בס"ד

מטלה 3

מחקר: [קישור]

קורס: תכנות אלגוריתמים מחקריים

מגיש: כפיר גולדפרב

<u>אלגוריתם חמדן (Greedy Algorithm - *CGA*):</u>

הסבר מקוצר:

בהינתן n מספרים שלמים חיוביים ו-k מספר הקבוצות שנרצה לחלק כך שסכום הקבוצות קרוב (שואף להיות שווה), תהליך האלגוריתם:

1. ממיין את המספרים בסדר יורד.
2. לכל מספר מקצה קבוצה עם סכום הכי קטן עד כה.

קלט 1:

$$n = \{12,9,8,4,2,3,5,1,6\}, k = 2$$

תהליך 1:

לאחר מיון נקבל: $\{1,2,3,4,5,6,8,9,12\}$, ניקח לכל קבוצה בתורה את המספר המינימלי בקבוצה זו.

פלט 1:

$$\{1,3,5,8,12\}, \{2,4,6,9\}$$

סכומי הקבוצות הם: $21, 29$

קלט 2:

$$n = \{12,9,8,4,2,3,5,1,6\}, k = 3$$

תהליך 2:

לאחר מיון נקבל: $\{1,2,3,4,5,6,8,9,12\}$, ניקח לכל קבוצה בתורה את המספר המינימלי בקבוצה זו.

פלט 2:

$$\{1,4,8\}, \{2,5,9\}, \{3,6,12\}$$

סכומי הקבוצות הם: 13, 16, 21

<u>אלגוריתם חמדן שלם (Complete Greedy Algorithm - *CGA*):</u>

הסבר מקוצר:

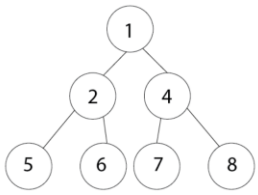
1. ממיין את המספרים בסדר יורד.
2. ולאחר מכן מחפש בעץ כאשר כל רמה בעץ (level) מיועד למספר אחר, וכל זרוע מיועדת לקבוצה אחרת.
3. כדי להימנע מכפילות ניקח את הפרמוטציות של כל תת-קבוצה וניקח את ההפרש ביניהם.
4. נמשיך לעקוב אחר הקבוצה עם הסכום הכי גבוה עם הפתרון הכי טוב.
   a. אם סכום הקבוצות חרג או שווה לגבול שהצבנו לו, האלגוריתם נכשל.
5. אם הגענו לחלוקה מושלמת או שהסכום של הקבוצה הכי גדולה שווה למספר הכי גדול, נחזיר את התוצאה כמו שהיא בלי חלוקה מושלמת

קלט:

$$n = \{5,4,6,8,7,1,2\}, k = 2$$

תהליך:

לאחר מיון: $\{1,2,4,5,6,7,8\}$, העץ יראה פחות או יותר כך:



כאשר כל זרוע (ימין ושמאל) מייצגת קבוצה אחרת, נשים לב שצד ימין סכומו 19 וצד שמאל סכומו 13 מכיוון שסכום הקבוצה השמאלית קטנה מימין, נוסף את הקודקוד (1) לקבוצה שמאל וקיבלנו את הסכומים 14,19.

פלט:

$$\{1,2,5,6\}, \{4,7,8\}$$

<u>אלגוריתם (Karmarkar-Karp Heuristic – *KK*) [מיועד לחלוקה של שני קבוצות בלבד]:</u>

הסבר מקוצר:

1. ממיין את המספרים בסדר יורד.
2. לאחר מכן מחליף בין שני המספרים הכי גדולים בקבוצה בסדר ממוין בין ההפרש שלהם.
3. לאחר מכן נפריד בניהם לשני קבוצות שונות (שני המספרים הכי גדולים).
4. נמשך את התהליך עד שנגיע למספר אחיד שהוא ההפרש הסופי בין שני סכומי התתי-קבוצות.
5. נוסיף את ההפרש הנותר לשני סכומי תתי-הקבוצות ואז נחלק ב-2.

קלט:

$$n = \{8,6,5,2,1\}, k = 2$$

תהליך:

לאחר מיון: $\{1,2,5,6,8\}$,

ניקח את שני המספרים הכי גדולים 8 ו-6 ונפריד אותם לשני קבוצות שונות $a=\{8\}, b=\{6\}$, ונחליף אותם בקבוצה n עם ההפרש שלהם $|8-6|=2$, ולכן $n=\{1,2,2,5\}$,

צעד שני: $a=\{8,5\}, b=\{6,2\}, n=\{1,2,3\}$

צעד שלישי: $a=\{8,5,3\}, b=\{6,2,2\}, n=\{1,1\}$

צעד רביעי: $a=\{8,5,3,1\}, b=\{6,2,2,1\}, n=\{1\}$

נקבל ש: $|a|=8+5+3+1+1=17$, $|b|=6+2+2+1+1=12$ נחלק את הסכומים ב-2.

פלט:

$$|a|=8.5, |b|=6$$

- הערה: מכיוון שקיבלנו סטייה של 0.5 ומדובר במספרים שלמים אז ניתן להתעלם מהחצי או להוסיף עוד חצי כדי להגיע למספר שלם למעלה ולכן נקבל 2 פתרונות אפשריים:

  $|a|=8, |b|=6$ או $|a|=9, |b|=6$

<u>אלגוריתם (*Complete* Karmarkar-Karp Heuristic – *CKK*):</u>

הסבר מקוצר:

אלגוריתם זה עובד כמו האלגוריתם הקודם רק עבור יותר מ-*2* קבוצות ולכן מיותר להראות את תהליך העבודה שלו, כיוון שהוא מביא את אותם תוצאות עבור 2 קבוצות ואותו אלגוריתם פועל על 3 קבוצות ומעלה גם באותה דרך רק כללית יותר.

<u>אלגוריתם (Horowitz and Sahni - *HS*):</u>

הסבר מקוצר:

1. ראשית מחלק את n המספרים לשתי חצאי קבוצות של גודל $\frac{n}{2}$.
2. מג'נרט את כל האפשרויות של תתי הקבוצות הללו (כולל הסט הריק).
3. ממיין אותן בסדר עולה לפי סכומיהם.
4. כל תת-קבוצה של המספרים המקוריים חייבת להיות מורכבת מתת-קבוצת מהמחצית הראשונה בתוספת קבוצת משנה מהמחצית השנייה.
5. כעת האלגוריתם מכיל שני מצביעים, אחד לכל רשימה ממוינת של סכומי תת-הקבוצות, המצביע הראשון מתחיל בתת-קבוצה הראשונה עם הסכום הקטן ביותר והמצביע השני מתחיל בתת-קבוצה השנייה עם הסכום הגדול ביותר.
   a. כעת נחבר בין שני הסכומים של שני המצביעים, אם הסכום שווה או גדול מהערך החיפוש מסתיים.
      i. אם הסכום הוא קטן מהערך, המצביע הראשון סכומו גדול מהערך.
      ii. אם הסכום גדול מהמטרה, המצביע השני מוקטן במיקום אחד.
   b. האלגוריתם מגדיל את המצביע הראשון ומקטין את השני עד שערך מטרה שלנו נמצא.

בס"ד

קלט:

$$\{8,5,-2,-3,-7\}$$

פלט:

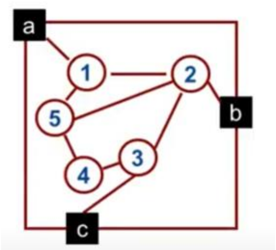
$$\{5,-3,-2\},\{8,-7\}$$

<u>אלגוריתם (*RNP* - Recursive Number Partitioning):</u>

הסבר מקוצר:

1. מפעיל את היוריסטיקה של *KK* כדי ליצור פתרון ראשוני.
   a. אם *k* הוא אי-זוגי, אז ניצור את כל תת-הקבוצות הראשונות שיכולות להיות חלק מ-*k* קבוצות, חלוקה זו טובה יותר מהטובה הנוכחית. עבור כל אחד, זה בצורה אופטימלית מחלק את המספרים הנותרים *k − 1* בדרכים.
   b. אם *k* זוגי, זה מייצר את כל המחיצות הדו-כיווניות האפשריות של המספרים, כך שחלוקה של כל תת-קבוצה ב-*k/2* דרכים עשויה להיגרם בחלוקת *k* טובה יותר מהטובה הנוכחית.
2. לאחר מכן הוא מחלק באופן אופטימלי את תת-הקבוצה הראשונה ב-$\frac{k}{2}$ דרכים, ואם התוצאה היא סכום מקסימלי של תת-קבוצה שקטן מזה של הטוב ביותר הנוכחי, כך שהוא מחלק בצורה אופטימלית את תת-הקבוצה השנייה *k/2* דרכים.
3. כדי ליצור את תת-קבוצות הראשונות עבור *k* אי זוגי, או לחלק את מספרים בשתי דרכים עבור *k* זוגי, *RNP* ממיין את המספרים בסדר יורד, ולאחר מכן מחפש עץ דו-נארי של הכללה-אי-הכללה. כל רמה בעץ מתאימה לרמה אחרת מספר, ובכל צומת סניף אחד כולל את המספר בקבוצת המשנה, והענף השני אינו כולל אותה. גיזום מתבצע כדי לחסל תת-קבוצות שהסכום שלהן ייפול מחוץ להן גבולות מוגדרים, שיידונו להלן. עבור חלוקה דו-כיוונית אופטימלית, *RNP* משתמש באלגוריתם *CKK*.

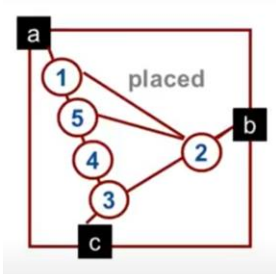
קלט:



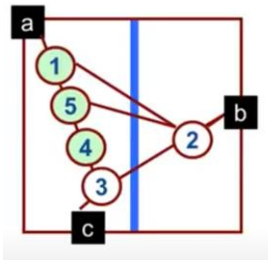
נשים לב כי יש $n = 5$ מספרים ו-$k = 3$ לחלוקה לקבוצות.

תהליך:

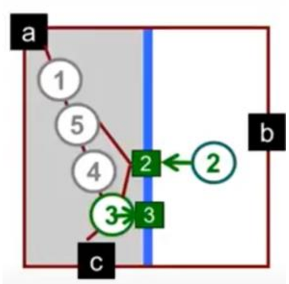
ראשית למען הנוחות נסדר את הצורה שבה הקודקודים מרווחים לצורה מאוזנת יותר, כלומר כל קודקוד יהיה ליד השער (a, b, c) שהוא אמור להיות (מבחינת מרחק), ונקבל:



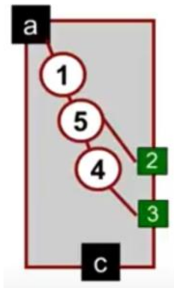
כעת נחלק את הגרף הנתון לשני חציים, נזכור שלפי החלוקה שלנו וסכום המספרים, לא הגיוני שבצד אחד יהיה 4 מספרים ובאחר יהיה רק 1, נדרש שיהיה בערך חלוקה כך שבכל צד יהיה 3 או 2 מספרים, ולכן ניקח את שלושת המספרים העליונים בקבוצה הגדולה (מסומנים בירוק):



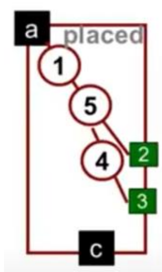
מכיוון ש-3 ו-2 הכי קרובים ל-b ו-c, נשאיר אותם איפה שהם ונרצה רק לעדכן מחדש את המיקום של 1,5 ו-4,



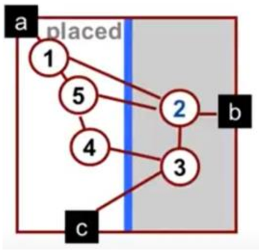
כעת נחשיב את 2 ו-3 שנמצאים על הקו הכחול כשערים בפני עצמם (כדי לקבע את 2 ו-3), ונקבל:



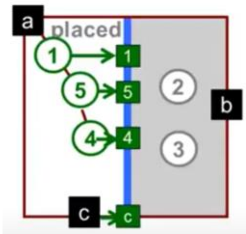
נסדר בצורה מאוזנת יותר (כל אחד יותר קרוב לשער שלו):



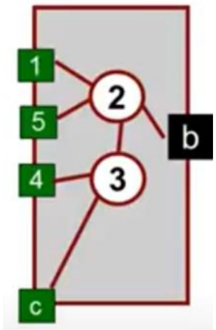
כעת נסדר את כל הגרף מחדש בצורה מאוזנת:



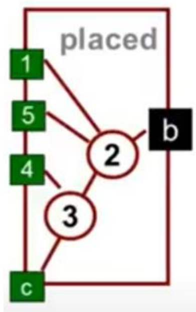
מכיוון ש-3 הוא זה שהיה הכי קרוב לשער c, שער c כרגע מחובר בצורה כללית למרכז (חיתוך עם צד ימין ששם נמצאים 2 ו-3), בנוסף נגדיר את 1, 5 ו-4 כשערים בפני עצמם המחוברים למרכז:



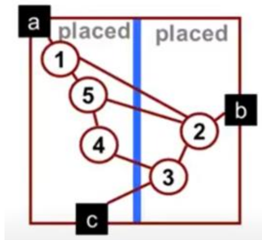
נסדר שוב את צד ימין:



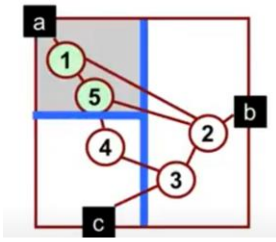
בצורה יותר מאוזנת:



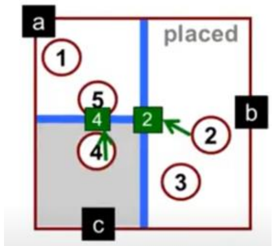
כעת נקבל:



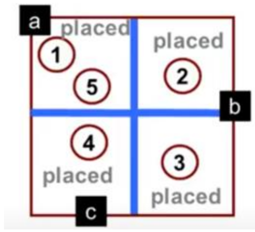
מבחינת סכום, 1 ו-5 קרוב כמעט ל-4 בצורה יחסית ולכן ננסה לחות בניהם קו כי אנחנו מנסים לחלק את הגרף לשלושה קבוצות (a, b, c) ונקבל:



נחזור על התהליך הקודם:



וכעת ניתן לראות שהגענו לחילוק מאוד טוב בצורה יחסית לבעיה זו:



כעת נמשיך את האלגוריתם בצורה רקורסיבית...