יונתן חאיק

תרגיל מס' 1 - אלגוריתמים

1) א) שיקול חמדני - נבחר את ~~המוצר~~ הדוגמא עבורה הפתרון האופטימלי ~~אשא~~ בשברים:

$w_1 =$ $\qquad$ $W=10$.

	x_1	x_2	x_3
W_i	2	8	9
C_i	3.96	15.92	18
V_i	1.92	1.99	2

אם ניקח את הדוגמא הנ"ל עם פתרון בעיית התרמיל בשברים, ע"י ~~הפתר~~ הוקטור יתקבל $\langle 0,0,1\rangle = 18$. אך הפתרון הנכון יהיה הוקטור $\langle 1,1,0\rangle = 19.88$

<u>סתירה</u>

ב) לא נכון. הדוגמא שהבאנו לעיל סותרת גם טענה זו.

2) ~~הפת~~ השיקול החמדני - להתחיל להחזיר את העודף במטבע הגדול ביותר c^i, כאשר $i=max$, ואיננו חורגים מ-A העודף הכולל לתת ללקוח. במידה שעלולה להיות חריגה, נעבור למטבע c^{i-1} עד למציאת A.

```
GetChange(A,k,C)
    currentChange=0
    Use Bucket-Sort on the coins by k            // Θ n
    While(currentChange < A) do                  // Θ n
        if (currentchange + c^k ≤ A)             // Θ 1
            currentChange = currentChange + c^k  // Θ 1
            Continue                             // Θ 1
        else k--                                 // Θ 1
```

הוכחת אופטימליות: I תת בעיה אופטימלית:

<u>טענה</u>: יהי $R=<r_1...r_n>$ פתרון אופטימלי המקיים את הבעיה הראשונה.

נגדיר תת-בעיה: $r_1...r_n$ כך ש $A'=A-C^k$

אזי $R'=<r_2..r_n>$ אופטימלי עבור תת-הבעיה.

הוכחה בשלילה: נניח ש-R' אינו אופטימלי עבור תת הבעיה, אזי קיים פתרון $F'=<f_2...f_n>$ כך ש $|F'|<|R'|$. אופטימלי.

נגדיר $F=<f_1...f_n>$ ~~אז נקבל את הפתרון שהגדרנו~~.

אם $|F|<|R|$ אזי R אינו אופטימלי בסתירה להנחה.

אם $|F|=|R|$ אזי $|F'|=|R'|$ כלומר R' אופטימלי.

~~תכונת~~ <u>הבחירה אופטימלית</u>

נניח שהמטבעות ממוינים לפי הערך הגדול תחילה.

בחירה חמדנית ראשונה:

<u>טענה</u> - קיים וקטור אופטימלי $R=<r_1...r_n>$

נבחר עבור כל מטבע ב-R מהוקטור ונחליף אותו בגדול.

אם החלפנו במטבע בעל ערך גבוה יותר - הערך הכולל יהיה גדול יותר מ-A.

אם החלפנו באותו ערך - נשארנו עם אותו וקטור

אם הכנסנו ערך נמוך יותר - עלינו להוסיף מטבעות על מנת להגיע לערך שהוגדר ולכן הפתרון כבר אינו אופטימלי.

(ב) דוגמה נגדית: $A=10$ וקיימים מטבעות $8,5,\frac{1}{2}$.

עפ"י האלגוריתם לעיל נבחר את הוקטור $<1,0,4>$ סה"כ 5 מטבעות

אבל הפתרון האופטימלי יהיה דווקא $<0,2,0>$ סה"כ 2 מטבעות

ולכן הפתרון שהצענו אינו מתאים לכל סוג מטבעות כלשהם.

שאלה 3

ניסיתי במחשבה

זה לא הספיק...

א-ב . נגדיר את הבתים להיות מסודרים במערך H כאשר כל אינדקס h_i יציין את מרחק הבית מתחילת הדרך. כמו כן, נגדיר מערך P שיציין את מיקום העמודים אותם שמנו.
הפיתרון החמדני – מציאת הבית הראשון והצבת העמוד במרחק X מאותו בית. וחיפוש הבית הראשון מאותה נקודה.

```
FindMinPoles(P,H,X,length)
  I<-0                                              \\ O(1)
  Current = 0                                       \\ O (1)
  While(current<length)                             \\ O (n)
        First = first house between current and length   \\ O (n)
        If (first+x =< length)                      \\ O (n)
                P[i]= first+x                       \\ O (1)
                current= current+2X                 \\ O (1)
                I++                                 \\ O (1)
                Continue                            \\ O (1)
        Else                                        \\ O (1)
                P[i] = length                       \\ O (1)
                Exit                                \\ O (1)
  Return P                                          \\ O (1)
```

סה"כ זמן ריצה O(n). (אם יש צורך במיון מיקומי הבתים, אזי זמן הריצה יהיה O(nLog(n)))

ג. הוכחת נכונות האלגוריתם:

a. הוכחת הפיתרון החמדני:

טענה: הוקטור P=<p1..pn> מכיל בתוכו את הבחירה האופטימלית על פי הפתרון החמדני שהצענו.

הוכחה (בשלילה): נניח כי קיים וקטור אחר Q = <q1..qn>, אשר מקיים |Q|<|P| .

במקרה כזה לפחות אחד מהאיברים בווקטור P לא מופיע בווקטור Q.
אזי, לפחות אחד מהבתים נמצא כעת ללא כיסוי.
על מנת לכסות את הבתים הללו נידרש לפחות לעמוד אחד חדש.
אם עמוד אחד יספק את כל הדרוש לנו – נשארנו עם אותו גודל של וקטור.
אם נכניס יותר מעמוד אחד – אזי, |Q|>|P|, מה שעומד בסתירה להנחה. מש"ל

b. הוכחת תת הפיתרון האופטימלי:
לאחר ביצוע האיטרציה הראשונה, נשאר כעת עם תת בעיה דומה.
נגדיר את P' = <p2…pn> ואת H' = <h?...hn> להיות המערך המכיל את הבתים בניכוי אלו (אחד או יותר) שמקושרים לעמוד p1.
טענה: עבור תת בעיה זו, הפתרון שהצענו עדיין אופטימלי
הוכחה (בשלילה): נניח כי יש פתרון אחר לתת הבעיה H'. שהינו אופטימלי ואינו הוקטור P'
ניתן לראות באופן דומה שכל הוצאה של אחד מהעמודים תגרור הכנסה של לפחות עמוד אחד חדש – הפתרון החדש אינו אופטימלי
מש"ל