

מטלת מנחה (ממ"ן) 13

מטלה תכנותית

הקורס: 20407 - מבני נתונים ומבוא לאלגוריתמים

חומר הלימוד למטלה: חוברת הקורס וספר הלימוד

משקל המטלה: 5 נקודות

מספר השאלות: 1

מועד אחרון להגשה: 19.12.2022

סמסטר: א2023

אופן הגשת המטלה:

שליחת המטלה תתאפשר רק באמצעות מערכת המטלות המקוונת באתר הבית של הקורס

הסבר מפורט ב"נוהל הגשת מטלות המנחה"

רצוי להתחיל לעבוד על המטלה לפחות שבועיים לפני מועד ההגשה

מותר לעבוד בזוגות, אך שימו לב להנחיות בהמשך

מבוא

בממ"ן זה עליכם לכתוב ולהריץ תכנית ב-Java או ב- $C/C++$ ¹, אשר משווה את מספר ההשוואות של שתי גרסאות של אלגוריתם מיון-מהיר (לא אקראי). לפני שתמשיכו בקריאה, קראו תחילה את סעיף 2.6 בחוברת הקורס (הנחיות לכתיבת ממ"ן תכנותי). ייתכן ותינתנה הנחיות נוספות לקראת מועד ההגשה – אנא בדקו באתר.

¹ למעוניינים להגיש בשפה אחרת (כגון python), אנא בדקו תחילה עם המנחה שבודק/ת את התרגילים שלכם

מטרת פרויקט זה היא להשוות את היעילות של שתי גרסאות של אלגוריתם מיון-מהיר (לא אקראי). נמדוד את היעילות באמצעות מניית מספר ההשוואות שכל אלגוריתם מבצע. שתי הגרסאות הן :

- 1) אלגוריתם מיון-מהיר (לא אקראי) המתואר בספר בעמוד 122
- 2) אלגוריתם מיון-מהיר שהוא שונה מגרסה 1 באופן הבא : במקום להשתמש באבר ציר שממוקם תמיד בקצה הימני של תת-המערך, האלגוריתם ישתמש באבר ציר שאינו המקסימום ואינו המינימום בתת-המערך (באופן היעיל ביותר).
(חלק מהפרויקט הוא להבין כיצד לתת לתיאור זה צורה מדויקת כאלגוריתם וכקוד לכן לא ייענו שאלות בנושא זה)

עליכם לתאר בטבלה את ההשוואה בין שתי הגרסאות של האלגוריתם באופן הבא :

הקלטים לאלגוריתמים יהיו כל $n!$ התמורות של $(1, \dots, n)$ ויש לתאר לכל אחד מהאלגוריתמים את מספר ההשוואות הממוצע של a המקרים הגרועים ביותר (כאשר $0 < a < 1$), מספר ההשוואות הממוצע של a המקרים הטובים ביותר וכן מספר ההשוואות הממוצע עבור כל $n!$ הקלטים האפשריים.

את שתי גרסאות האלגוריתם וכן את כל השגרות בהן השתמשתם לצורך הניתוח יש לצרף כקבצי קוד (בשפה שבחרתם).

הקושי העיקרי בפרויקט ומטרתכם בו היא להפיק את המידע לעיל עבור n גדול ככל הניתן ועבור a גדול ככל הניתן (אך $a \geq 0.5$ לא יהיה מעניין מן הסתם, קיבעו בעצמכם ערכי a מעניינים). שימו לב שכבר $n=15$ מאתגר שכן $15!$ גדול מטריליארד וכנראה שלא תוכלו להחזיק את כל התמורות הללו במערך אחד... תוכלו להעזר בפתרון השאלה האחרונה בממ"ן 12 לצורך כך. (ניתן להחליף את select הדטרמיניסטי ב RandomizedSelect היותר פשוט לכתיבה). כמו-כן תוכלו להעזר באלגוריתם הבא להפקת תמורות :

GenerateNextPermutation(A, n)

$i \leftarrow n - 1$

while $i \geq 1$ and $A[i] > A[i + 1]$

$i--$

If $i = 0$ *return*

$c \leftarrow i + 1$

for $j \leftarrow i + 2$ *to* n

If $A[i] < A[j] < A[c]$

$c \leftarrow j$

Swap(A, i, c)

Reverse($A, i + 1, n$)

כאשר $\text{Reverse}(A, p, r)$ הופכת את תת-המערך בין p ל r כולל. הקלט לאלגוריתם הוא מערך A (שהאינדקס הראשון שלו 1) כך ש $A = [1, \dots, n]$ (אך למעשה יכול להיות תמורה כלשהיא של $(1, \dots, n)$). הריצו את האלגוריתם הזה (לעצמכם) עבור ערכי n קטנים על-מנת לראות מה הוא עושה וכיצד ניתן להשתמש בו בפרוייקט (ניתן להשתמש באלגוריתם הזה כפי שהוא ללא הוכחת נכונות או ניתוח סיבוכיות אך שימו לב שקיימות גרסאות יעילות יותר ממנו כך שייתכן שתמצאו לשפרו).

לבסוף יש לנתח את הטבלה שהתקבלה להשוואת שתי הגרסאות ולהסביר כיצד (אם בכלל) ערכיה מתיישבים עם ניתוח סיבוכיות מספר ההשוואות האסימפטוטי של שתי הגרסאות. (חלק זה כולל הטבלה יצורף כקובץ מוקלד נפרד).