**שאלה – זמני ריצה של אלגוריתמים רקורסיביים**

לכל אחת מנוסחאות הנסיגה הבאות, תנו דוגמה לאלגוריתם שנלמד בקורס, שהנוסחה מתארת את זמן הריצה שלו (אם קיים כזה) במקרה טוב/גרוע/ביניים, וציינו מהו זמן הריצה האסימפטוטי (במונחים של O).

להלן תזכורת לאלגוריתמים רקורסיביים שראינו בקורס:

* חיפוש בינארי
* מיון מהיר
* מיון מיזוג
* Select (לבעיית הבחירה)
* העלאה בחזקה (תרגיל בית 4 שאלה 3): 2 קריאות רקורסיביות והשיפור עם קריאה בודדת
* עצרת
* חישוב מקסימום (שתי גרסאות: 1:n-1 ו- ½n: ½n)

|  |  |  |
| --- | --- | --- |
| Recurrence | Solution | Examples |
| t(n) = t(n/2) + 1 | O(logn) | * Binary search
* Exponentiation with 1 call from HW3 Q4
 |
| t(n) = t(n/2) + n | O(n) | * Select, an intermediate case
 |
| t(n) = 2t(n/2) + 1 | O(n) | * Exponentiation with 2 calls from HW3 Q4
* Max (½n: ½n)
 |
| t(n) = 2t(n/2) + n | O(nlogn) | * Mergesort
* QuickSort best (and average)
 |
| t(n) = t(n-1) + 1 | O(n) | * Factorial
* Max (the 1:n-1 bersion)
 |
| t(n) = t(n-1) + n | O(n2) | * QuickSort worst
* Select worst
 |

**שאלה – חיפוש בינארי**

1. (15 נק') כתבו פונקציה בשם find\_steady שמקבלת כקלט רשימה *L* של מספרים שלמים שונים, ממוינת בסדר עולה, ומחזירה כפלט אינדקס  *i ≥*0 המקיים  *L*[*i*] == *i*.

אם לא קיים כזה, יוחזר הערך None.

הסבירו ראשית את האלגוריתם במילים, כיתבו אותו בקוד פייתון, וציינו מהי סיבוכיות זמן הריצה (בסימון O(…), כפונקציה של אורך הרשימה *n*).

על הפונקציה לרוץ בזמן יעיל. זמן ריצה לינארי יזכה בניקוד חלקי בלבד.

1. (10 נק') תנו דוגמה לרשימה ממוינת בסדר עולה שבה המספרים לא בהכרח שונים זה מזה, ועבורה הפתרון שלכם בסעיף א' לא עובד נכון. ציינו מה יחזיר האלגוריתם ומדוע פלט זה אינו נכון.

**שאלה – hash**

נתונה רשימה *L* של *n* מספרים כלשהם ומספר נוסף *z*.
אנו מעוניינים באלגוריתם שמחזיר זוג מספרים ב- *L* שסכומם *z.* אם אין זוג מספרים כזה, האלגוריתם יחזיר None.

1. תארו במילים אלגוריתם נאיבי, ונתחו את זמן ריצתו.
2. כיתבו אלגוריתם שרץ בסיבוכיות זמן O(*n*log*n*) במקרה הגרוע.
3. כיתבו אלגוריתם שרץ בסיבוכיות זמן ליניארית (O(*n*)) בממוצע (אך לא בהכרח במקרה הגרוע).