Hw4_212077333

Friday, May 19, 2023 6:38 PM

:1 שאלה

: סעיף א

דטרמיניסטי	רנדומלי	גודל
0.0002843200010829605	0.00033103000168921426	(100,10)
0.002427003998309374	0.0027644680003868416	(1000,25)
0.0084204719986883	0.009925205999461468	(2500,50)
0.06074533500010148	0.06780176800049958	(5000,100)

לכן ניכר שעדיף את האלגוריתם הדטרמיניסטי (כי בחירת האיבר הרנדומלי מאיטה אותנו)

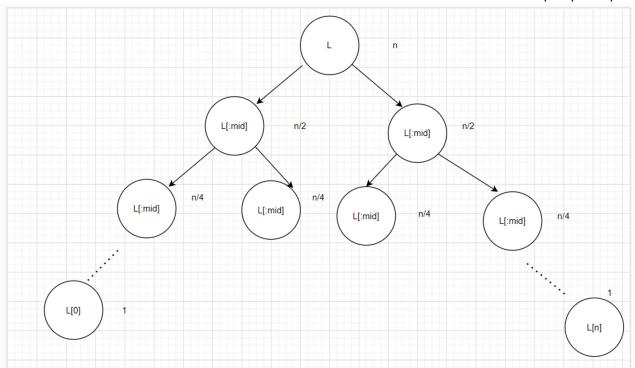
: סעיף ב

גודל	דטרמיניסטי	רנדומלי
(100,10)	0.0005859500030055642	0.002335250000760425
(500,25)	0.020772587999817915	0.0015375920006772503
(750,25)	0.04450219199934509	0.002446947999414988
(750,50)	0.045125100000295786	0.0024503679989720693

לכן ניבר שעדיף האלגוריתם הרנדומלי

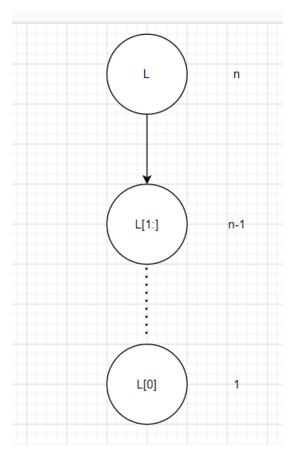
שאלה 2

סעיף א : עץ הרקורסיה



nlogn ובכל שלב אנו מבצעים O(n) ובכל שלב אנו מבצעים logn נשים שגובה העץ הוא סה"כ

סעיף ב : עץ הרקורסיה



זמן הריצה הוא סכום n^2)O= 1+2+3+...+n זמן

: סעיף ב

בשיפור של 10 גובה העץ הוא עדיין logn ובכל שלב אנחנו עדיין מצבעים (n) לכן (n) אך בשיפור של 10 איננו חותכים את המחרוזת שזו כאומר לוקחת N צעדים לכן זמן הריצה יהיה מהיר יותר (אנחנו בעצם מעבירים את המידע על הרשימה בדרך של אינדקסים ולא בדרך של העברת כל הרשימה שזה הרבה יותר חכם וחסכוני) בשיפור של 2V גובה העץ הוא N ובכל שלב אנחנו מבצעים פעולה אחת לכן (yay an improvement) O(n) כאשר ב2V ללא השיפור זמן הריצה היה (O(N)

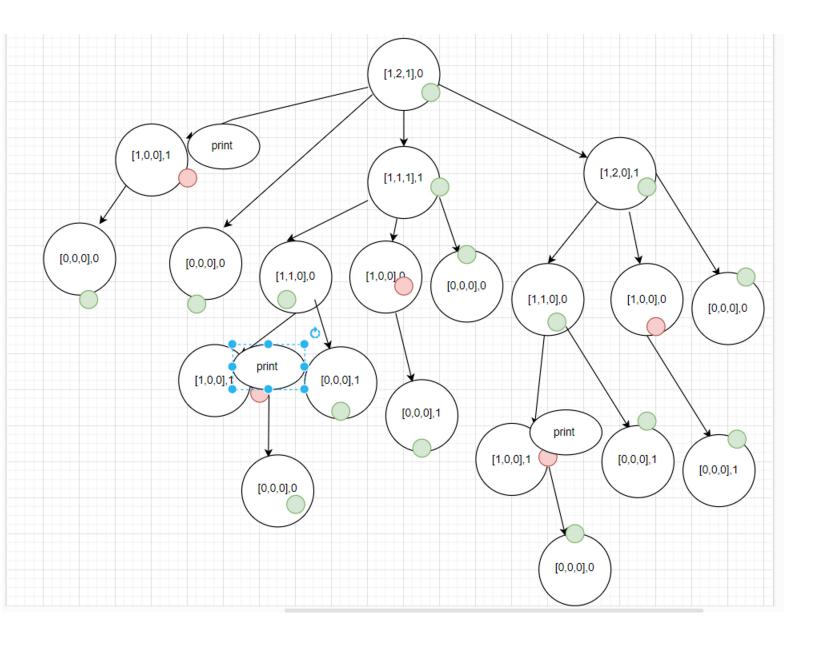
: סעיף ג

זמן הריצה של הפונקציה , נשים לב שיש לנו n/2 קריאות רקורסיביות כשבכל אחת אנו מבצעים (O(1) פעולות לכן זמן הריצה הוא O(N)

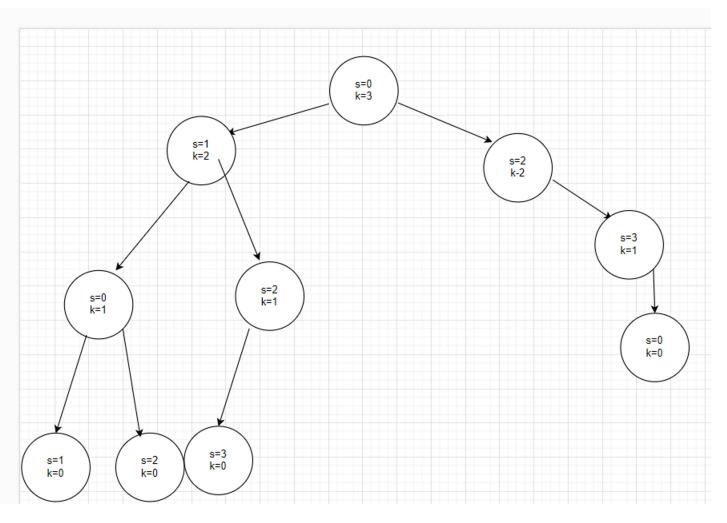
שאלה 3

: סעיף א

Player יודפס המהלך הממולץ לשחקן מספר recommended move: $[1, 2, 1] ext{ --> } [1, 0, 0]$ recommended move: $[1, 1, 0] ext{ --> } [1, 0, 0]$ recommended move: $[1, 1, 0] ext{ --> } [1, 0, 0]$



שאלה 4 סעיף ב : i. עץ הרקורסיה



ii.

נשים לב שלגרף שמכל נקודה יוצאת צומת לכל נקודה אחרת זמן הריצה הוא לפחות וכאשר k=n-1 זמן הריצה הוא לפחות (n-2)^(n-2) שזה כמובן זמן ריצה אקספוננציאלי א לבן 2−12 N−1×2 ו 0−1 א 1−2 (N-1)^n-1=2^N-1*(N-1)^N-1>c^N א לבן 2 לבן 2−1 N−1 לבן מתקיים כי לכל

: סעיף ג

נשים לב שעבור הקלט [A,0,0,1] [A,0,0,1] וגם ל [A,0,0,0] נבצע קריאה רקורסיבית ל הגענו ללופ אינסופי ולכן לא נקבל את הפלט הרצוי

lii

נבחר את הקלט ([0,0,0,...,0],...,[0,0,0,0,...,0],0,0,n-1)

נשים לב שיש לנו לולאה באורך n אשר קוראת לפונקציה הרקורסיבית 2 פעמים ובנוסף עומק העץ הוא (Log(n-1 לכן נקבל כי זמן הריצה הוא גדול מ (n-1) לכן

שזה כאומר סופרפולינומיאלי

סעיף ד

false א יוחזר st א יוחזר st עבור כל קלט או שניבנס ללופ אינסופי או שנעבור על כל הקשרים שלו ולכן אם יהיה מסלול בין

B עבור הקלט [[0,0,1],[1,0,1],[1,0,1],[0,1,1] נשים לב שכמובן שיש מסלול בין 0 ל 2 אך מהגדרת הפונקציה אנחנו B נתקע בלופ אינסופי של קפיצות בין 0 ל 1 זאת משום ש0 שולח ל 1 אך מהגדרת הפונקציה היא נשלחת למקום הראשון שבו מופיעה הספרה אחת לכן תחזור ל 0 וכך ימשך לאינסוף

True א יוחזר st לא יהיה מסלול בין st לא יהיה מסלול בין א לא יהיה מסלול בין C

שבור הקלט [[0,0,0],[1,0,0],[1,0,0]] (שים לב שכמובן שאין מסלול בין 0 ל 2 אך מהגדרת הפונקציה אנחנו (A,0,2) A=[[0,1,0],[1,0,0],[0,0,0]] נתקע בלופ אינסופי של קפיצות בין 0 ל 1 זאת משום ש0 שולח ל 1 אך מהגדרת הפונקציה היא נשלחת למקום הראשון שבו מופיעה הספרה אחת לכן תחזור ל 0 וכך ימשך לאינסוף

.ii

במקרה הכי גרוע נעבור בכל הצמתים (אנחנו בודקים לפני שנכנסים לצומת חדש אם היינו בו לכן לא נהיה ב 2 צמתים) ובכל צומת נבצע (O(n) פעולות בגלל ה for loop לכן (o(n^2)

שאלה 5

ח-1 סעיף א : נשים לב שלכל קריאה רקורסיבית אנו קוראים ל 2 קריאות רקורסיביות על n-1 עומק העץ הוא n בכל שלב אנו מבצעים (1)Oפעולות לכן זמן הריצה הוא (0(2^n)

: 6 שאלה

נשים לב שלולאת ה while מבצעת כמות איטרציות השווה למספר הביטים של b כעט בכל פעם שאנו מכפילים שני מספרים בעלי n ביטים נקבל מספר בעל מספר ביטים קטן שווה ל 2n לכן בכל איטרציה אנו מכפילים מספר בעל n ביטים בעצמו הכפלת מספר זה בעצמו גוררת שנצטרך לבצע 2ⁱ n פעולות כפל

לגבי פעולות חיבור נשים לב כי לכל ביט של a אנו מזיחים שורה לכן נקבל

0 0 0 . 1 0 1 .

מן מקבילית כזו שבה נצטרך לעשות לכל היותר i^2 פעולות חיבור לכן נקבל כי כמות הפעולות שמתבצעות של a לאורך כל הלולאה היא

$$\sum_{i=1}^{m} (2^{i}n)^{2} = n^{2} \sum_{i=1}^{m} 4^{i} = n^{2} * \frac{4^{m} - 1}{3} \le n^{2} 4^{m}$$

בנוסף לכפל בa יש גם כפל ב result המקרה הכי גרוע בו הוא שאנו מבצעים גם אותו בכל פעולה אך נשים לב שאם אנו result מבצעים את הכפל ב a אנו מבצעים את אותם פעולות כפל כמו בכפל בa רק שמתחילים מa במקום מa לכן חסם גם מבצעים את החסם שמצאנו ולכן לסיכום החסם לכל הפעולות יחדיו הוא $O(n^24^m)$