0) =$$

$$\frac{1}{2}\left(\mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 0\right) + \mathbb{P}\left(\sum_{j=1}^{d-1} r_{ij}s_j = 1\right)\right) = \frac{1}{2}\cdot 1 = \frac{1}{2}$$

לכן $\mathbb{E}[X] = \mathbb{E}[\sum X_i] = \frac{1}{2}m$.

נגדיר $Y = m - X$ כמות המשוואות שלא מסתפקות. נשים לב $E(Y) = \frac{1}{2}m$

נזכר באי"ש מרקוב: יהי $X$ מ"מ אי-שלילי ויהי $a>0$ אזי $p(X \geq a) \leq \frac{E[X]}{a}$

נזכר בלמה מתרגול שלמדנו:

> **למה:** יהי $Y$ מ"מ המקבל ערכים טבעיים. אזי לכל $c, m \in \mathbb{N}$ מתקיים:
>
> $$Y > \frac{1}{c}\cdot m \Longleftrightarrow Y \geq \frac{1}{c}\cdot m + \frac{1}{c}$$
>
> מוטיבציה: שימוש מועיל באי"ש מרקוב
>
> **הוכחה:**
>
> $$Y > \frac{1}{c}\cdot m \Longleftrightarrow c\cdot Y > m \Longleftrightarrow c\cdot Y \geq m+1 \Longleftrightarrow Y \geq \frac{1}{c}\cdot m + \frac{1}{c}$$

ומכאן:

$$\mathbb{P}\left(Y > \frac{1}{2}m\right) \overset{\text{כישלון}}{=} \mathbb{P}\left(Y \geq \frac{1}{2}m + \frac{1}{2}\right) = \mathbb{P}\left(Y \geq \frac{1}{2}m\left(1 + \frac{1}{m}\right)\right) \overset{\text{מרקוב}}{\leq} \frac{1}{1+\frac{1}{m}} = \frac{m}{m+1}$$

**אלגוריתם כללי:** נריץ את האלגוריתם הבסיסי $D$ פעמים. אם באחת הפעמים קיבלנו השמה $S$ שמספקת $\frac{1}{2}$ מהמשוואות נחזיר אותה, אחרת נחזיר fail.

נחשב כעת את ההסתברות להצלחה:

$$P\left(\text{האלגוריתם הכללי נכשל}\right)=P\left(\text{האלגוריתם הבסיסי נכשל } D \text{ פעמים}\right)=P\left(\text{אלגוריתם בסיסי נכשל}\right)^D$$

$$\le\left(\frac{m}{m+1}\right)^D=\left(1-\frac{1}{m+1}\right)^D=\left(\left(1-\frac{1}{m+1}\right)^{m+1}\right)^{\frac{D}{m+1}}<\left(\frac{1}{e}\right)^{\frac{D}{m+1}}=e^{-\frac{D}{m+1}}$$

שואפת ל-$\frac{1}{e}$

נמצא את $D$: $D=K(m+1)\Longleftrightarrow\frac{D}{m+1}=K$, נציב ונקבל $\frac{1}{e^K}$.

לכן האלגוריתם הכללי מצליח בהסתברות של הפחות $1-\frac{1}{e^K}$, כנדרש.

סמן ריצה: האלג' הבסיסי רץ ב-$O(m\cdot n)$, מריצים אותו $O(K\cdot m)$ פעמים
לכן סה"כ זמן הריצה הכללי: $O(K\cdot m^2\cdot n)$.

## צביעה מקסימלית של גרף ב-3 צבעים

בעיית הצביעה: קלט: גרף לא מכוון $G=(V,E)$
פלט: מס' הצבעים המינימלי שניתן לצבוע את קודקודי $G$ כך שאין זוג של קודקודים שכנים בו אותו חולקים צבע.

בעיה זו היא NP-שלמה ללא פתרון פולינומי, לכן נציג תת-בעיה שלה שגם היא NP-שלמה:

בעיה אחרת: קלט: גרף לא מכוון $G=(V,E)$
פלט: האם ניתן לצבוע את קודקודי $G$ ב-3 צבעים כך שאין זוג קודקודים שכנים שצבועים באותו הצבע.

## ננסה לקרב אותה כי היא NP-שלמה:

**קלט:** גרף לא מכוון $G=(V,E)$

**פלט:** צביעה של הקודקודים ב-3 צבעים כך שמס' הצלעות שקודקודיהן אינם צבועים באותו צבע מקסימלי.

**באופן פורמלי:** רוצים למצוא פונקציה $c:V\to\{1,2,3\}$ שממקסמת את $A=\{(i,j)\in E \mid c(i)\neq c(j)\}$ נשים לב שאם $|A|=|E|$ אז יש פתרון לבעיה האחרת.

## צביעה מקסימלית של גרף ב-3 צבעים - אלג' הסתברותי $\frac{3}{2}$-מקרב

**אלג' בסיסי:**

1. עבור כל קודקודי הגרף ולכל קודקוד נגריל צבע בהסתברות אחידה $(\frac{1}{3})$.
2. נחשב את $A$ ע"י מעבר על כל $E$ ואם $|A|\geq\frac{2}{3}|E|$ נחזיר את הצביעה $c$ אחרת נחזיר fail.

**טענה:** האלג' הבסיסי מצליח בהסתברות לפחות $\frac{1}{|E|+1}$.

**הוכחה:** נסמן ב-$X$ את כמות הצלעות ב-$A$. $X=\sum_{(i,j)\in E}\mathbb{1}[c(i)\neq c(j)]$

נשים לב כי $\mathbb{1}[c(i)\neq c(j)]$ היא פונ' אינדיקטור, המקבלת 1 אמ"ם $(i,j)\in A$.

נניח בה"כ $c(i)=1$. ההסתברות לכך היא:

$$\mathbb{P}((i,j)\in A)=\mathbb{P}(c(j)=2\vee c(j)=3)=\mathbb{P}(c(j)=2)+c(j)=3)=\frac{2}{3}$$

אם כך, $\mathbb{E}(X)=\frac{2}{3}|E|$, נגדיר $Y=|E|-X$ אזי $\mathbb{E}[Y]=\frac{1}{3}|E|$

נשים לב שהאלג' הבסיסי נכשל אומ"ר $Y > \frac{1}{3}|E|$

וזה קורה (לפי הלמה) אומ"ר $Y \ge \frac{1}{3}|E| + \frac{1}{3}$. ומכאן:

$$\mathbb{P}(Y \overset{\text{כישלון}}{>} \tfrac{1}{3}|E|) = \mathbb{P}(Y \ge \tfrac{1}{3}|E| + \tfrac{1}{3}) = \mathbb{P}\left(Y \ge \tfrac{1}{3}|E| \cdot \left(1+\tfrac{1}{|E|}\right)\right) \overset{\text{מרקוב}}{\le} \frac{1}{1+\frac{1}{|E|}} = \frac{|E|}{|E|+1}$$

$$\mathbb{P}(\text{אלג' בסיסי מצליח}) = 1 - \mathbb{P}(\text{אלג' בסיסי נכשל}) \ge 1 - \frac{|E|}{|E|+1} = \frac{1}{|E|+1}$$

**אלגוריתם כללי:** נריץ את האלגוריתם הבסיסי $D$ פעמים. אם באחת הפעמים קיבלנו צביעה עבורה $|A| \ge \frac{2}{3}|E|$ נחזיר אותה, אחרת נחזיר fail.

**טענה:** האלג' מצליח בהסתברות לפחות $1 - \frac{1}{e^{\frac{D}{|E|+1}}}$.

**הוכחה:**

$$\boxed{\mathbb{P}(\text{האלג' הכללי מצליח})} = 1 - \mathbb{P}(\text{האלגוריתם הכללי נכשל}) = 1 - \mathbb{P}(\text{האלגוריתם הבסיסי נכשל } D \text{ פעמים}) =$$

$$1 - \mathbb{P}(\text{אלגוריתם בסיסי נכשל})^D \ge 1 - \left(\frac{|E|}{|E|+1}\right)^D = 1 - \left(\frac{1}{|E|+1}\right)^D = 1 - \left(\left(\frac{1}{|E|+1}\right)^{|E|+1}\right)^{\frac{D}{|E|+1}}$$

$$\boxed{\ge 1 - \frac{1}{e^{\frac{D}{|E|+1}}}}$$

שאיפה

ל-$\frac{1}{e}$

נמצא את $D$: $D = K(|E|+1) \iff \frac{D}{|E|+1} = K$ נציב ונקבל $\frac{1}{e^K}$.

לכן האלגוריתם הכללי מצליח בהסתברות לכל הפחות $1 - \frac{1}{e^K}$, כנדרש.

**זמן ריצה:** כל צביעה לוקחת $O(|V|)$. חישוב $A$ לוקח $O(|E|)$.

לכן האלג' הבסיסי לוקח $O(|E|+|V|)$, חוזרים על זה $O(K\cdot|E|)$ פעמים לכן

זמן הריצה סה"כ $O(K\cdot|E|^2 + K\cdot|V|\cdot|E|)$.

**תזכורת:** פולינום מדרגה $d$ מעל שדה $\mathbb{F}$ במשתנה $X$ הוא ביטוי מהצורה:

$$p(x)=\sum_{i=1}^{d} a_i x^i$$

כאשר $d\in\mathbb{N}$, $a_i\in\mathbb{F}$. ביטוי מהצורה $a_i x^i$ נקראו מונום.

אפשר להכליל את הגדרת הפולינום לפונקציה מעל $n$:

$$P(x_1,\dots,x_n)=\sum_{\substack{\alpha_1,\dots,\alpha_n\in\\(d_1,\dots,d_n)}} a_{\alpha_1,\dots,\alpha_n} x_1^{\alpha_1} x_2^{\alpha_2} x_3^{\alpha_3}\cdot\ldots\cdot x_n^{\alpha_n}$$

**דוגמה:**

$$P(x_1,x_2)=3\cdot x_1+5x_1^2x_2^2-x_1\cdot x_2^2$$

במקרה כזה: $a_{1,1}=0,\ a_{1,0}=3,\ a_{2,2}=5,\ a_{1,2}=-1$

**הגדרה:** דרגה של פולינום מעל $n$ משתנים היא סכום החזקות במונום המקסימלי.

**פורמלית:**

$$\deg(p(x_1,\dots,x_n))=\max_{\substack{\alpha_1,\dots,\alpha_n\in\\(d_1,\dots,d_n)}}\left\{\sum_{j=1}^{n}\alpha_j \;\middle|\; a_{\alpha_1},\dots,a_{\alpha_n}\neq 0\right\}$$

כלומר, המקסימלי מבין המונומים שלא מתאפסים.

15.6.21 יום שלישי

מקרא:

הוכחה -
דוגמה -
משפט -
טענה -
הגדרה -

# אלגוריתמים
## שבוע 13 - תרגול 13

הגדרה: פולינום בעל $n$ משתנים מעל שדה $\mathbb{F}$ היא פונקציה $p: \mathbb{F}^n \to \mathbb{F}$

מהצורה: $p(x_1,\dots,x_n) = \sum_{i_1=0}^{d_1},\dots,\sum_{i_n=0}^{d_n} a_{i_1,\dots,i_n} x_1^{i_1}\cdot\ldots\cdot x_n^{i_n}$ כאשר $d_1,\dots,d_n \in \mathbb{N}$.

כל איבר מהצורה $a_{i_1,\dots,i_n} x_1^{i_1}\cdot\ldots\cdot x_n^{i_n}$ נקרא <u>מונום</u>.

בכלל, נתעלם ממונומים עם מקדם 0.

הגדרה: דרגה של מונום (עם מקדם שונה מ-0) היא סכום החזקות של המשתנים בו:

$$\deg(a_{i_1,\dots,i_n} x_1^{i_1}\cdot\ldots\cdot x_n^{i_n}) = \sum_{j=1}^{n} i_j$$

הגדרה: דרגה של פולינום היא המקסימום של סכום הדרגות של המונומים בפולינום.

$$\deg(p(x_1,\dots,x_n)) = \max_{i_1,\dots,i_n}\left\{\sum_{j=1}^{n} i_j \;\middle|\; a_{i_1,\dots,i_n} \neq 0\right\}$$

דוגמה: $p(x_1,x_2) = 3x_1 + 5x_1^2x_2^2 - x_1x_2^2$

$a_{1,0} = 3,\ a_{1,2} = 1,\ a_{2,2} = 5,\ a_{0,2} = 0$

הגדרה: שורש של פולינום $p$ הוא איבר $x \in \mathbb{F}^n$ שעבורו מתקיים $p(x) = 0$.

נגיד כי פולינום הוא פולינום האפס אם לכל $x \in \mathbb{F}^n$ מתקיים $p(x) = 0$.

טענה: יהי $\mathbb{F}$ שדה בעל $|\mathbb{F}|$ איברים. יהי פולינום $p$ המקיים $\deg(p) \leq |\mathbb{F}|$.

הפולינום $p$ הוא פולינום האפס אם"ם כל המקדמים בו הם אפסים.

נשים לב, שבייצוג המקדמים קל לראות אם פולינום הוא פולינום האפס.
אבל, לא תמיד פולינומים נתונים בייצוג זה.

דוגמה: $p(x_1,x_2) = (x_1+3)(x_2-5) + (x_1+2)(x_2+4) + x_1 - x_2 + 7$

$a_{1,0}=3,\ a_{1,2}=-1,\ a_{2,2}=5,\ a_{0,2}=0,\ \deg(p)=4$

רעיון: ננסה לפתח את כל המכפלות לתלמיד לייצוג המקדמים ולבדוק אם קיים מקדם שונה מ-0.

הבעיה: עשוי להיות יקר חישובית. עבור $m$ סוגריים עם $n$ איברים בכל אחד, נידרש ל- $O(n^m)$ פעולות.

נרצה למצוא אלג' הסתברותי שמקבל פולינום (בייצוג כלשהו) ובהסתברות גבוהה אומר לנו האם זה פולינום האפס.

אלגוריתם בסיסי:

1. נבחר קבוצה סופית $A \subseteq \mathbb{F}$.
2. נגריל בהתפלגות אחידה וקטור $x \in \mathbb{F}^n$
(כל קואורדינטה של $x$ מגרילים איבר מ-$A$ בהתפלגות אחידה).
3. נציב $x$ ב-$p$. אם $p(x)=0$ נחזיר שכן, אחרת - נחזיר שלא.

זמן ריצה: נניח שדגימה בהסתברות אחידה מ-$A$ לוקחת $O(\log|A|)$.
בהנחה שבפולינום $O(g)$ ביטויים ושבכל ביטוי יש $O(m)$ סוגריים ובכל סוגריים $O(n)$ איברים, זמן הריצה הוא: $O(g\cdot m\cdot n + n\cdot\log|A|)$.

## תזכורת מההרצאה:

### משפט Schwartz-Zippel

יהי פולינום $P \in F[X_1,..,X_n]$ שאינו זהותית 0.
תהי $A$ קבוצה סופית של איברי $F$.
נגריל הצבה $X_i = a_i$ לכל $i = 1,..,n$.
כאשר ה- $a_i$ים כולם תלויים הדדית וכל אחד מתפלג אחיד ב-$A$.
אזי: $\mathbb{P}[P(a_1,..,a_n) = 0] \leq \frac{\deg(P)}{|A|}$ כאשר $\deg P$ הוא המעלה של $P$.

## משפט Schwartz-Zippel

האלג' הבסיסי טועה בהסתברות: $\mathbb{P}(fail) \leq \frac{\deg P}{|A|}$

אם נבחר $|A| > \deg P$ נקבל שהאלג' צודק בהסתברות גדולה מ-0.
נוכל כעת להשתמש בניפוח כדי להשיג הסתברות גדולה כרצוננו.

## ניפוח באמצעות גודל הקבוצה A

נשים לב שאפשר לבחור קבוצה $A$ עם $|A| > \deg P \cdot e^k$ ונקבל $\mathbb{P}(fail) \leq \frac{1}{e^k}$.

במקרה כזה- זמן הריצה יהיה:

$$O(g \cdot m \cdot n + n\log(\deg(P) \cdot e^k)) = O(gmn + n\log(\deg(p)) + nk)$$

## ניפוח באמצעות חזרות של אלג' בסיסי

**אלגוריתם כללי:**

1. נחזור על האלג' הבסיסי $\frac{K}{\ln\left(\frac{|A|}{degp}\right)}$ פעמים.
2. אם באחת הפעמים קיבלנו שהפולינום אינו 0, נחזיר שלא - $p$ אינו זהותית 0. אחרת, נחזיר ש-$p$ הוא זהותית 0.

**הערה:** אם מתישהו קיבלנו תוצאה שהיא לא אפס אז הפולינום בהכרח לא פולינום האפס. אך אם קיבלנו תוצאות שהן אפסים יכול להיות שמדובר בשורשים של הפולינום.

**טענה:** האלגוריתם נכשל בהסתברות קטנה שווה $\frac{1}{e^k}$.

**הוכחה:**

$$\mathbb{P}\left(\substack{\text{אלג' כללי} \\ \text{נכשל}}\right) = \mathbb{P}\left(\substack{\text{אלג' בסיסי} \\ \text{נכשל}}\right)^{\frac{K}{\ln\left(\frac{|A|}{degp}\right)}} = \mathbb{P}\left(\substack{\text{אלג' בסיסי} \\ \text{נכשל}}\right)^{K\cdot\frac{-1}{\ln\left(\frac{degp}{|A|}\right)}} =$$

$$\mathbb{P}\left(\substack{\text{אלג' בסיסי} \\ \text{נכשל}}\right)^{\frac{\ln\left(\frac{1}{e^k}\right)}{\ln\left(\frac{degp}{|A|}\right)}} = \mathbb{P}\left(\substack{\text{אלג' בסיסי} \\ \text{נכשל}}\right)^{\log_{\frac{degp}{|A|}}\left(\frac{1}{e^k}\right)} \leq \left(\frac{deg(p)}{|A|}\right)^{\log_{\frac{degp}{|A|}}\left(\frac{1}{e^k}\right)} = \frac{1}{e^k}$$

## כפל מטריצות

**קלט:** מטריצות $A,B,C\in\mathbb{R}^{n\times n}$

**פלט:** 1 אם $AB=C$ ו-0 אם $AB\neq C$.

**רעיון:** נחשב את $AB$ ונשווה איבר איבר ל-$C$.

**זמן ריצה:** כפל מטריצות עולה $O(n^3)$, השוואת האיברים $O(n^2)$ סה"כ $O(n^3)$.

**פתרון הסתברותי:**

1. נגדיר וקטור $x \in \mathbb{R}^n$.
2. נחשב את $A(B \cdot x), C \cdot x$
3. נגדיר קבוצה סופית $S \in \mathbb{R}$ ונדגום לרכיב מ-$S$ באופן אחיד לוקטור $x$.
4. נדגום ב-$A(Bx)$ ו-$Cx$.

אם קיבלנו שכל הרכיבים שווים נחזיר 1, אחרת - נחזיר 0.

**טענה:** לכל $k \in \mathbb{N}$ ניתן לבחור $|S| \geq e^k$ כך שיתקיים: $\mathbb{P}(fail) \leq \frac{1}{e^k}$

**הוכחה:** נשים לב כי: $ABx = Cx$ $\forall x \in \mathbb{R}^n$ $\iff$ $AB = C$

$$\iff \forall x \in \mathbb{R}^n \ (AB-C)x = 0$$

$$\iff (AB-C)x \text{ הוא פולינום האפס}$$

נשים לב, $(AB-C)x$ הוא פולינום מדרגה 1 שהוא בהכרח 0 אמ"מ $AB=C$.

נסמן $(AB-C)x = P(x_1,..,x_n)$ ונשתמש בשוורץ-זיפל:

$$\mathbb{P}(fail) = \mathbb{P}((AB-C)x = 0)) \leq \frac{\deg P}{|S|} = \frac{1}{e^k}$$

**זמן ריצה:**

1. חישוב $Bx$ לוקח $O(n^2)$
2. חישוב $A(Bx)$ לוקח $O(n^2)$
3. חישוב $Cx$ לוקח $O(n^2)$
4. לדגום מ-$S$ לוקח $O(nk)$
5. בדיקה האם הוקטורים $ABx, Cx$ שווים לוקחת $O(n)$

ולכן בסה"כ נקבל: $O(n^2+nk)$

# המטריצה של אדמונדס

## דטרמיננטות

תזכורת: תהי $A$ מטריצה מסדר $n\times n$. נסמן ב- $S_n$ את קבוצת הפרמוטציות של $\{1,\dots,n\}$. הדטרמיננטה של $A$ מוגדרת להיות:

$$\boxed{|A| = \det(A) = \sum_{\sigma\in S_n} \operatorname{sign}(\sigma)\cdot A_{1,\sigma(1)},\dots,A_{n,\sigma(n)}}$$

מדובר בפולינום בעל $n^2$ משתנים.

הערה: ניתן לחשב את הדטרמיננטה בסיבוכיות דורג ב- $O(n^3)$. הפולינום שיתקבל לא יהיה בהצגה סטנדרטית אלא ייצוג כמכפלה של $n$ פולינומים. פיתוח לצורה הסטנדרטית ידרוש מספר אקספוננציאלי של מונומים.

שאלה: יהיה $G=(L,R,E)$ גרף דו״צ עבורו $n=|L|=|R|$.
המטריצה של אדמונדס (edmonds) מוגדרת באופן הבא:

$$M_{i,j} = \begin{cases} X_{ij} & (i,j)\in E \\ 0 & \text{else} \end{cases}$$

שאלה: מטריצה $M$ עבור $G$ מקיימת כי $|M|=0$ אמ״מ אין ב- $G$ שידוך מושלם.

## אלגוריתם הסתברותי לשידוך מושלם

1. בהינתן $G=(L,R,E)$ נחשב את מטריצת אדמונדס של $G$.
2. נבחר הקבוצה $S\subseteq\mathbb{R}$ ונגריל ממנה ערכים ל-$M$ בהתפלגות אחידה.
3. נחשב את $|M|$. אם יצא $0$, נחזיר שאין זיווג מושלם.
4. אם לא החזרנו משהו עד הסוף, נחזיר שישנו זיווג מושלם.

**טענה:** לכל $K\in\mathbb{N}$ ניתן לבחור $|S|=n\cdot e^k$ כך שיתקיים: $\mathbb{P}(fail)\le\frac{1}{e^k}$

**הוכחה:** מהטענה הקודמת $|M|=0$ אמ"מ אין ב-$G$ שידוך מושלם.

נסמן $|M|=p(x_1,..,x_n)$. ממשפט שוורץ-זיפל:

$$\mathbb{P}(fail)=\mathbb{P}(p(x_1,..,x_n)=0)\le\frac{\deg p}{n\cdot e^k}\overset{(*)}{\le}\frac{n}{ne^k}=\frac{1}{e^k}$$

(*) - בגלל ההגדרה של הדטרמיננטה שבה כופלים $A_{1,\sigma(1)}\cdot\ldots\cdot A_{n,\sigma(n)}$ ובה $n$ משתנים.

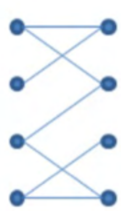
**זמן ריצה:** חישוב $M$ לוקח $O(n^2)$ פעולות.

להציב במטריצה לוקח $O((\log(n)+K)n^2)=O(n^2\log(e^k\cdot n))$ פעולות,

ולחשב את הדטרמיננטה לוקח $O(n^3)$ פעולות. כלומר, סה"כ $O(n^3+n^2K)$.

נשים לב שלרוב $n>>K$ ולכן סה"כ כנראה $O(n^3)$

**דוגמת הרצה:**



$$\longrightarrow \overset{M}{\begin{bmatrix} x_{11} & x_{12} & 0 & 0 \\ x_{21} & 0 & 0 & 0 \\ 0 & x_{32} & 0 & x_{34} \\ 0 & 0 & x_{43} & x_{44} \end{bmatrix}} \longrightarrow \begin{bmatrix} 1 & 2 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 3 \\ 0 & 0 & 1 & 2 \end{bmatrix} \longrightarrow \det(M)=24$$