

# מטלת מנחה (ממ"ן) 16

## פרויקט מסכם

הקורס: 20407 – מבני נתונים ומבוא לאלגוריתמים

משקל המטלה: 6 נקודות

מספר השאלות: 1

מועד אחרון להגשה: 30.7.2017

סמסטר: 2017

### קיימות שתי אפשרויות להגשת המטלות:

- שליחת המטלות באמצעות מערכת המטלות המקוונת באתר הבית של הקורס
  - שליחת המטלות באמצעות הדואר או הגשה ישירה למנחה במפגשי ההנחה
- הסבר מפורט ב"נוהל הגשת מטלות המנחה"

בפרויקט זה נתכנן אלגוריתם מקוון (*Online*) מוגבל מקום עבור הבעיה הבאה.

$k$  האיברים הקטנים ביותר

קלט: מערך של מספרים באורך  $n$  ומספר  $k$ .

פלט: רשימה של  $k$  האיברים הקטנים ביותר במערך בסדר עולה.

**אלגוריתמים מקוונים.** המודל של אלגוריתמים מקוונים מדמה מצב בו אין לנו מידע על כל הקלט מראש. לדוגמה, ב"בעיית המועמדים", המוצגת בפרק 5.1 בספר הלימוד על האלגוריתם לקבל בכל רגע נתון החלטה על זהות המועמד הטוב ביותר עד כה. אלגוריתם מקוון מקבל את הקלט בחלקים, ולא כיחידה אחת. בכל שלב האלגוריתם מקבל פיסה חדשה של קלט, ומאבד גישה לקלט מהשלבים הקודמים. אחת הדרכים לדמות אלגוריתם כזה היא לתכנן אלגוריתם העובר על הקלט פעם אחת בדיוק (*One-Pass*). כך, למשל, אלגוריתם *Hire-Assistant* בעמוד 77 בספר מדמה (*Emulates*) אלגוריתם מקוון ל"בעיית המועמדים". דוגמה נוספת לאלגוריתם המדמה אלגוריתם מקוון ראינו בממ"ן 11. בבחינה מחודשת של אלגוריתם majority מהממ"ן תוכלו לראות כי האלגוריתם מבצע מעבר יחיד על הקלט. אותו האלגוריתם בדיוק יבצע את אותה הפעולה לו יקבל את אברי המערך אחד-אחד ממקור חיצוני. מכיוון שאלגוריתם מקוון אינו יודע את "העתיד", בכל שלב בריצת האלגוריתם הוא אמור להיות מסוגל לתת תשובה על מה שראה עד כה. אלגוריתם *Hire-Assistant* שבעמוד 77 מחזיק בכל רגע נתון במשתנה *best* את המועמד הטוב ביותר שבחן עד כה.

**אלגוריתמים מוגבלי מקום.** האלגוריתם הבא הוא אלגוריתם פשוט עבור בעיית  $k$  האיברים הקטנים ביותר. בכל שלב נשמור את כל האיברים שראינו עד עכשיו, וברגע שנתבקש להחזיר

תשובה, נמיין את  $k$  האיברים הקטנים ביותר מתוך המאגר שברשותנו. אלגוריתם כזה מאד בזבזני במקום. אם  $k$  מאד קטן ו  $n$  מאד גדול, לדוגמה, סיבוכיות המקום עשויה להיות  $\Omega(n)$ , בשעה שבכל רגע נתון "יעניינו" אותנו רק  $k$  איברים. על כן נהיה מעוניינים להגביל את המקום שבו ישתמש האלגוריתם.

**תכנון האלגוריתם.** האלגוריתם שנתכנן מקבל כקלט מצביע למערך  $A[1, \dots, n]$ , מספר  $k < n$ ,

ושלוש נקודות בדיקה  $n_1 < n_2 < n_3 < n$ .

על האלגוריתם לעבור על המערך במעבר אחד ולהדפיס את  $k$  האיברים הקטנים ביותר שבחן עד לנקודות הבדיקה  $n_1, n_2, n_3$  במערך.

בכל רגע נתון האלגוריתם שומר רק את  $k$  האיברים הקטנים ביותר (לכל היותר) שבחן עד כה. כלומר, סיבוכיות המקום של האלגוריתם היא  $\Theta(k)$ .

לשם כך עליכם לתכנן מבנה נתונים המוגבל ל  $k$  איברים, ותומך בפעולות הבאות.

$insert(x)$  – הכנסת מפתח  $x$ . סיבוכיות זמן הריצה  $\Theta(\log k)$ .

1. אם המבנה מכיל פחות מ  $k$  איברים, האלגוריתם יכניס את  $x$ .

2. אם המבנה מכיל  $k$  איברים, האלגוריתם יכניס את  $x$  רק במקרה בו  $x$  קטן מהאיבר

המקסימלי במבנה. במקרה כזה, האלגוריתם יוציא מהמבנה את האיבר המקסימלי.

$printkMin()$  – הדפסת  $k$  האיברים הקטנים ביותר שנבחנו עד כה. סיבוכיות זמן הריצה  $\Theta(k)$ .

ניתן להשתמש בתור קדימויות (עמוד 116 בספר) בתור שכבת בסיס למבנה הנתונים שלכם.

הריצו את האלגוריתם שתכנתם על שלושה מערכים  $A, B, C$  באורכים 200,400,800 בהתאמה.

מלאו את המערכים בעזרת פונקצית ספרייה המיועדת ליצירת מספרים אקראיים; כל מערך יכול

איברים בתחום 0..1023.

עבור כל מערך כקלט, הפעילו את האלגוריתם שכתבתם עם הערכים  $k = 10, 50, 100$  ועם

נקודות בדיקה  $n_1 = \frac{n}{4}, n_2 = \frac{n}{2}, n_3 = \frac{3n}{4}$ .