# נוסחאות עם רקורסיה

לאורך הסמסטר ניתקל בנוסחאות כמו:

$$T\left(n\right)=\begin{cases}2T\left(\frac{n}{2}\right)+n & n>1\\1 & n=1\end{cases}$$

נרצה למצוא פונק' מפורשת $f\left(n\right)$ כך שמתקיים $T\left(n\right)=\Theta\left(f\left(n\right)\right)$.

# דרכים למצוא את $f\left(n\right)$

## דרך א' - ניחוש

מנחשים $f\left(n\right)$ ומוכיחים באינדוקציה.

### דוגמה

$$T\left(n\right)=\begin{cases}T\left(n-1\right)+n^{2} & n>1\\1 & n=1\end{cases}$$

ננחש: $T\left(n\right)=n^{3}$ וננסה למצוא $c_{1},\,c_{2}$ כך שמתקיים:

$$c_{1}n^{3}\leq T\left(n\right)\leq c_{2}n^{3}$$

אם $c_{1}<1$ ו$c_{2}>1$ אז:

$$c_{1}\cdot1\leq T\left(1\right)\leq c_{2}\cdot1$$

ננסה לבחור $c_{2}$ כך ש$T\left(n\right)\leq c_{2}n^{3}$.
נניח שזה נכון לכל $n^{'}<n$.

$$\begin{aligned}T\left(n\right) & = T\left(n-1\right)+n^{2}\leq c_{2}\left(n-1\right)^{3}+n^{2}\\ & = c_{2}n^{3}-3c_{2}n^{3}+3c_{2}n-c_{2}+n^{2}\\ & = c_{2}n^{3}-\left[\left(3c_{2}-1\right)n^{2}-3c_{2}n+c_{2}\right]\end{aligned}$$

אנו רוצים שלכל $n$ יתקיים:

$$\left(3c_{2}-1\right)n^{2}-3c_{2}n+c_{2}\geq0$$

נבחר $c_{2}=3$ ואז אי השוויון הוא

$$8n^{2}-9n+3\geq0$$

אין שורשים כי

$$9^{2}-4\cdot8\cdot3=-15<0$$

ולכן אין שורשים לביטוי ולכן אי השוויון נכון תמיד.
לכן הוכחנו באינדוקציה שמתקיים עבור $c_{2}=3$:

$$T\left(n\right)\leq c_{2}n^{3}$$

## דרך ב׳ - פיתוח הנוסחה

פותחים את הרקורסיה ומוצאים נוסחה מפורשת.

### דוגמה

$$T(n)=\begin{cases}\sqrt{n}T\left(\sqrt{n}\right)+n & n\geq 2\\ 1 & n\leq 2\end{cases}$$

אזי:

$$\begin{aligned}
T(n) &= \sqrt{n}T\left(\sqrt{n}\right)+n\\
&= n^{\frac{1}{2}}T\left(n^{\frac{1}{2}}\right)+n\\
&= n^{\frac{1}{2}}\left(n^{\frac{1}{4}}T\left(n^{\frac{1}{4}}\right)+n^{\frac{1}{2}}\right)+n\\
&= n^{1-\frac{1}{4}}T\left(n^{\frac{1}{4}}\right)+n+n\\
&= n^{1-\frac{1}{4}}\left(n^{\frac{1}{8}}T\left(n^{\frac{1}{8}}\right)+n^{\frac{1}{4}}\right)+2n\\
&= n^{1-\frac{1}{8}}T\left(n^{\frac{1}{8}}\right)+3n\\
&= n^{1-\frac{1}{2^k}}T\left(n^{\frac{1}{2^k}}\right)+kn
\end{aligned}$$

כעת נבדוק מתי $n^{\frac{1}{2^k}}\leq 2$:

$$\begin{aligned}
\lg n^{\frac{1}{2^k}} &\leq \lg 2\\
\frac{1}{2^k}\lg n &\leq 1\\
\lg n &\leq 2^k\\
\lg\lg n &\leq k
\end{aligned}$$

(הערה: כאשר נכתוב $\lg$ אנו מתכוונים ל-$\log_2$, וכאשר נכתוב $\log$ אנו מתכוונים ל-$\ln$).
ניקח $k=\lg\lg n$ ונקבל מהפיתוח הקודם:

$$\begin{aligned}
T(n) &= \frac{n}{n^{\frac{1}{2^k}}}T\left(n^{\frac{1}{2^k}}\right)+n\lg\lg n\\
&= \frac{n}{2}T(2)+n\lg\lg n\\
&= \frac{n}{2}+n\lg\lg n=\Theta(n\lg\lg n)
\end{aligned}$$

## משפט שיטת האב

שיטת האב מבוססת על המשפט הבא:

**משפט 4.1 (משפט שיטת האב)**

יהיו $a\geq 1$ ו-$b>1$ קבועים, תהי $f(n)$ פונקציה, ותהי $T(n)$ פונקציה המוגדרת על השלמים האי־שליליים על־ידי נוסחת הנסיגה:

$$T(n)=aT(n/b)+f(n)$$

כאשר אנו מבינים את $n/b$ כ- $\lfloor n/b \rfloor$ או $\lceil n/b \rceil$.

בתנאים אלה ניתן לחסום אסימפטוטית את $T(n)$ באופן הבא:

1. אם קיים קבוע $\epsilon > 0$ כך ש- $f(n) = O(n^{\log_b a - \epsilon})$, אזי $T(n) = \Theta(n^{\log_b a})$.
2. אם $f(n) = \Theta(n^{\log_b a})$, אזי $T(n) = \Theta(n^{\log_b a} \lg n)$.
3. אם קיים קבוע $\epsilon > 0$ כך ש-$f(n) = \Omega(n^{\log_b a + \epsilon})$, ואם קיים קבוע $c < 1$ כך ש-$af(n/b) \le cf(n)$ עבור כל ה-$n$-ים הגדולים דיים, אזי $T(n) = \Theta(f(n))$. ■

# דוגמאות

1.

$$T(n) = 7T\left(\frac{n}{2}\right) + n^2$$

במקרה הזה

$$b = 7,\, a = 2$$

אזי

$$\log_a b = \log_2 7 > \log_2 4 = 2$$

ומתקיים

$$n^2 = O\left(n^{\log_a b - \epsilon}\right)$$

לכן

$$T(n) = \Theta\left(n^{\log_2 7}\right) = \Theta\left(n^{2.81}\right)$$

2.

$$T(n) = 4T\left(\frac{n}{2}\right) + \Theta\left(n^2\right)$$

במקרה זה

$$a = 2,\, b = 4$$

אזי

$$\log_a b = \log_2 4 = 2$$

ומתקיים

$$\Theta\left(n^2\right) = \Theta\left(n^{\log_a b}\right)$$

לכן

$$T(n) = \Theta\left(n^{\log_a b} \log n\right) = \Theta\left(n^2 \log n\right)$$

הערה: בתוך $O, \Theta, \Omega$ בדר"כ אין משמעות לבסיס הלוגריתם, כי $\log_r n = \frac{\ln n}{\ln r} = \Theta(\ln n)$.

3.

$$T(n) = 4T\left(\frac{n}{5}\right) + \Theta(n)$$

במקרה זה

$$\log_5 4 < \log_5 5 = 1$$

לכן קיים $\epsilon > 0$ כך ש:

$$n = \Omega\left(n^{\log_5 4 + \epsilon}\right)$$

ומתקיים המקרה השלישי, לכן

$$T(n) = \Theta(n)$$

.4

$$T(n) = 4T\left(\frac{n}{5}\right) + \Theta(n\lg n)$$

שוב,

$$\log_5 4 < \log_5 5 = 1$$

לכן קיים $\epsilon > 0$ כך ש

$$n\lg n = \Omega\left(n^{\log_5 4+\epsilon}\right)$$

ולכן

$$T(n) = \Theta(n\lg n)$$

.5

$$T(n) = 5T\left(\frac{n}{5}\right) + n\log_5 n$$

במקרה זה

$$\log_5 5 = 1$$

אבל

$$f(n) \neq \Theta(n)$$

ולא קיים $\epsilon > 0$ כך ש

$$f(n) \quad = \quad \Omega\left(n^{1+\epsilon}\right)$$

או

$$f(n) = O\left(n^{1-\epsilon}\right)$$

לכן אי אפשר להשתמש במשפט המאסטר.
בבית - פתרו עם פיתוח וגלו שמתקיים:

$$T(n) = \Theta\left(n\log^2 n\right)$$

## מחסנית

- Top - מחזיר את האיבר בראש המחסנית ($O(1)$)
- Pop - מוציאים את האיבר הראשון במחסנית ומחזירים אותו.
- $\text{Push}(x)$ - דוחפים את $x$ לראש המחסנית.
- IsEmpty - האם המחסנית ריקה.

## תור

אותן הפעולות כמו מחסנית עם ההבדל שpush מכניסה איבר לסוף התור.

### דרך קלה לזכור

תור - FIFO : First In First Out.
מחסנית - LIFO : Last In First Out.

## תרגיל 1

יש שתי מחסניות, אחת מלאה ואחת ריקה.
כתבו אלגוריתם המעביר את תוכל המחסנית הראשונה למחסנית השניה ומרוקן את הראשונה.

### פתרון

קלט: $S_1$ מלאה, $S_2$ ריקה.

---

אלגוריתם 3 פתרון תרגיל 1

---

ניצור מחסנית ריקה $T$.

```
while S_1.notempty():
 T.Push(S_1.Pop())
while T.notempty():
 S_2.Push(T.Pop())
```

---

זמן ריצה: $\Theta(n)$ כאשר $n$ הוא מס' האיברים ב$S_1$.

## תרגיל 2

יש $n$ אנשים ששמותיהם $1, ..., n$ שעומדים במעגל. מתחילים מאיש מס' 1.
מדלגים $k$ אנשים ומוציאים את האיש ה-$k$-י מהמעגל.
חוזרים על התהליך (כשמתחילים היכן שעצרנו) עד אשר אין אנשים במעגל.
כתבו אלגוריתם המדפיס את האנשים המוצאים לפי הסדר של ההוצאה ("תמורת יוספוס").

### פתרון

נשתמש בתור.

---

אלגוריתם 4 פתרון תרגיל 2

---

```
Q = empty queue
for i=1 to n:
 Q.push(i)
while Q.not_empty():
 for j=1 to k-1:
  Q.Push(Q.Pop())
 print(Q.Pop())
```

---

סיבוכיות זמן: $\Theta(nk)$.