

מטלה - חלוקת חפצים בדידים

יש לענות על שאלה אחת לבחירתכם.

שאלה 2: אלגוריתם מעגלי הקנאה - סדר החפצים

כזכור, אלגוריתם מעגלי-הקנאה מתחיל בכך שהוא מסדר את החפצים בסדר שרירותי כלשהו. כלומר, בכל סדר שבו נסדר את החפצים, אנחנו עשויים לקבל תוצאה שונה.

א. הראו דוגמה עם 3 אנשים ו-3 חפצים, שבה כל סידור של החפצים נותן תוצאה אחרת (סה"כ 6 תוצאות).

ב. הראו דוגמה עם 3 אנשים ו-3 חפצים, שבה כל סידור של החפצים נותן אותה תוצאה.

ג. הראו דוגמה עם 3 אנשים ו-3 חפצים, שבה מספר התוצאות האפשריות גדול מ-1 וקטן מ-6.

שאלה 3: מיקסום מכפלת הערכים - חפצים רציפים

כזכור, למדנו שאלגוריתם מיקסום-מכפלת-הערכים נותן חלוקה שהיא ללא קנאה כאשר החפצים הם רציפים (ניתנים לחיתוך). תנו דוגמאות לחלוקה של 3 חפצים רציפים בין 2 אנשים, שבהן:

א. אלגוריתם מיקסום-מכפלת-הערכים לא צריך לחתוך אף חפץ.

ב. אלגוריתם מיקסום-מכפלת-הערכים צריך לחתוך חפץ אחד בדיוק.

ג. אלגוריתם מיקסום-מכפלת-הערכים צריך לחתוך שני חפצים בדיוק.

ד. אלגוריתם מיקסום-מכפלת-הערכים צריך לחתוך שלושה חפצים בדיוק.

שאלה 4: מיקסום מכפלת הערכים - חפצים בדידים

כזכור, למדנו שאלגוריתם מיקסום-מכפלת-הערכים נותן חלוקה שהיא ללא-קנאה-מלבד-1 כאשר החפצים הם בדידים (לא ניתנים לחיתוך). תנו דוגמאות לחלוקה של 4 חפצים בדידים בין 2 אנשים, שבהן:

א. אלגוריתם מיקסום-מכפלת-הערכים מחזיר חלוקה שהיא ממש ללא קנאה.

ב. אלגוריתם מיקסום-מכפלת-הערכים נותן חפץ 1 לאדם אחד ו-3 חפצים לשני.

ג. אלגוריתם מיקסום-מכפלת-הערכים נותן נותן 2 חפצים לכל אחד.

שאלה 2: הגינות, יעילות, ומיקסום המכפלה

בשיעור הראינו, שכל חלוקה הממקסמת את מכפלת הערכים, היא יעילה-פארטו וללא-קנאה (עד כדי חפץ אחד).

הראו שהכיוון ההפוך לא בהכרח נכון: הראו דוגמה לחלוקה ללא-קנאה ויעילה-פארטו, שאינה ממקסמת את מכפלת הערכים.

שאלה 5: תיכנות - חלוקה ללא-קנאה-בקירוב

נתונה המחלקה הבאה:

```
class Agent:
```

```
def item_value(item_index:int)->float: ...
```

המחלקה מייצגת שחקן המשתתף במשחק חלוקה הוגנת. יש בה פונקציה אחת המתארת את הערך שהשחקן מייחס לחפץ שהאינדקס שלו הוא `item_index`.

כיתבו פונקציה הבודקת האם חלוקת-חפצים נתונה כלשהי היא EF1. כותרת הפונקציה:

```
def is_EF1(agents:List[Agent], bundles:List[int])->bool: ...
```

הפרמטר `agents` הוא מערך בגודל `n` המייצג את השחקנים.

הפרמטר `bundles` הוא מערך באותו גודל - `n` - המייצג את החלוקה: `bundle[i]` הוא אוסף אינדקסי החפצים שמקבל שחקן `i`.

שאלה 6: תיכנות מתקדם - מיקסום המכפלה עם מספרים שלמים

תיזכורת: בבעיית החלוקה שלמדנו בשיעור האחרון, יש `n` אנשים ו-`m` חפצים בדידים, וכל אדם מייחס ערך לכל חפץ. ניתן לייצג כל בעיה כזאת כמטריצה עם `n` שורות ו-`m` עמודות, שבה הערך בשורה `i` ועמודה `j` הוא הערך שמייחס אדם `i` למשאב `j`.

כיתבו פונקציה בפייתון, בעזרת `cvxpy`, המקבלת כקלט מטריצה כזאת, וכותבת למסך תיאור של חלוקה הממקסמת את מכפלת הערכים. הניחו שכל הערכים הם מספרים שלמים בין 1 ל-10.

לדוגמה, אם הקלט הוא:

```
[[1, 4, 9],  
 [9, 5, 1]]
```

אז הפלט הוא:

Agent #1 gets item #3,
Agent #2 gets items #1, #2.

שאלה 7: כמה חלוקות יעילות-פארטו יש?

תארו בעיית חלוקה של חמישה חפצים בין שני שחקנים, שבה מספר החלוקות היעילות-פארטו ללא שיתופים הוא:

א. בדיוק 6; ב. בדיוק 7; ג. בדיוק 10; ד. בדיוק 17; ה. בדיוק 32.

(למה דווקא מספרים אלה?)