

## מטלה – מיזוג הצעות תקציב

יש לענות על שאלה אחת לבחירתכם. שאלות המסומנות בכוכבית \* מזכות בניקוד כפול.

### שאלה 1: תיקון טעות בהרצאה

בהרצאה אמרתי, שאלגוריתם החציון המוכלל עם פונקציות עולות ליניאריות הוא יעיל-פארטו. זה לא נכון. לדוגמה, נניח שיש שני אזרחים ושלושה נושאים, התקציב הכולל הוא 100, וההצבעות הן (0, 20, 80) ו-(20, 0, 80).

א. מהו התקציב המתקבל מהפעלת האלגוריתם החציון המוכלל עם פונקציות עולות ליניאריות? פרטו את שלבי החישוב.

ב. הראו שהתקציב המתקבל אינו יעיל פארטו (הראו שיפור פארטו שלו).

### שאלה 2: אלגוריתם חלופי

נתון אלגוריתם חלופי לקביעת תקציב.

1. עבור כל נושא  $j$ , מצא את החציון  $x_j$ .

2. סמן ב- $x$  את הסכום של כל ה- $x_j$ .

3. התקציב לכל נושא  $j$  יהיה:

$$x_j + (C-x)/m$$

כאשר  $C$  הוא הסכום הכולל בקופה, ו- $m$  הוא מספר הנושאים. שימו לב שסכום התקציבים של כל הנושאים הוא בדיוק  $C$  (הנירמול מתבצע ע"י הורדת סכום קבוע מכל נושא).

האם האלגוריתם החלופי מגלה-אמת? אם כן – הוכיחו. אם לא – הראו דוגמה נגדית.

### \* שאלה 3: הגינות לקבוצות – תנאי משופר

שאלה זו מתייחסת לאלגוריתם החציון המוכלל עם פונקציות עולות ליניאריות. הוכחנו בהרצאה, שכאשר האזרחים מתחלקים לקבוצות "ממוקדות", כך שהאזרחים בכל קבוצה  $j$  נותנים 100% מהתקציב לנושא  $j$ , התקציב המתקבל מתחלק בין הנושאים ביחס ישר למספר התומכים של כל נושא: נושא שיש לו  $k$  תומכים יקבל  $k/n$  מהתקציב.

עכשיו נניח שיש רק קבוצה אחת ממוקדת: יש קבוצה של  $k$  אזרחים, שנותנים 100% מהתקציב לנושא  $j$ . שאר האזרחים יכולים לחלק את התקציב באופן כלשהו – לא דווקא באופן ממוקד.

האם האלגוריתם מבטיח, שנושא  $j$  יקבל לפחות  $k/n$  מהתקציב? אם כן – הוכיחו. אם לא – הראו דוגמה נגדית.

(מומלץ לבדוק קודם את המקרה  $k=1$ : נניח שיש אזרח אחד הנותן 100% מהתקציב לנושא  $j$ . האם מובטח שנושא  $j$  יקבל לפחות 1 חלקי  $n$  מהתקציב?)

#### שאלה 4: תיכנות: חישוב תקציב

כתבו פונקציה בפייטון, המקבלת כקלט את כמות הכסף בקופה והצבעות האזרחים, ומחשבת את התקציב בעזרת אלגוריתם החציון המוכלל עם פונקציות עולות ליניאריות. כותרת הפונקציה:

```
def compute_budget(  
    total_budget:float,  
    citizen_votes:List[List]  
) → List[float]
```

הנה דוגמת קריאה לפונקציה עבור תקציב עם שלושה סעיפים, ושני אזרחים:

```
compute_budget(100, [ [100, 0, 0], [0, 0, 100] ])
```

הערך המוחזר הוא רשימה עם מספרים כמספר הסעיפים, למשל:

```
[50, 0, 50]
```

#### שאלה 5: תיכנות: בדיקת הגינות לקבוצות

נתונה פונקציה בשם `compute_budget`, עם חתימה כמו בשאלה 4. בשאלה זו עליכם לבדוק אם הפונקציה מחזירה תקציב הוגן לקבוצות. כתבו קוד הבודק את ההגינות של הפונקציה `compute_budget` עבור תקציב עם 3 נושאים ומדינה עם 10 אזרחים (השתמשו בהגדרת "הגינות לקבוצות" שלמדנו בהרצאה).