מבני נתונים 1
234218

תרגיל יבש 1

מספר

הוגש ע"י:

גיא אוחיון	315823856
שם	מספר זהות
רון קנטורוביץ'	207733015
שם	מספר זהות

ציון:

לפני בונוס הדפסה:

כולל בונוס הדפסה:

נא להחזיר לתא מס':

שאלה 1

חלק א

- צ"ל האם לכל k טבעי קיימים קבועים $c, n_0 > 0$ כל שלכל $n \geq n_0$: $f(n) \leq c(f(n)+k)$. k טבעי ולכן $k \geq 0$. לכן לדוגמא אם נבחר $c = 1$ נקבל $f(n) \leq f(n) + k$ לכל n טבעי. לכן מקרה זה מתקיים.

- צ"ל האם לכל k טבעי קיימים קבועים $c, n_0 > 0$ כל שלכל $n \geq n_0$: $f(n) + k \leq cf(n)$. מקרה זה אינו נכון. לדוגמא עבור $f(n) = \frac{1}{n}$ נבחר $k = 1$ ואז לכל $c, n_0 > 0$ נבחר $n > \max(n_0, c-1)$ ואז יתקיים:
$$f(n) + k = \frac{1}{n} + k = \frac{1}{n} + 1 = \frac{1+n}{n} >* cf(n) = \frac{c}{n}$$
*כי הרי $n > c - 1$, כלומר $\mathrm{n} + 1 > \mathrm{c}$.

- צ"ל האם לכל k טבעי קיימים קבועים $c, n_0 > 0$ כל שלכל $n \geq n_0$: $f(n+k) \leq cf(n)$. מקרה זה אינו מתקיים. לדוגמא עבור $f(n) = e^{e^n}$, ניקח $k = 1$ ואז לכל $c > 0$ ולכל $n_0 > 0$ נבחר $n > \max(n_0, \ln(\ln(\sqrt[e-1]{c})))$ כך שיתקיים:
$$f(n+k) = e^{e^{n+k}} = e^{e^n e} = \left(e^{e^n}\right)^e > cf(n) = ce^{e^n}$$
כי הרי:
$$n > \ln(\ln(\sqrt[e-1]{c})) \rightarrow \left(e^{e^n}\right)^{e-1} > c$$

חלק ב

תחילה נבין שההגדרה של $f(n) = P(g(n))$ היא מקרה פרטי של $f(n) = O(g(n))$ בו $0 < c < 1$. לכן אם $f(n) \neq O(g(n))$ אז בפרט $f(n) \neq P(g(n))$.

- לא נכון. לדוגמא $f(n) = n$ ו-$g(n) = 1$. ברור ש-$f(n) = \omega(g(n))$ כי $\lim_{n \to infinity} \frac{g(n)}{f(n)} = 0$ אבל גם ברור ש-$f(n) \neq O(g(n))$ כי לכל $c > 0$ ולכל $n_0 > 0$ ניקח $n > \max(n_0, c)$ ואז:
$$f(n) = n > c = g(n)$$

  לכן גם בפרט $f(n) \neq P(g(n))$ כי זה מתקיים לכל $c > 0$.
- נתון $f_1(n) = P(g_1(n))$ וגם $f_2(n) = P(g_2(n))$, לכן קיימים $n_1, n_2 > 0$ ו-$0 < c_1, c_2 < 1$ כך שלכל $n > n_0 = \max(n_1, n_2)$ מתקיים:
$$f_1(n) \leq c_1 g_1(n), f_2(n) \leq c_2 g_2(n)$$
נבחר $c = \max(c_1, c_2)$ ואז נקבל:
$$f_1(n) + f_2(n) \leq c_1 g_1(n) + c_2 g_2(n) \leq c\left(g_1(n) + g_2(n)\right)$$
לכן הוכחנו ש- $f_1(n) + f_2(n) = P(g_1(n) + g_2(n))$ כי מצאנו $0 < c < 1$ ו-$n_0 > 0$ שעבורם זה נכון.

- אגף שמאל זה סכום סדרה הנדסית עם $q = 2$. מהנוסחה לסכום סדרה הנדסית מתקיים:
$$f(n) = \sum_{i=1}^{n-1} 2^i = \frac{2 - 2^n}{1 - 2} = 2^n - 2$$
לכל $n_0 > 0$ ולכל $0 < c < 1$ נבחר $n > \max(n_0, \log_2 \frac{2}{1-c})$ ואז יתקיים:
$$f(n) = 2^n - 2 > c2^n$$
כי הרי:

$$n > \log_2 \frac{2}{1-c} \rightarrow 2^n > 2^{\log_2 \frac{2}{1-c}} = \frac{2}{1-c} \rightarrow 2^n - c2^n > 2 \rightarrow 2^n - 2 > c2^n$$

לכן מקרה זה אינו מתקיים.

<u>חלק ג</u>

- נתון $f(n) = L_{g(n)}(\log(g(n)))$. לכן קיימים $n_0, c > 0$ כך שלכל $n > n_0$ מתקיים:

$$|\log(\frac{f(n)}{g(n)})| \leq c * \log(g(n))$$

לדוגמא עבור $g(n) = n$ ו-$f(n) = n^2$ נבחר $c = 2, n_0 > 1$ ואז לכל $n \geq n_0$ יתקיים:

$$|\log(\frac{f(n)}{g(n)})| = |\log(\frac{n^2}{n})| = \log(n) \leq c * \log(g(n)) = 2\log(g(n)) = 2\log(n)$$

כלומר הנתון מתקיים, אבל $f(n) = n^2 \neq O(g(n) = O(n)$ (ראינו זאת בתרגול מספר 1). לכן הטענה אינה נכונה.

- נכון. נתון $f(n) = \theta(g(n))$ לכן קיימים $n_0, c_0 > 0$ כך שלכל $n > n_0$ מתקיים:

$$f(n) \leq c_0 g(n)$$

וגם קיימים $n_1, c_1 > 0$ כל שלכל $n > n_1$ מתקיים:

$$f(n) \geq c_1 g(n)$$

ומכאן לכל $n > \max(n_0, n_1)$ יתקיים:

$$c_1 g(n) \leq f(n) \leq c_0 g(n)$$

$$c_1 \leq \frac{f(n)}{g(n)} \leq c_0$$

נבחר $n_2 = \max(n_0, n_1)$.

הפונקציה הלוגוריתמית מונוטונית עולה, לכן נוכל לכתוב שלכל $n > n_2$:

$$\log(c_1) \leq \log\left(\frac{f(n)}{g(n)}\right) \leq \log(c_0)$$

נבחר $c_2 = \max(|\log(c_0)|, |\log(c_1)|)$ ואז:

$$\left|\log\left(\frac{f(n)}{g(n)}\right)\right| \leq c_2 = c_2 * 1$$

לכן הוכחנו ש:

$$\left|\log\left(\frac{f(n)}{g(n)}\right)\right| = O(1)$$

כלומר

$$f(n) = L_{g(n)}(1)$$

- נכון. נתון $f(n) = L_{g(n)}(h(n))$ ולכן קיימים $c_0, n_0 > 0$ כך שלכל $n > n_0$ מתקיים:

$$\left|\log\left(\frac{f(n)}{g(n)}\right)\right| \leq c_0 h(n)$$

בנוסף נתון $g(n) = L_{l(n)}(h(n))$ ולכן קיימים $c_1, n_1 > 0$ כך שלכל $\mathrm{n} > \mathrm{n}_1$ מתקיים:

$$\left|\log\left(\frac{g(n)}{l(n)}\right)\right| \leq c_1 h(n)$$

נבחר $n_2 = \max(n_1, n_0)$ ו-$c_2 = c_0 + c_1$ ואז לכל $n > n_2$ יתקיים:
נרשום:

$$\left|\log\left(\frac{f(n)}{l(n)}\right)\right| = \left|\log\left(\frac{f(n)}{g(n)} * \frac{g(n)}{l(n)}\right)\right| = \left|\log\left(\frac{f(n)}{g(n)}\right) + \log\left(\frac{g(n)}{l(n)}\right)\right|$$
$$\leq \left|\log\left(\frac{f(n)}{g(n)}\right)\right| + \left|\log\left(\frac{g(n)}{l(n)}\right)\right| \leq c_0 h(n) + c_1 h(n) = (c_0 + c_1)h(n) = c_2 h(n)$$

ולכן הוכחנו ש-$f(n) = L_{l(n)}(h(n))$ וזה מה שצריך להוכיח.

שאלה 2

הסדר הוא:

$$f_4 \preceq f_{13} \preceq f_2 \preceq f_7 \preceq f_{10} \preceq f_8 \preceq f_9 \preceq f_3 \preceq f_1 \preceq f_{15} \preceq f_6 \preceq f_{14} \preceq f_5 \preceq f_{12} \preceq f_{11}$$

נוכיח זאת בשלבים, משמאל לימין:

<u>$f_4 \preceq f_{13}$:</u>
מהגדרת ערך שלם נקבל $\lfloor \log(n) \rfloor \geq \log(n) - 1$ ולכן:
עבור $c = 1$ וגם $n_0 = 1$ נקבל שלכל $n > n_0$ מתקיים:

$$\begin{aligned}(\log(n) - \lfloor \log(n) \rfloor)^n \leq (\log(n) - \log(n) + 1)^n = 1^n \leq 1 \leq \\ \leq 1 + \sum_{i=2}^{n} \frac{1}{i^3} = \sum_{i=1}^{n} \frac{1}{i^3} = c \cdot f_{13}(n)\end{aligned}$$

ולכן לפי ההגדרה $f_4 = O(f_{13})$.

<u>$f_{13} \preceq f_2$:</u>
נבחר $c = 1$ וגם $n_0 = 2^{\frac{3}{2}}$ ואז לכל $n > n_0$מתקיים:

$$\begin{aligned}f_{13}(n) = \sum_{i=1}^{n} \frac{1}{i^3} \leq 1 + \int_1^n \frac{1}{i^3} di \leq 1 + \int_1^\infty \frac{1}{i^3} di = 1 + \frac{1}{2} = \frac{3}{2} = \\ = \log(2^{\frac{3}{2}}) = \log(n_0) < \log(n) \leq \log(\frac{1}{2} n^{50}) = \log(n^{50} - \frac{1}{2} n^{50}) \leq \\ \leq \log(n^{50} - n^{40}) = f_2(n)\end{aligned}$$

ומכאן לפי ההגדרה מתקיים $f_{13} = O(f_2)$.

<u>$f_2 \preceq f_7$:</u>
עבור $c = 1$ וגם $n_0 = 200$ נקבל:

$$\begin{aligned}c \cdot \sum_{i=1}^{n} \log(i) \geq \sum_{i=\frac{n}{2}}^{n} \log(i) \geq \frac{n}{2} \log(\frac{n}{2}) = \frac{n}{2}(\log(n) - \log(2)) = \\ = \frac{n}{2} \log(n) - \frac{n}{2} \cdot 1 = \frac{n}{4} \log 9n) + \frac{n}{4} \log(n) - \frac{n}{4} \cdot 2 = \\ = \frac{n}{4} \log(n) + \frac{n}{4}(\log(n) - 2) > \frac{200}{4} \log(n) + \frac{200}{4}(\log(200) - 2)) \geq \\ \geq 50 \log(n) = \log(n^{50}) = \log(n^{50}) \geq \log(n^{50} - n^{40}) \\ = \log(n^{50} - n^{40})\end{aligned}$$

ולכן לפי ההגדרה $f_2 = O(f_7)$

<u>$f_7 \preceq f_{10}$:</u>
עבור $c = 2$ וגם $n_0 = 1$ נקבל שלכל $n > n_0$ מתקיים $\log(n) < n$ ולכן:

$$\sum_{i=1}^{n} \log(i) \leq n \cdot \log(n) \leq n^2 \leq n^2 \log(n) = n \cdot \log(n^n) \leq$$
$$\leq n \log(n^n \cdot n!) \leq 2 \cdot n \cdot \frac{1}{2} \log(n^n \cdot n!) = 2 \cdot n \cdot \log(\sqrt{n^n \cdot n!}) =$$
$$= c \cdot f_{10}$$

ומההגדרה מקבלים $f_7 = O(f_{10})$.

<u>$f_{10} \preceq f_8$:</u>
נחשב את f_8 באמצעות סכום סדרה חשבונית וסדרת ריבועים:
ועבור $c = \frac{1}{3}$ וגם $n_0 = 16$ נקבל שלכל $n > n_0$ $\frac{n}{4} \geq \log(n)$ ולכן

$$c \cdot \sum_{i=1}^{n} \sum_{j=1}^{n-i+1} (j-1) = 3 \sum_{i=1}^{n} \frac{(0 + n - i) \cdot (n - i + 1)}{2} = 3 \sum_{i=1}^{n} \frac{(i-1)(i)}{2} =$$
$$= 3 \sum_{i=1}^{n} \frac{1}{2} \cdot i^2 - 3 \sum_{i=1}^{n} \frac{1}{2} i = 3 \cdot \frac{n(n+1)(2n+1)}{12} - 3 \cdot \frac{n(n+1)}{4} =$$
$$= 3 \cdot \frac{(n-1) \cdot n \cdot (n+1)}{6} \geq \frac{n^2}{2} \cdot (n-1) \geq \frac{n^2}{2} (\log(n) + \log(n)) =$$
$$= \frac{n}{2} (n \log(n) + n \log(n)) \geq \frac{n}{2} (\log(n^n) + \sum_{i=1}^{n} \log(i)) =$$
$$= n \cdot \frac{1}{2} (\log(n^n) + \log(n!)) = n \cdot \frac{1}{2} \cdot \log(n^n \cdot n!) =$$
$$= n \cdot \log(\sqrt{n^n \cdot n!}) = f_{10}(n)$$

כלומר לכל $n > n_0$ מקבלים $f_{10} \leq c \cdot f_8$ ולכן לפי ההגדרה:
$f_{10} = O(f_8)$.

<u>$f_8 \preceq f_9$:</u>
נמשיך מהחישוב של f_8 בסעיף הקודם:
עבור $c = \frac{1}{6}$ וגם $n_0 = 2^{2^{2^4}}$ נקבל שלכל $n > n_0$:

(בעזרת חוקי לוגריתמים)

$$f_8(n) = \frac{(n-1)\cdot n\cdot(n+1)}{6} \overset{n\geq 2}{\leq} \frac{n\cdot n\cdot n^2}{6} = \frac{n^4}{6} \leq \frac{1}{6}\cdot n^{\log\log\log(2^{2^{2^4}})} <$$
$$< \frac{1}{6}\cdot n^{\log\log\log(n)} = \frac{1}{6}\cdot n^{\frac{\log(n)\cdot\log\log\log(n)}{\log(n)}} = \frac{1}{6}\cdot n^{\frac{\log((\log\log(n))^{\log(n)})}{\log(n)}} =$$
$$= \frac{1}{6}\cdot n^{\log_n((\log\log(n))^{\log(n)})} = \frac{1}{6}(\log\log(n))^{\log(n)} = \frac{1}{6}\cdot f_9(n)$$

ולכן לפי ההגדרה $f_8(n) = O(f_9(n))$

<u>$f_9 \preceq f_3$:</u>

$$(\log\log(n))^{\log n} = 5^{\log_5((\log\log(n))^{\log n})} = 5^{\log(n)\cdot\log_5(\log\log(n))} \leq 5^{\log(n)\cdot\log(n)} = 5^{\log^2(n)}$$

אם נראה $\log^2(n) \leq \sqrt{n}$ החל מ־n_0 מסוים נוכל לסיים את ההוכחה:
ניעזר בהגדרת הגבול מחדו"א ובכלל לופיטל:

$$\lim_{x\to\infty}\frac{\log^2 x}{\sqrt{x}} = \lim_{x\to\infty}\frac{2\cdot\log(x)\cdot\frac{1}{x}}{\frac{1}{2\sqrt{x}}} = \lim_{x\to\infty}\frac{4\cdot\log x}{\sqrt{x}} = \lim_{x\to\infty}\frac{4\cdot\frac{1}{x}}{\frac{1}{2\sqrt{x}}} = \lim_{x\to\infty} = \frac{8}{\sqrt{x}} = 0$$

לפי הגדרת הגבול לכל $\epsilon > 0$ ובפרט עבור $\epsilon = 1$ קיים $n_0 > 0$ כך שלכל $n > n_0$ מתקיים:

$$\frac{\log^2 x}{\sqrt{x}} < 1 \Rightarrow \log^2 x < \sqrt{x}$$

ולכן אם נבחר $c = 1$, נקבל שלכל $n > n_0$ (שקיבלנו מהגדרת הגבול) מתקיים ממונוטוניות הפונקציה האקספוננציאלית:

$$(\log\log(n))^{\log n} \leq 5^{\log^2(n)} < 5^{\sqrt{n}} = c\cdot f_3(n)$$

לכן לפי ההגדרה $f_9 = O(f_3)$.

<u>$f_3 \preceq f_1$:</u>
מתקיים:

$$\binom{n}{\frac{n}{2}} = \frac{n!}{(\frac{n}{2})!\cdot(\frac{n}{2})!} = \frac{n\cdot(n-1)\cdot\ldots\cdot(\frac{n}{2}+1)}{\frac{n}{2}\cdot(\frac{n}{2}-1)\cdot\ldots\cdot 1} = \prod_{i=0}^{\frac{n}{2}-1}\frac{n-i}{\frac{n-2i}{2}} = 2^{\frac{n}{2}}\cdot\prod\frac{n-i}{n-2i} \geq 2^{\frac{n}{2}}\cdot\prod_{i=0}^{\frac{n}{2}-1} 1 = 2^{\frac{n}{2}}$$

מנגד:

$$\lim_{x\to\infty}\frac{\sqrt{x}\cdot\log 5}{\frac{x}{2}}=\lim_{x\to\infty}\frac{2\log 5}{x}=0$$

לכן עבור $\epsilon=1$ קיים $n_0>0$ (לפי הגדרת הגבול) כך שלכל $n>n_0$:

$$\frac{\sqrt{x}\cdot\log 5}{\frac{x}{2}}<1\Rightarrow\sqrt{x}\cdot\log 5<\frac{x}{2}$$

ולכן אם ניקח $c=1$ אז לכל $n>n_0$ (שקיבלנו מהגדרת הגבול) מתקיים:

$$5^{\sqrt{n}}=2^{\log(5^{\sqrt{n}})}=2^{\sqrt{n}\cdot\log 5}<2^{\frac{n}{2}}\le\binom{n}{\frac{n}{2}}$$

ולכן לפי ההגדרה $f_3=O(f_1)$.

<u>$f_1\preceq f_{15}$:</u>
לפי הבינום של ניוטון:

$$f_{15}(n)=\sum_{i=0}^{n-2}\binom{n-1}{i}+\sum_{i=1}^{n-1}\binom{n-1}{i}=(2^{n-1}-\binom{n-1}{n-1})+(2^{n-1}-\binom{n-1}{0})=2^n-2$$

בסעיף הקודם ראינו שקיים n_0 כך שלכל $n>n_0$ מתקיים

$$5^{\sqrt{n}}<2^{\frac{n}{2}}$$

ולכן אם ניקח $n_1=max\{n_0,2\}$ נקבל שעבור $c=1$, לכל $n>n_1$ מתקיים:

$$5^{\sqrt{n}}<2^{\frac{n}{2}}=2^{\frac{n}{2}}+0\le 2^{\frac{n}{2}}+(2^{\frac{n}{2}}-2)=2^n-2=f_{15}(n)$$

ולכן לפי ההגדרה $f_1=O(f_{15})$.

<u>$f_{15}\preceq f_6$:</u>
נשתמש בנוסחה לסכום סדרה הנדסית (עם מנה 2 ואיבר ראשון 2) על מנת לחשב במפורש:

$$f_6=\sum_{i=1}^{n}2^i=\frac{2\cdot(2^n-1)}{2-1}=2^{n+1}-2$$

עבור $c=\frac{1}{2}$ ועבור $n_0=1$ נקבל שלכל $n>n_0$ מתקיים:

$$f_{15}(n)=2^n-2=\frac{1}{2}\cdot(2^{n+1}-4)<\frac{1}{2}\cdot(2^{n+1}-2)=c\cdot f_6(n)$$

ולכן $f_{15} = O(f_6)$.
אך יתרה מכך, נוכיח את הכיוון ההפוך:
עבור $c = 3$ וגם $n_0 = 3$ נקבל שלכל $n > n_0$ מתקיים

$$f_6(n) = 2^{n+1} - 2 = 2^{n+1} - 4 + 2 = 2 \cdot (2^n - 2) + 2 < 2 \cdot (2^n - 2) + (2^n - 2) =$$
$$= c \cdot f_{15}(n)$$

ולכן $f_6 = O(f_{15})$.
ובסך הכול $\underline{f_{15} = \Theta(f_6)}$.
$\underline{f_6 \preceq f_{14}}$:
נחשב את f_{14} באופן מפורש בעזרת זהות הבינום:

$$(x+1)^n = \sum_{i=0}^{n} \binom{n}{i} \cdot x^i \Rightarrow n \cdot (x+1)^{n-1} = \sum_{i=0}^{n} \binom{n}{i} \cdot i \cdot x^{i-1} \Rightarrow$$
$$\Rightarrow n \cdot 2^{n-1} = \sum_{i=0}^{n} \binom{n}{i} \cdot i = f_{14}(n)$$

ולכן עבור $c = 1$ ועבור $n_0 = 1$ נקבל שלכל $n > n_0$ מתקיים:

$$f_6(n) = 2^{n+1} - 2 < 2^{n+1} = 1 \cdot 2^{n+1} \le n \cdot 2^{n+1} = f_{14}(n)$$

ולכן לפי ההגדרה $f_6 = O(f_{14})$.

$\underline{f_{14} \preceq f_5}$:
עבור $c = 3$ ועבור $n_0 = 1$ נקבל שלכל $n > n_0$ מתקיים:
($n < 2^n \iff \log n < n$)

$$f_{14}(n) = n \cdot 2^{n-1} \le n \cdot 2^n < 2^n \cdot 2^n = 4^n = \frac{16^n}{4^n} = 3 \cdot \frac{16^n}{3 \cdot 4^n} =$$
$$= 3 \cdot \frac{16^n}{4^n + 4^n + 4^n} \le 3 \cdot \frac{16^n}{4^n + 3^n + 2^n} \le 3 \cdot \frac{16^n + 14^n + 13^n}{4^n + 3^n + 2^n} = c \cdot f_5(n)$$

ולכן לפי ההגדרה $f_{14} = O(f_5)$.

$\underline{f_5 \preceq f_{12}}$:
עבור $c = 3$ ועבור $n_0 = 2^{33}$ נקבל שלכל $n > n_0$ מתקיים:
($n > 6 \Rightarrow \log(n) - 1 > 32$)

$$f_5(n) = \frac{16^n + 14^n + 13^n}{4^n + 3^n + 2^n} \le 16^n + 14^n + 13^n \le 3 \cdot 16^n = 3 \cdot 32^{\frac{n}{2}} <$$
$$< 3 \cdot (\log(n) - 1)^{\frac{n}{2}} = 3 \cdot (\log(\frac{n}{2}))^{\frac{n}{2}} \le 3 \cdot \prod_{i=\frac{n}{2}}^{n} \log(i) \le 3 \cdot \prod_{i=2}^{n} \log(i) = 3 \cdot f_{12}(n)$$

ולכן לפי ההגדרה $f_5 = O(f_{12})$.

<u>$f_{12} \preceq f_{11}$:</u>
ראשית נחשב את הגבול עם לופיטל:

$$\lim_{x\to\infty} \frac{n^2}{9^n} = \lim_{x\to\infty} \frac{2\cdot n}{9^n \cdot \ln(9)} = \lim_{x\to\infty} \frac{2}{9^n \cdot \ln^2 9} = 0$$

לכן לפי הגדרת הגבול עבור $\epsilon = 1$ נקבל שקיים $n_0 > 0$ כך שלכל $n > n_0$ מתקיים:

$$\frac{n^2}{9^n} < 1 \Rightarrow n^2 < 9^n$$

ולכן עבור $c = 1$ וגם n_0 שקיבלנו מהגדרת הגבול נקבל שלכל $n > n_0$ מתקיים:

$$f_{12} \leq (\log(n))^n = 2^{\log(\log^n(n))} = 2^{n\cdot\log(\log(n))} \leq 2^{n\cdot\log(n)} \leq 2^{n^2} < 2^{9^n} = f_{11}(n)$$

ולכן לפי ההגדרה $f_{12} = O(f_{11})$.

ובסך הכול הוכחנו את כל השרשרת.

שאלה 3

סעיף א'

ראשית נוכיח שלכל $n \in \mathbb{N}$ מתקיים $n = \lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil$ מתכונות ערך עליון ותחתון:

אם $n = 2k$ זוגי אז הטענה טריוויאלית כי $n = 2k = \lfloor k \rfloor + \lceil k \rceil = \lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil$.

אם $n = 2k+1$ אי זוגי אז:

$$\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil = \lfloor k + \frac{1}{2} \rfloor + \lceil k + \frac{1}{2} \rceil = k + k + 1 = 2k + 1 = n$$

כעת, נוכיח באינדוקציה שלמה טענה חזקה יותר על T שממנה תנבע הטענה המקורית:

נוכיח שקיים $c = 3$ כך שעבור $n_0 = 1$, לכל $n > n_0$ מתקיים $T(n) < c \cdot n - 1$ (וגם $T(n) < c \cdot n$).

נסמן $c = 3, n_0 = 1$.

<u>בסיס $n = 1$:</u>

עבור $T(1) = 1 < 2 = 3 \cdot 1 - 1 = c \cdot n - 1$, הטענה נכונה

<u>צעד:</u>

נניח שהטענה מתקיימת לכל $k < n$.

בפרט הטענה מתקיימת עבור $\lceil \frac{n}{2} \rceil, \lfloor \frac{n}{2} \rfloor$.

כלומר $\lfloor \frac{n}{2} \rfloor \le c \cdot \lfloor \frac{n}{2} \rfloor - 1$ וגם $\lceil \frac{n}{2} \rceil \le c \cdot \lceil \frac{n}{2} \rceil - 1$, אז נקבל

$$T(n) = T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + 1 < c \cdot \lfloor \frac{n}{2} \rfloor - 1 + c \cdot \lceil \frac{n}{2} \rceil - 1 + 1 =$$
$$= c \cdot \lfloor \frac{n}{2} \rfloor + c \cdot \lceil \frac{n}{2} \rceil - 1 = c \cdot (\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil) - 1 = c \cdot n - 1 < c \cdot n$$

כלומר הראינו גם $T(n) < c \cdot n - 1$ וגם $T(n) < c \cdot n$.

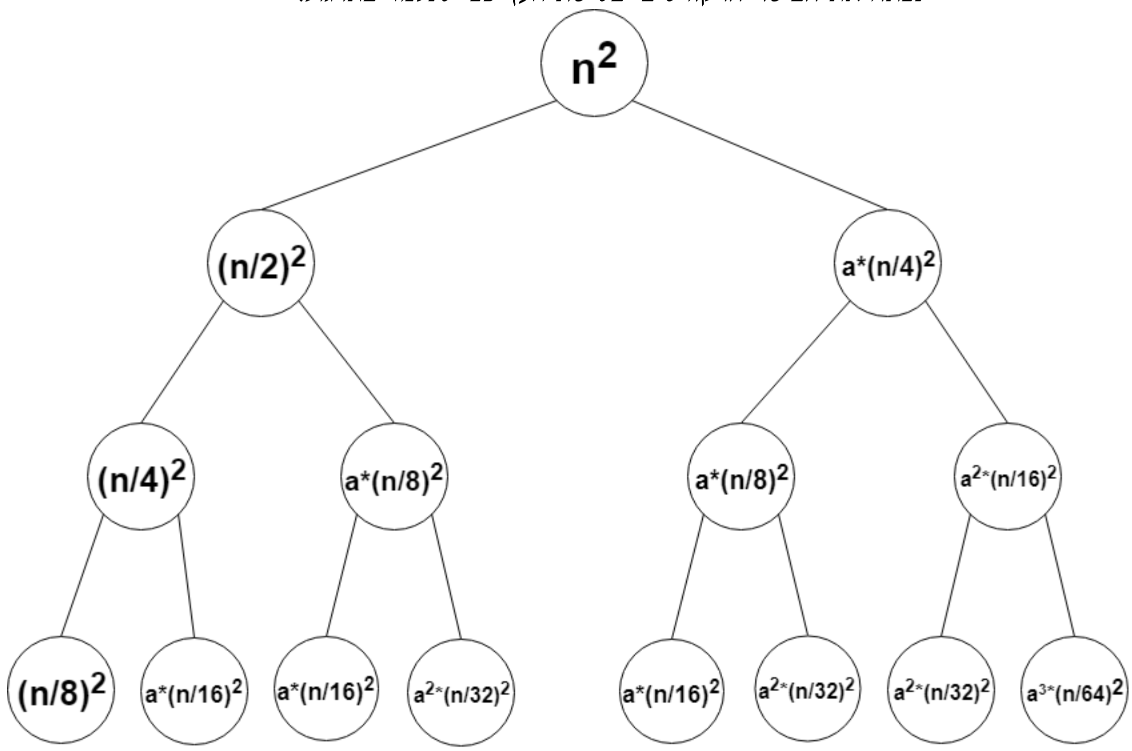
לפי עיקרון האינדוקציה נובע שהטענה נכונה.

כלומר עבור $n_0 = 1, c = 3$ מתקיים שלכל $n > n_0$ מתקיים $T(n) < c \cdot n$.

לכן $T(n) = O(n)$.

סעיף ב'

נפתח את הביטוי הרקורסיבי בשיטת העץ כפי שנלמד בתרגול:



...

נחשב סכום כל שורה:

$$R_0 = n^2$$

$$R_1 = (\frac{n}{2})^2 + a \cdot (\frac{n}{4})^2 = n^2 \cdot (\frac{a+4}{4^2}) = n^2 \cdot (\frac{a+4}{16})^1$$

$$R_2 = (\frac{n}{4})^2 + 2 \cdot a \cdot (\frac{n}{8})^2 + a^2 \cdot (\frac{a}{16})^2 = n^2 \cdot (\frac{a^2 + 8 \cdot a + 16}{16^2}) = n^2 \cdot (\frac{a+4}{16})^2$$

$$R_3 = ... = n^2 \cdot (\frac{a^3 + 3 \cdot 4a^2 + 3 \cdot 4^2 a + 4^3}{16^3}) = n^2 \cdot (\frac{a+4}{16})^3$$

ומכאן כבר ברור שמתקיים:

$$R_i = n^2 \cdot (\frac{a+4}{16})^i$$

כאשר בענף העץ הימני ביותר נמצא הביטוי:

$$a^i \cdot (\frac{n}{4^i})^2$$

ובענף העץ השמאלי ביותר נמצא הביטוי:

$$(\frac{n}{2^i})^2$$

כלומר ענף העץ הימני יגיע לתנאי העצירה כאשר:

$$\frac{n}{4^i} \leq 10 \Rightarrow i = \log_4(\frac{n}{10}) = \log_4(n) - \log_4(10)$$

ואילו ענף שמאל יגיע לעצירה כאשר:

$$\frac{n}{2^i} \leq 10 \Rightarrow i = \log(\frac{n}{10}) = \log(n) - \log(10)$$

נשים לב שאם נסכום לפי i, נקבל סדרה הנדסית שמנתה $\frac{a+4}{16}$.
לכן כאשר $\frac{a+4}{16} < 1$, כלומר $a < 12$, נקבל שהסכום $\sum_{i=0}^{\infty}(\frac{a+4}{16})^i$ מתכנס למספר קבוע S, ולכן עבור כל תת־סכום סופי שלו נקבל סיבוכיות קבועה.
כלומר עבור $a < 12$ קיימים $S > 0$ וגם $n_0 = 10$ כך שלכל $n > n_0$ מתקיים:

$$T(n) \leq \sum_{i=0}^{\log(n)-\log(10)} n^2 \cdot (\frac{a+4}{16})^i \leq n^2 \cdot \sum_{i=0}^{\infty}(\frac{a+4}{16})^i = S \cdot n^2$$

$$\Rightarrow \lim_{n\to\infty} \frac{T(n)}{n^2} \leq S \Rightarrow \lim_{n\to\infty} \frac{T(n)}{n^2} \neq \infty$$

לכל n חיובי מתקיים $T(n) > 0$ וגם $n^2 > 0$ ולכן לפי משפט של חדו"א נקבל שניתן להפוך את הגבול ולקבל:

$$\lim_{n\to\infty} \frac{n^2}{T(n)} \neq 0$$

ובפרט $T(n) \neq \omega(n^2)$ על פי ההגדרה.
לכן a הוא לפחות 12. נבדוק מה קורה עבור $a = 12$:

$$T(n) \geq \sum_{i=0}^{\log_4(n)-\log_4(10)} n^2 \cdot (\frac{12+4}{16})^i = n^2 \cdot \sum_{i=0}^{\log_4(n)-\log_4(10)} 1 = (\log_4(n) - \log_4(10)) \cdot n^2$$

ולכן נקבל לפי כלל הפיצה (מאינפי 1):

$$\lim_{n\to\infty} \frac{T(n)}{n^2} \geq \lim_{n\to\infty} \frac{(\log_4(n) - \log_4(10)) \cdot n^2}{n^2} = \lim_{n\to\infty} \log_4(n) - \log_4(10) = \infty$$

כלומר לפי משפט של חדו"א נקבל:

$$\lim_{n\to\infty} \frac{n^2}{T(n)} = 0$$

ולכן לפי ההגדרה $T(n) = \omega(n^2)$ עבור $a = 12$, והראינו שזהו ה־a המינימלי המקיים זאת.
נמצא את ערכה האסימפטוטי של $T(n)$:
כבר הראינו $T(n) \geq (\log_4(n) - \log_4(10)) \cdot n^2$, לכן עבור $c = 4$, $n_0 = 100$ נקבל שלכל $n > n_0$ מתקיים:

$$4 \cdot T(n) \geq 4 \cdot (\log_4(n) - \log_4(\sqrt{n})) \cdot n^2 \geq 4 \cdot (\log_4(n) - 4 \cdot \frac{1}{2}\log_4(n)) \cdot n^2 =$$

$$= 4 \cdot \frac{1}{2}\log_4(n) \cdot n^2 = 4 \cdot \frac{1}{2} \cdot \frac{\log(n)}{\log 4} \cdot n^2 = 4 \cdot \frac{1}{4} \cdot \log(n) \cdot n^2 = \log(n) \cdot n^2$$

ולכן לפי ההגדרה $T(n) = \Omega(n^2 \cdot \log(n))$.
באופן דומה מהענף השמאלי של העץ, עבור $c = 1$ וגם $n_0 = 10$ נקבל שלכל $n > n_0$ מתקיים:

$$\cdot T(n) \leq \cdot (\log_2(n) - \log_2(10)) \cdot n^2 \leq \log_2(n) \cdot n^2$$

ולכן $T(n) = O(n^2 \log(n))$, ובסך הכול $T(n) = \Theta(n^2 \log(n))$.

סעיף ג'

$$T(n)=8T(\frac{n}{2})+n^3+n\cdot\log^2 n$$

מקרה זה מתאים לשיטת המאסטר, כאשר:

$$a=8,\ b=2,\ f(n)=n^3+n\cdot\log^2 n\Rightarrow n^{\log_b a}=n^3$$

נוכיח $f(n)=\Theta(n^3)$:
עבור $c=2$ ועבור $n_0=1$ נקבל שלכל $n>n_0$ מתקיים:

$$n^3+n\cdot\log^2 n\le n^3+n\cdot n^2=2\cdot n^3=c\cdot n^3$$

ולכן לפי ההגדרה $f(n)=O(n^3)$.
כמו כן עבור $c=1$ ועבור $n_0=1$ נקבל שלכל $n>n_0$ מתקיים:

$$n^3+n\cdot\log^2 n\ge n^3$$

ולכן לפי ההגדרה $f(n)=\Omega(n^3)$, ובסך הכול $f(n)=\Theta(n^3)$.
לכן לפי שיטת המאסטר נובע:

$$T(n)=\Theta(n^{\log_b a}\cdot\log n)=\Theta(n^3\cdot\log(n))$$

וזהו החסם ההדוק ביותר.

$$T(n)=2T(\frac{n}{2})+\sqrt{n}$$

נשתמש שוב בשיטת המאסטר:

$$a=2,\ b=2,\ f(n)=\sqrt{n}\Rightarrow n^{\log_b a}=n$$

נוכיח $f(n)=O(n^{\log_b a-\epsilon})$ עבור $\epsilon=\frac{1}{2}$.
אך זה טריוויאלי, שכן עבור $c=1$ ו־$n_0=1$ נקבל שלכל $n>n_0$ מתקיים:

$$f(n)=\sqrt{n}=n^{\frac{1}{2}}=n^{1-\frac{1}{2}}=n^{\log_b a-\epsilon}$$

כלומר לפי ההגדרה $f(n)=O(n^{\log_b a-\epsilon})$ ולכן לפי שיטת המאסטר נובע:

$$T(n)=\Theta(n^{\log_b a})=\Theta(n)$$

וזהו החסם ההדוק ביותר.

$$T(n) = n^2 + T(\frac{n}{2}) + ... + T(\frac{n}{2^k})$$

נראה $T(n) = \Theta(n^2)$:
נוכיח תחילה $T(n) = O(n^2)$ באינדוקציה שלמה.
<u>בסיס:</u>
עבור $n = 1$ ניקח $c = 2$ ונקבל:

$$T(1) = 1 \leq 2 \cdot 1^2 = c \cdot n^2$$

<u>צעד:</u>
נניח שעבור $c = 2$ ו־$n_0 = 1$ מתקיים שלכל $n > k > n_0$:

$$T(k) \leq c \cdot k^2$$

אזי עבור $T(n)$ נקבל:

$$T(n) = n^2 + T(\frac{n}{2}) + ... + T(\frac{n}{2^k}) \leq n^2 + c \cdot (\frac{n}{2})^2 + c \cdot (\frac{n}{4})^2 + ... + c \cdot (\frac{n}{2^k})^2 =$$
$$= n^2 + n^2 \cdot c \cdot (\sum_{i=1}^{k} \frac{1}{4^i}) \leq n^2 + n^2 \cdot c \cdot (\sum_{i=1}^{\infty} \frac{1}{4^i}) = n^2 + n^2 \cdot c \cdot \frac{\frac{1}{4}}{1 - \frac{1}{4}} = n^2 + n^2 \cdot c \cdot \frac{1}{3} \leq$$
$$\leq n^2 \cdot c \cdot \frac{2}{3} + n^2 \cdot c \cdot \frac{1}{3} = n^2 \cdot c$$

כלומר $T(n) \leq c \cdot n^2$ ולכן לפי עיקרון האינדוקציה הטענה נכונה לכל $n > n_0$.
ולכן נובע שקיים $n_0 = 1$ כך שלכל $n > n_0$ מתקיים $T(n) \leq c \cdot n^2$ עבור $c = 2$, ולכן:

$$T(n) = O(n^2)$$

כעת נוכיח $T(n) = \Omega(n^2)$, אך הטענה טריוויאלית, שכן עבור $c = 1$ וגם $n_0 = 1$ נקבל שלכל $n > n_0$:

$$T(n) = n^2 + \sum_{i=1}^{k} T(\frac{n}{2^i}) \geq n^2$$

ולכן $T(n) = \Omega(n^2)$ ובסך הכול $T(n) = \Theta(n^2)$, וזהו החסם ההדוק ביותר.

$$T(n) = 16 \cdot n^4 \cdot T(\sqrt{n}) + 2 \cdot n^8 \cdot \log^4(n)$$

נשתמש בפישוט מקדמים:

$$\frac{T(n)}{n^8} = 16 \cdot \frac{T(\sqrt{n})}{n^4} + 2 \cdot \log^4(n)$$

$$U(n) = \frac{T(n)}{n^8} \Rightarrow U(n) = 16 \cdot U(\sqrt{n}) + 2 \cdot \log^4(n)$$

ואז נבצע החלפת משתנה:

$$S(n) = U(2^n) \Rightarrow S(n) = 16 \cdot U(2^{\frac{n}{2}}) + 2 \cdot \log^4(2^n) =$$
$$=16 \cdot S(\frac{n}{2}) + 2 \cdot n^4$$

וכעת ניתן להשתמש בשיטת המאסטר:

$$a = 16,\ b = 2 \Rightarrow \log_b(a) = 4$$

$$f(n) = 2 \cdot n^4 = \Theta(n^4) = \Theta(n^{\log_b(a)}) \Rightarrow$$

$$\Rightarrow S(n) = \Theta(n^4 \cdot \log(n)) \Rightarrow$$

$$\Rightarrow U(n) = S(\log(n)) = \Theta(\log^4(n) \cdot \log(\log(n))) \Rightarrow$$

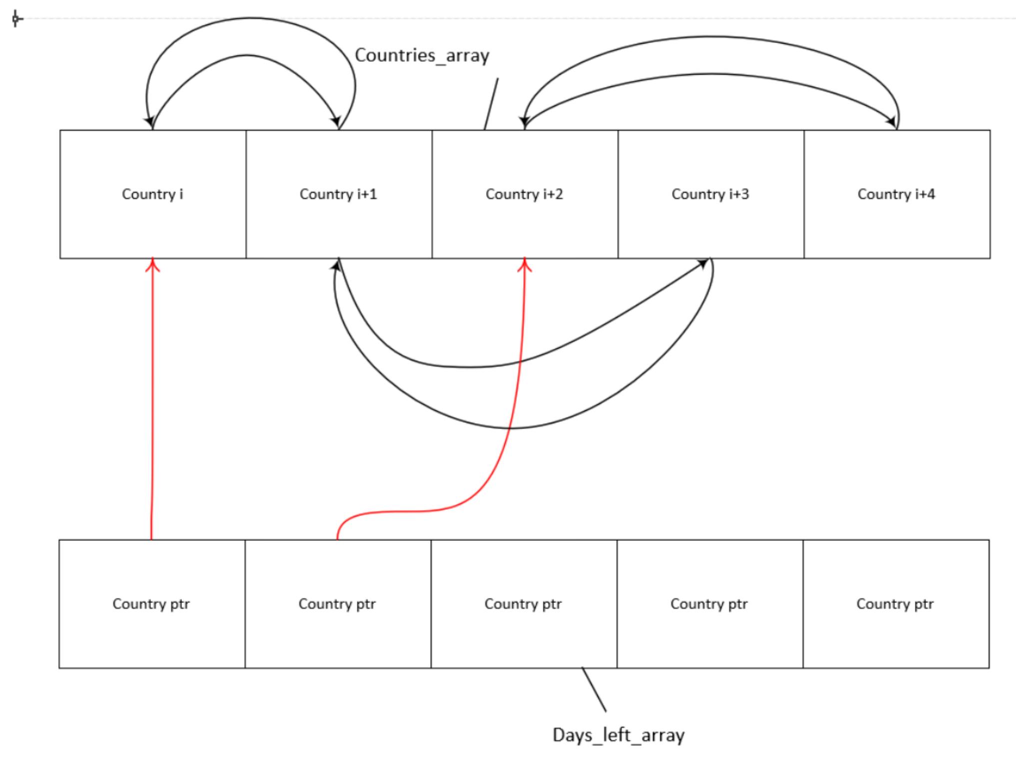
$$\Rightarrow T(n) = U(n) \cdot n^8 = \Theta(n^8 \cdot \log^4(n) \cdot \log(\log(n)))$$

וזהו החסם ההדוק ביותר.

שאלה 4

מבנה הנתונים כולל:

- struct בשם $country$ שהוא node של רשימה מקושרת דו כיוונית: תא המכיל משתנה ששמו $total\_days$, ו2 תאים לכתובת הבאה והקודמת של country. מבנה זה מכיל 3 משתנים בסה"כ.
- מערך $Countries\_array$ באורך $N+1$ המכיל איברים מסוג country (הגודל הוא $N+1$ כדי שהמדינה ה-i תהיה באינדקס ה-i, כלומר האינדקס 0 לא משמש אותנו כאן). המשמעות של מערך זה היא שהוא שומר את כל המדינות שהוכנסו כבר למערכת.
- משתנה $days\_passed$ ששומר כמה ימים עברו מהיום שבוצע $Init$ מודולו $D+1$. כלומר $Days\_from\_init\%(D+1)$.
- מערך $Days\_left\_array$ באורך $D+1$ המכיל מצביעים לאיברים מסוג country. המשמעות של מערך זה היא שהוא שומר במקום ה-i מצביע לאיבר הראשון ברשימה המקושרת של כל המדינות שנותרו $\left(i-days_{passed}\right)\%(D+1)$ ימים לבחירות שלהן.
- 2 מערכים $Days\_left\_array\_1$ ו-$Days\_left\_array\_2$ שכל אחד מהם מאורך $D+1$, ומשתנה $Days\_left\_top$, אשר שנועדו לאפשר תמיכה ב-$Init$ בסיבוכיות זמן של $O(1)$.
- 2 מערכים $Countries\_array\_1$ ו-$Countries\_array\_2$ שכל אחד מהם מאורך $N+1$, ומשתנה $Countries\_top$, אשר נועדו לאפשר תמיכה ב-$Init$ בסיבוכיות זמן של $O(1)$.
- מערך $Countries\_index$ באורך $N+1$ (שימוש במערך זה יודגם בהמשך).



הסבר על סיבוכיות המקום:
יש לנו 4 מערכים באורך N + 1 כאשר באחד מהם (מערך $Countries\_array$) יש לכל היותר $3 \cdot (N+1)$ איברים בגלל שהוא מכיל איברים מסוג country. בנוסף, יש לנו 3 מערכים באורך D + 1, ועוד 3 משתני עזר. סה"כ יש לכל היותר $3(D+1)+3(N+1)+3(N+1)+3=3D+6N+12$ משתנים בסה"כ, ולכן סיבוכיות המקום היא $O(N+D)$.

- $Init(N,D)$: נבצע את האלגוריתם של init שלמדנו בכיתה ב- O(1) ונאתחל את $Countries\_array$, $Days\_left\_array$. המערך $Countries\_index$ מסונכרן עם $Countries\_array$. כלומר באינקס i ב-$Countries\_array$ יש ערך חוקי אם"ם יש ערך חוקי באינקס i ב-$Countries\_array$. נאפס את משתני ה-top כדי לתמוך ב-$Init$. בנוסף נאתחל את days_passed ל-0.
סיבוכיות הזמן היא $O(1)$ בהנחה שסיבוכיות הזמן של הקצאת זיכרון היא $O(1)$.

- $Add\_country(i,D_i,days)$: נוסיף את המדינה למערך $Countries\_array$ (נעדכן $total\_days=D_i$ בתוך המשתנה $country$ החדש שייצרנו) ואז נוסיף את מדינה זו להתחלה של הרשימה המקושרת שכתובת תחילתה נמצאת באינדקס $(days+days\_passed)\%(D+1)$ במערך $Days\_left\_array$. נבצע השמה $Countries\_index\ [i]=(days+days\_passed)\%(D+1)$. ההסבר להגיון הוא כזה: נניח שהוספנו מספר מדינות כלשהו ביום כלשהו ונותרו 50 ימים לבחירות עבור כל המדינות הללו. בכל יום שעובר נותר יום אחד פחות לבחירות. אם למשל הוספנו את המדינות הללו ביום שאתחלנו בו את מבנה הנתונים, אז הרשימה המקושרת של המדינות הללו תתחיל באינדקס ה-50 במערך $Days\_left\_array$. לאחר 5 ימים שחלפו, האינדקס ה-50 כבר אינו מייצג את המדינות שנותרו להם 50 ימים לבחירות, אלא מייצג את המדינות שנותרו להם 45 ימים לבחירות, ולכן המשתנה $days\_passed$ נוצר ומתעדכן בשביל לאפשר לעקוב אחר הימים שחולפים. כעת לאחר 5 ימים אם נוסיף מדינה חדשה שיש לה 45 ימים לבחירות, אז נצטרך להוסיף אותה לרשימה המקושרת הנמצאת באינדקס ה-50, כי הרי המדינות הנמצאות שם כרגע מייצגות את המדינות שנותרו להם 45 ימים לבחירות. לכן לאחר 5 ימים כאשר נוסיף מדינה חדשה שנותרה לה 45 ימים לבחירות, נצטרך למקם אותה באינדקס $(\text{days}+\text{days_passed})\%(D+1)$במערך Days_left_array.
סיבוכיות הזמן היא O(1) מכיוון שאנו מבצעים מספר קבוע של פעולות של השמה וקריאה מהזכרון, שלא תלוי בגדלי המערכים.

- $Elections\_today()$: תחילה נוסיף 1 למשתנה $days\_passed$ כדי לעדכן את העובדה שעבר יום. אם $days\_passed==D+1$, כלומר אם עברו $D+1$ ימים, נאפס את $days\_passed$. נעבור ונדפיס את כל המדינות הנמצאות ברשימה המקושרת שכתובת תחילתה נמצא באינדקס $days\_passed$ במערך $Days\_left\_array$. לאחר הדפסה של כל מדינה, נעביר את המדינה לרשימה המקושרת שכתובת התחלתה נמצא באינדקס $(total\_days+days\_passed)\%(D+1)$, ועבור כל מדינה i שהדפסנו נעדכן $Countries\_index\ [i]=(total\_days+days\_passed)\%(D+1)$ כדי לזכור באיזה אינדקס של רשימה מקושרת כל מדינה נמצאת.

סיבוכיות הזמן היא O(k) מכיוון שאנחנו עוברים בדיוק על k מדינות – כל המדינות שאנו מדפיסים, ועבור כל אחת ממנה אנו מבצעים מספר קבוע של פעולות (הדפסה, העברה לרשימה מקושרת דו כיוונית אחרת).

- $Predate\_Elections(i, days)$: נשלוף את $Countries\_index\ [i]$. אם $days > (Countries\_index\ [i] - days\_passed)\%(D+1)$ זה אומר שהערך של $days$ לא תקין (למשל אם $D = 50, Countries\_index\ [i] = 10, days\_passed = 2, days = 12$ אז זה אומר שנותרו $Countries\_index\ [i] - days\_passed = 8$ ימים לבחירות, והרי שלא נוכל להקדים אותן ב12 ימים). לכן במקרה זה נחזיר ערך שגיאה. אחרת, נעביר את המדינה i לתחילתה של הרשימה המקושרת הנמצאת באינדקס $(Countries\_index\ [i] - days)\%(D+1)$. כלומר, בעצם הקדמנו את יום הבחירות של המדינה ב-days ימים. נעדכן גם $Countries\_index\ [i] = (Countries\_index\ [i] - days)\%(D+1)$. לדוגמא: נניח שהכנסנו את מדינה מספר 1 ביום שביצענו init והגדרנו שנותרו 50 ימים לבחירות של המדינה (כלומר $Countries\_index\ [i] = 50$), ונניח ועברו 5 ימים מאז שהכנסנו את המדינה למערכת. כעת נותרו 45 ימים לבחירות. המדינה כעת ברשימה המקושרת שכתובת התחלתה נמצא באינדקס 50 במערך $Days\_left\_array$. נבצע $Predate\_Elections(1,2)$ כלומר נרצה להקדים את הבחירות של מדינה מספר 1 ביומיים. כעת נעביר את מדינה מספר 1 לרשימה המקושרת המתחילה באיבר הנמצא באינדקס $(Countries\_index\ [i] - days)\%(D+1)$ כלומר לרשימה המקושרת הנמצאת באינדקס $50 - 2 = 48$. כעת לאחר 48 ימים שיעברו בסה"כ המשתנה $days\_passed$ יהיה 48 והבחירות במדינה מספר 1 יתקיימו בזמן זה ולא לאחר 50 ימים כפי שהיה אמור להיות בהתחלה.

סיבוכיות הזמן היא $O(1)$ מכיוון שבחרנו להשתמש ברשימה מקושרת דו כיוונית, וברשימה מקושרת מסוג זה סיבוכיות הזמן להוציא איבר מהרשימה ולהכניס איבר אל הרשימה היא O(1) כאשר יש לנו את הכתובת של האיבר מראש (ויש לנו את הכתובת כי אנחנו שומרים במערך $Countries\_array$ את כל האיברים הנמצאים בכל הרשימות המקושרות).

שאלה 5

מבנה הנתונים כולל מטריצה בגודל n עמודות ו־6 שורות, הממומשת כמערך (מעתה תיקרא $items$).

כל עמודה מסמלת דגם שונה, וכל שורה מתוך ה־5 הראשונות מסמלת מידה שונה.

בכל תא בשורות אלו יימצא מספר המכיל את מספר הפריטים שנותרו מהמידה ומהדגם המתאים.

השורה השישית התחתונה תכיל במהלך פעולת המבנה מצביעים לחוליות ברשימה שתתואר, כאשר כל מצביע מתאים לחוליה ברשימה שמתארת את הדגם בעמודה שלו.

בנוסף המבנה כולל רשימה דו כיוונית **ממוינת** (מעתה תיקרא $model\_sells$), שבה כל חוליה מכילה מספר ומצביע ל"תת־רשימה" דו כיוונית, כאשר בתתי הרשימות כל חוליה מכילה ערך מספרי שהוא מספר הדגם שאותו היא מתארת, וכמו כן מצביע אל ראש הרשימה שבה הדגם נמצא (כלומר אל החוליה המתאימה ב־$model\_sells$).

כאשר מבנה הנתונים יתמלא, החוליות ב־$model\_sells$ יסמלו כל אחת כמות מחירות כלשהי, ובתת הרשימה שנמצאת בכל חוליה יימצאו כל מספרי הדגמים שמכרו את הכמות הזו באותו היום.

על הרשימה $model\_sells$ נשמור ממוינת באופן שיתואר בהמשך, כאשר בסוף הרשימה נמצאת חוליית הדגמים שמכרו את מספר הפריטים הנמוך ביותר (שאינו 0), ובראש הרשימה נמצאת רשימת הדגמים שמכרו את מספר הפריטים הרב ביותר ביום.

לרשימה זו המבנה שומר 2 מצביעים: מצביע לתחילת הרשימה ומצביע לסופה (על מנת שניתן יהיה לגשת לשניהם באופן מיידי).

בנוסף, המבנה מאחסן 2 מונים מספריים (ייקראו מעתה $sold$ ו־$returned$) המונים את מספר הפריטים הכולל שנמכרו ומספר הפריטים שהוחזרו.

<u>$Init(n,c)$</u> - המטריצה $items$ מאותחלת ב־$O(1)$ לפי השיטה שראינו בהרצאה.

ערך האתחול עבור התאים ב־5 השורות הראשונות הוא c, וערך ההתחול עבור התאים בשורה התחתונה הוא מצביע ל־$NULL$.

נציין שבביצוע האתחול אין חשיבות לכך ששורות שונות מתאימות לערכי אתחול שונים, זאת משום שבאלגוריתם שתואר בהרצאה ניתן לקבל עבור כל תא במערך את המידע "האם התא מכיל את ערכו ההתחלתי או לא", לכן כאשר תתבצע גישה לתאי המערך, אם נקבל מהאלגוריתם שהתא מכיל את ערכו ההתחלתי, נוכל להשתמש בביטוי תנאי הבודק את השורה שאליה התבצעה גישה, ובהתאם מחזיר c או מצביע ל־$NULL$ בסיבוכיות $O(1)$ עבור כל גישה.

המונים $sold, returned$ מאותחלים שניהם ל־0.

המצביעים לתחילת הרשימה הדו כיוונית $model\_sells$ מאותחלים שניהם ל־$NULL$, כלומר באתחול המבנה הרשימה ריקה.

בסך הכול כל הפעולות מתבצעות בסיבוכיות $O(1)$.

<u>$Return(i,k)$</u> - ראשית נבצע בדיקה ש־k בטווח המידות החוקי, ואחרת נחזיר שגיאה.

כעת נבצע גישה אל התא במטריצה המתאים לדגם ולמידה שהוחזרו: $items[i-1][k]$.

נגדיל את מספר הפריטים שנותרו מאותם מידה+דגם ב־1 בתוך התא, ולאחר מכן נגדיל את המונה $returned$ ב־1.

כל הפעולות התבצעו ב־$O(1)$ (גישה למערך והגדלת מונים), לכן זוהי גם הסיבוכיות הכוללת.

$\underline{Sold()}$ ־ נחזיר את ערכו של המונה $sold$. בהמשך נתאר כיצד אנו מגדילים אותו בכל פעולת קנייה, כך שערכו תקין. (בסיבוכיות $O(1)$).

$\underline{Returned()}$ ־ נחזיר את ערכו של המונה $returned$. ערכו תקין מכיוון שבכל פעולת החזרה הגדלנו אותו ב־1 (בסיבוכיות $O(1)$).

$\underline{Popular(m)}$ ־ בהנחה שאנו שומרים על היותה של רשימת $model\_sells$ ממוינת (בתיאור הפעולה Buy נסביר כיצד עושים זאת), מספיק להדפיס את m האיברים הראשונים שבה:

קיימת גישה מיידית לראש הרשימה משום שאנו שומרים אליו מצביע, וכעת ניתן לגשת אל רשימת הדגמים שמכרו את מספר הפריטים הרב ביותר (היא נמצאת בחוליה הראשונה ב־$model\_sells$) ולהדפיס את המספרים שכל החוליות בה מכילות בצורה איטרטיבית, תוך שמירה של מונה לוקאלי הסופר את מספר הדגמים שהדפסנו, ותוך שמירה על המצביע לחוליה שאת הרשימה שבתוכה אנו מדפיסים.

אם בשלב מסוים הדפסנו m דגמים נעצור, ואם הדפסנו את רשימת כל הדגמים שמכרו הכי הרבה ועדיין לא הדפסנו m דגמים, נוכל לעבור לרשימת הדגמים הבאה אחריה ב־$model\_sells$ בסיבוכיות $O(1)$ (כי יש לנו מצביע לחוליה הנוכחית שאנו מדפיסים ב־$model\_sells$) ולהמשיך את ההדפסה של איבריה, שהם הדגמים הבאים בתור מבחינת כמות המכירות. ברשימה מעבר עד האיבר ה־m היא בסיבוכיות $O(m)$, לכן אם נעבור ונדפיס את המספרים השמורים בכל חוליה, נקבל את m הדגמים הנמכרים ביותר בסיבוכיות $O(m)$.

$\underline{IsSuccessful()}$ ־ קיימת לנו גישה ב־$O(1)$ לערכים שאנו שומרים $sold$ ו־$returned$, ולכן ב־$O(1)$ נוכל להשוות ביניהם, ולהחזיר $true$ אם ורק אם $sold$ גדול יותר.

$\underline{Buy(i,k)}$ ־ זוהי הפעולה המורכבת ביותר, משום שעליה לשמור על היותה של רשימת $model\_sells$ ממוינת ותקינה.

ראשית מתבצע וידוא ש־k בטווח המידות החוקי, ואחרת מוחזרת שגיאה.
לאחר מכן מתבצעת גישה אל התא המתאים במטריצת הדגמים, כלומר אל $items[i-1][k]$ וקריאה של ערכו.
אם ערך התא 0, סימן שלא נותרו עוד פריטים ומוחזר הערך $sold\,out$.
אחרת, נקטין את ערך התא ב־1 ונגדיל את ערך המונה $sold$ ב־1.

הפעולות שיתוארו כעת מוודאות שהרשימה $model\_sells$ נותרת תקינה וממוינת לאחר הקנייה:

נקרא את ערך המצביע המתאים לדגם שנקנה, הנמצא ב־$items[i-1][5]$ בגישה ישירה.

אם זוהי **הפעם הראשונה** שדגם זה נקנה, המצביע יהיה ריק. במקרה זה ניצור חוליה חדשה לתת רשימה דו כיוונית, שהשדה המספרי שלה, המתאר את מספר הדגם, מאותחל ל־$i-1$.

אם החוליה שאנו מעדכנים היא הראשונה שנקנתה היום, כלומר הרשימה $model\_sells$ ריקה, ניצור חוליה ראשונה גם ב$model_sells$, שתוכנה יהיה המספר 1 (שכן היא מתארת את הפריטים שנקנו פעם אחת) ומצביע לתת רשימה חדשה, שלתוכה נכניס את החוליה שיצרנו עבור הדגם שזה עתה נקנה. כל הפעולות הללו מתבצעות ב־$O(1)$.

אחרת, אם נקנו היום חוליות לפני החוליה שנקנתה עתה, סימן ש־$model\_sells$ כבר קיימת, וגם החוליה בסופה מכילה את רשימת הדגמים שמכרו את מספר הפריטים הנמוך ביותר.

אם מספר זה הוא 1, נותר רק להכניס לתוכה את החוליה החדשה שזה עתה יצרנו, והדבר מתבצע ב־$O(1)$ כי שמרנו מצביע לסופה של $model\_sells$.

אחרת, יש ליצור חוליה בסוף $model\_sells$ המתארת את הדגמים שמכרו פריט אחד בלבד, ובתוכה ליצור רשימת חוליות דגמים, שאליה נכניס את הדגם שנקנה.

לבסוף נשמור בשני המקרים מצביע אל החוליה החדשה שנוצרה, בשורה האחרונה בעמודה המתאימה לדגם במטריצה $items[i-1][5]$ לגישה עתידית.

בסך הכול לאחר הוספת הדגם החדש לרשימה $model\_sells$, היא נשארת תקינה וממוינת לפי השמורה שהגדרנו עבורה.

נותר לטפל במקרה שבו זוהי **אינה הפעם הראשונה** שבה הדגם נקנה, כלומר כבר קיימת עבורו חוליה באחת מתתי הרשימות של $model\_sells$:

באמצעות המצביע המתאים לדגם, שנמצא בשורה האחרונה של המטריצה $items[i-1][5]$ נבצע גישה ישירה לחוליה המתאימה לדגם ב־$O(1)$.

ראשית נגדיל את $sells$ של הדגם בחוליה ב־1 (משום שהרגע נקנה פריט חדש מאותו הדגם).

כעת נשתמש במצביע שבתוך החוליה, המצביע לחוליה ב־$model\_sells$ שהכילה את תת הרשימה שבה הדגם היה.

מכיוון ש־$model\_sells$ דו כיוונית, נוכל לגשת ב־$O(1)$ לחוליה שלפני החוליה שיש לנו (כלומר לחוליית הדגמים הנמכרים יותר). אם המספר שבחוליה זו, מספר המכירות שהיא מתארת, שווה למספר בחוליית הדגם שאנו מעדכנים, נכניס לראשה ב־$O(1)$ את הפריט שזה עתה נקנה, כלומר מספר המכירות שלו אכן גדל ב־1 ולכן מקומו ברשימה זו.

אך קיים מקרה שבו לא קיימת רשימה המתארת את מספר המכירות המתאים, למשל אם הפריט שזה עתה נקנה היה הפריט הנמכר ביותר עד כה, או אם היו פריטים נמכרים יותר אך לא פריטים שמספר המכירות שלהם היה גדול בדיוק ב־1 ממספר המכירות הקודם של החוליה שאנו מעדכנים, כלומר לא קיימת חוליה ב־$model\_sells$ עם רשימת פריטים מתאימה.

במקרה זה נוכל ליצור חוליה כזו ב־$O(1)$, לקרוא את המספר שהיה בחוליה הקודמת ב־$model\_sells$, שהוא מספר המכירות של הפריט הנמכר ביותר עד כה, להוסיף 1 למספר זה ולשמור את הערך החדש בחוליה, שכעת יכולה להכיל את רשימת הדגמים שמכרו פריט אחד יותר. נכניס את הדגם שנקנה לרשימה שבתוכה, ונכניס אותה ל־$model\_sells$ באמצעות המצביע לחוליה שלפניה (שמהרשימה שבתוכה הוצאנו את הדגם שאנו מעדכנים), נקשר בינה לבין החוליה שאחריה, כל זאת ב־$O(1)$ משום ש־$model\_sells$ דו־כיוונית.

לאחר שהעברנו את חוליית הדגם הנמכר לרשימה אחרת ב־$model\_sells$, אם הרשימה הקודמת שבה היה נותרה ריקה, כלומר זה היה הדגם היחיד עם אותו מספר מכירות, נוציא מ־$model\_sells$ את החולייה המכילה את הרשימה הריקה, ומכיוון ש־$model\_sells$ היא רשימה דו כיוונית נוכל לעשות זאת ב־$O(1)$. הדבר מבטיח שלא נאגר עם הזמן ב־$model\_sells$ מספר לא חסום של חוליות עם רשימות ריקות.

הסיבוכיות הכוללת היא $O(1)$ משום שכל הפעולות היו הוצאה והכנסה של איברים מרשימות דו כיווניות באמצעות מצביעים שהיו קיימים אליהם.

<u>סיבוכיות מקום:</u>

לאחר הפעולה $Init$ קיימת במבנה המטריצה $items$. כל תא שלה מכיל ערך יחיד: עבור 5 השורות העליונות זוהי כמות הפריטים שנמכרות ועבור השורה התחתונה זהו מצביע שעתיד להצביע על חוליה ברשימה דו כיוונית.

מכיוון שהיא ממומשת באמצעות מערכים, ובמימוש המאפשר איתחול ב־$O(1)$ כפי שנלמד בכיתה, סיבוכיות המקום שלה היא לפי גודלה:

$$3 \cdot (n \cdot 6) = \Theta(n)$$

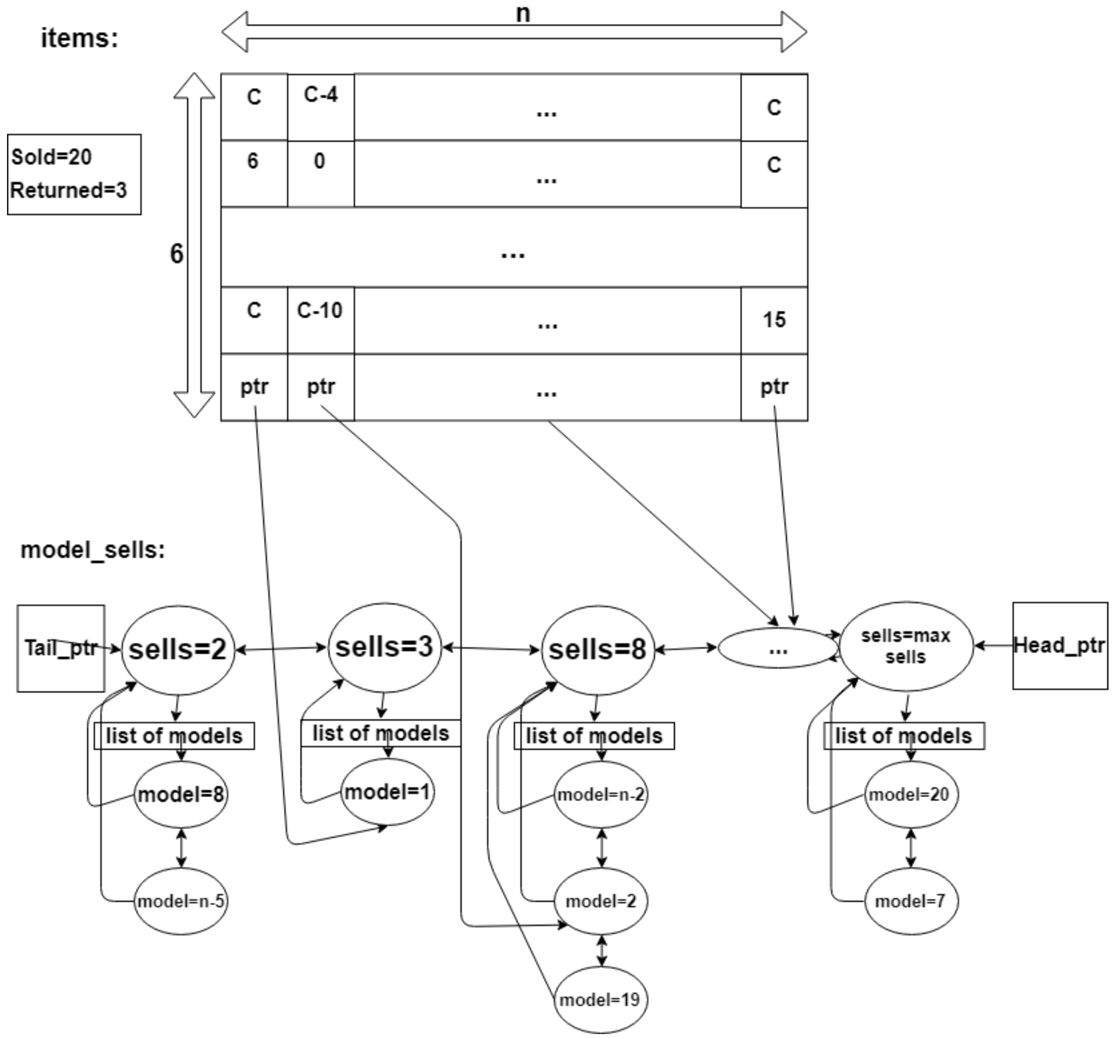
בנוסף המצביעים לתחילת הרשימה ולסופה, והמונים $sold$ ו־$returned$ כולם בסיבוכיות $O(1)$.

המצב "הגרוע ביותר" של המבנה מבחינת סיבוכיות מקום הוא כאשר נמכר לפחות פריט אחד מכל דגם, כי אז בנוסף למה שתואר לעיל, הרשימה $model\_sells$ תכיל את מספר החוליות הרב ביותר: היא תכיל לכל היותר n חוליות המכילות רשימות (במקרה שכל דגם מכר כמות שונה), אך מכיוון שלכל דגם קיימת באחת מתתי הרשימות הללו חולייה מתאימה אחת בלבד, נקבל ש־$model\_sells$ ותתי הרשימות שבה מכילות לכל היותר $n + n$ חוליות.

כלומר סיבוכיות המקום של $model\_sells$ ותתי רשימותיה היא בסדר גודל של $O(n)$
לכן במצב הגרוע ביותר, סיבוכיות המקום של המבנה היא בקירוב:

$$3 \cdot (n \cdot 6) + 2 \cdot n + 1 + 1 = \Theta(n)$$

להלן דיאגרמה המתארת את המבנה:


items:
n
6
Sold=20
Returned=3
C
C-4
...
C
6
0
...
C
...
C
C-10
...
15
ptr
ptr
...
ptr
model_sells:
Tail_ptr
sells=2
sells=3
sells=8
...
sells=max sells
Head_ptr
list of models
list of models
list of models
list of models
model=8
model=n-5
model=1
model=n-2
model=2
model=19
model=20
model=7