

שאלה 1

א

1

$$n(\log_2 n)^k = n\left(\frac{\log_{10} n}{\log_{10} 2}\right)^k = \frac{1}{(\log_{10} 2)^k} n(\log_{10} n)^k = O(n(\log_{10} n)^k)$$

נכון

2

$$\begin{aligned} 2^{\log_2 n} &= 2^{\frac{\log_{10} n}{\log_{10} 2}} = (2^{\frac{1}{\log_{10} 2}})^{\log_{10} n} \geq (2^{3.3})^{\log_{10} n} \\ &= 2^{2.3 \log_{10} n} 2^{\log_{10} n} \Rightarrow 2^{\log_2 n} \notin O(2^{\log_{10} n}) \end{aligned}$$

לא נכון

3

$$f(n) = c \log_2 n \Rightarrow 2^{f(n)} = 2^{c \log_2 n} = n^c \Rightarrow 2^{f(n)} \notin O(n)$$

נכון

4

$$\sum_{i=1}^n \sqrt{2^i} = \sum_{i=1}^n (\sqrt{2})^i = \frac{\sqrt{2}(\sqrt{2}^n - 1)}{\sqrt{2} - 1} = \frac{\sqrt{2}}{\sqrt{2} - 1} \sqrt{2}^n - \frac{\sqrt{2}}{\sqrt{2} - 1} \in O(\sqrt{2}^n)$$

נכון

ב

1

הלולאה מבצעת n איטרציות. בכל איטרציה מוסיפים לרשימה i איברים. כלומר שהפונקציה

$$\sum_{i=0}^{n-1} i = \frac{(n-1)n}{2} = \frac{n^2}{2} - \frac{n}{2} = O(n^2) \text{ מבצעת}$$

2

הלולאה מבצעת n איטרציות. בכל איטרציה מוסיפים ללולאה את מספר האיברים שיש ב

$$\text{lst} \quad (n + 2n + 4n + \dots + 2^{n-1}n) = \sum_{i=0}^{n-1} 2^i n = n \sum_{i=0}^{n-1} 2^i = \text{כלומר שהפונקציה מבצעת} \quad n(2^n - 1) = O(n2^n)$$

3

הלולאה מבצעת n איטרציות. בכל איטרציה יוצרים רשימה חדשה מאיברי הרשימה הנוכחית ועוד רשימת מספרים באורך זהה. כלומר שהפונקציה מבצעת

$$(2n + 2^2n + 2^3n + \dots + 2^nn) = \sum_{i=1}^n 2^i n = n \sum_{i=1}^n 2^i$$

$$= n \sum_{i=0}^n 2^i - 1 = n(2^{n+1} - 1) = O(n2^{n+1}) = O(n2^n)$$

4

הלולאה מבצעת n איטרציות. בכל איטרציה אנחנו מוסיפים לרשימה רשימה חדשה של איברים בגודל 10000 שלא ממש אכפת לנו מהאיברים שנמצאים בה. כלומר שהפונקציה

$$\sum_{i=0}^{n-1} 10000 = 10000n = O(n)$$

ג

בכל איטרציה של הלולאה מתווסף איבר לרשימה 1. הלולאה ממשיכה כל עוד היא לא הגיעה לאיבר האחרון ברשימה, ולכן היא לעולם לא מסתיימת.

שאלה 2

ב

-0.824

ג

התוכנית עוצרת כיוון שהיא מבצעת מקסימום מספר סופי של פעולות (עבור הארגומנטים הנתונים), אך רק לאחר זמן רב (עבור האפסילון הדיפולטי) כיוון שהיא נדרשת לבצע מעל ל 10^{13} פעולות. df

שאלה 3

א

בניית הרשימה all לוקחת $O(nm)$ זמן, וזהו גם גודלה. לולאת ה while מתבצעת $O(nm)$ פעמים (מוסיפים איבר אחד בכל פעם). בכל איטרציה מציאת המינימום היא הפעולה היקרה ביותר וסיבוכיותה $O(nm)$. לכן סהכ סיבוכיות הפונקציה היא $O(n^2m^2)$

ד

הלולאה מבצעת m איטרציות. בכל איטרציה הפונקציה ממזגת את הרשימה הממוזגת שהיא בנתה עד כה עם רשימה נוספת מ lst_of_lsts. כל מיזוג לוקח $O(n1 + n2)$. כלומר שסיבוכיות

הפונקציה היא

$$\begin{aligned} & ((0+n) + (n+n) + (2n+n) + (3n+n) + \dots + ((m-1)n+n)) \\ &= \sum_{i=0}^{m-1} (in+n) = n \sum_{i=0}^{m-1} i + \sum_{i=0}^{m-1} n = n \frac{(m-1)m}{2} + (m-1)n \\ &= \frac{nm^2}{2} + \frac{nm}{2} - n = O(nm^2) \end{aligned}$$

ה

הסיבוכיות הכללית של הדרך הראשונה היא:

$$O(nm \log_2(nm))$$

הסיבוכיות הכללית של הדרך השנייה היא:

$$O(mn \log_2 n + nm^2) = O(mn(\log_2 n + m))$$

a

הסיבוכיות של הדרך הראשונה:

$$O(n \log_2(cn)) = O(n(\log_2 n + \log_2 c)) = O(n \log_2 n)$$

הסיבוכיות של הדרך השנייה:

$$O(n(\log_2 n + c)) = O(n \log_2 n)$$

לכן מבחינת סיבוכיות שתי הדרכים עדיפות באותה מידה.

b

הסיבוכיות של הדרך הראשונה:

$$O(n \log_2 n \log_2(n \log_2 n)) = O(n \log_2 n (\log_2 n + \log_2 \log_2 n)) = O(n \log_2^2 n)$$

הסיבוכיות של הדרך השנייה:

$$O(n \log_2 n (\log_2 n + \log_2 n)) = O(n \log_2 n (2 \log_2 n)) = O(n \log_2^2 n)$$

לכן מבחינת סיבוכיות שתי הדרכים עדיפות באותה מידה.

c

הסיבוכיות של הדרך הראשונה:

$$O(n^2 \log_2 n^2)$$

הסיבוכיות של הדרך השנייה:

$$O(n^2(\log_2 n + n)) = O(n^3)$$

לכן מבחינת סיבוכיות הדרך הראשונה עדיפה.

שאלה 4

א

$$2c \log_2 100 = c \log_2 100^2 = c \log_2 10000 \Rightarrow n' = 10000$$

ב

$$2(c \cdot 100) = c \cdot 200 \Rightarrow n' = 200$$

ג

$$2c \cdot 100^2 = c \cdot 20000 = c \cdot \sqrt{2} \cdot 100 \Rightarrow n' = \lfloor \sqrt{2} \cdot 100 \rfloor = 141$$

ד

$$2c \cdot 2^{100} = c \cdot 2^{101} \Rightarrow n' = 101$$

שאלה 6

א

ראשית נשים לב כי השורש של $x^{1000000000}$ ושל x^2 הוא 1.

1

לפי חישוב הנגזרת ב0 של f_1 קרובה עד כדי אפסילון ל0 ולכן הגיוני שתודפס ההודעה בשורה 24. בנוסף השורש של f_2 הוא 1 ולכן הגיוני שקיבלנו $\text{root}=1$. לפיכך האפשרות תיתכן.

2

כיוון שהשורש של הפונקציה f_2 הוא 1 ו root קרוב ל0 האפשרות לא תיתכן.

3

לפי חישוב הנגזרת שמחשב האלגוריתם ב0 של f_1 היא 0.001 כיוון שאינה קרובה עד כדי אפסילון ל0, לא יתכן כי מודפסת ההודעה בשורה 24.

4

כיוון שהשורש של הפונקציה f_2 הוא 1 ו root קרוב ל0 האפשרות לא תיתכן.