בעיות תזמון עם דד-ליין

קלט: נתונות n משימות $\{1,2 \dots, n\}$ כל המשימות מתבצעות ביחידת זמן אחת.

לכל משימה i נתון רווח $g_i > 0$, ודדליין $d_i > 0$.

רק אם המשימה מבוצעת עד הדדליין הרווח מתקבל, כלומר לא חייבים לבצע את כל המשימות.

פתרון חוקי- תזמון שלאו דווקא מכיל את כל המשימות אך כל משימה שנמצאת בו מתבצעת עד הדדליין.

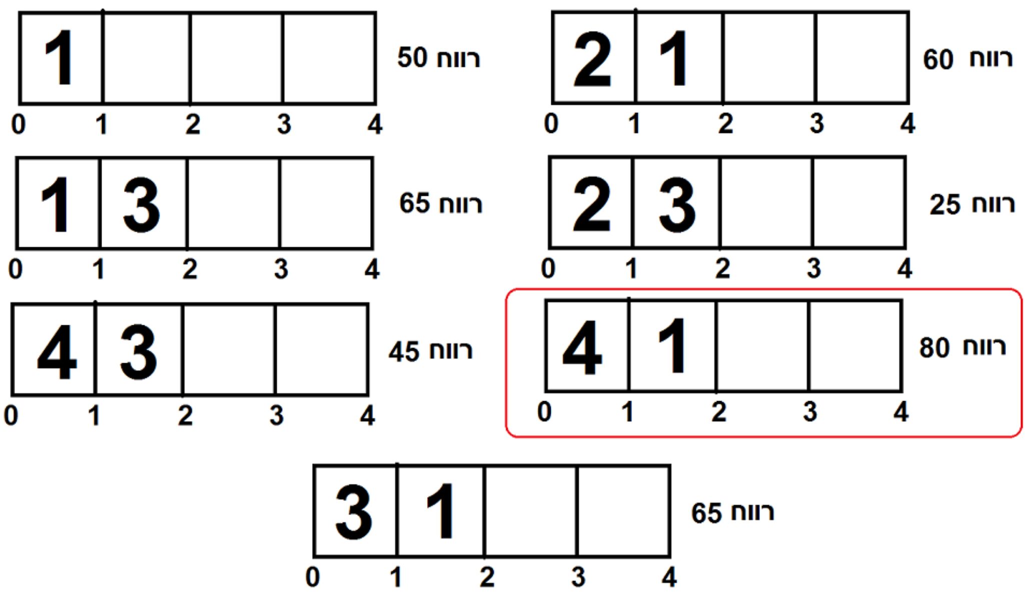
ערך של פתרון- סכום רווחי המשימות שבתזמון

פלט: פתרון חוקי בעל ערך מקסימלי.

דוגמה

i	1	2	3	4
g_i	50	10	15	30
d_i	2	1	2	1

תזמונים אפשריים:



הגדרה

קבוצת משימות $J = \{1, .. k\}$ תקרא פיזיבלית.
אם קיים סידור של המשימות כך שכולן מתבצעות עד הדדליין שלהן.

פיזיבלי – $\{4,1\}$

פיזיבלי – $\{3,1\}$

$\{1,2\}$ – פיזיבלי
$\{2,4\}$ – לא פיזיבלית
$\{1,2,3\}$ – לא פיזיבילית

<u>האלגוריתם</u>

1. נמיין את המשימות לפי רווח בסדר יורד.
2. מתחילים מפתרון ריק.
3. נעבור על המשימות לפי הסדר מסעיף 1:
אם הוספת המשימה משאירה את הקבוצה פיזיבילית נוסיף, אחרת נעבור למשימה הבאה.

<u>משפט</u>
האלגוריתם מחזיר פתרון אופטימלי – ללא הוכחה .
השאלה איך בודקים האם קבוצה היא פיזיבילית?

<u>שיטה ראשונה</u>

<u>למה 1 - lemma (לחצו CTRL ועל המילה לפירוש)</u>

תהי J קבוצת משימות $\{1,\dots k\}$ ונניח ש $d_1 \le d_2 \le \cdots \le d_k$

J פיזיבילית אם ורק אם התזמון $1,2,\dots,k$ ניתן לביצוע.

<u>הוכחה</u>
→ טריוויאלי נובע מההגדרה של פיזיביליות.

← תהי J קבוצה פיזיבילית ונניח בשלילה שהתזמון $1 \dots k$ לא ניתן לביצוע.

כלומר קיימת משימה r כך ש $d_r < r$

← $d_r \le r-1$ מההנחה $d_1 \le d_2 \dots \le d_r \le r-1$

כלומר יש r משימות שצריך לשבץ עד הזמן $r-1$.
דבר זה לא אפשרי, בסתירה לפיזיביליות של J.

ולכן התזמון $1 \dots k$ ניתן לביצוע, כנדרש.

<u>מימוש האלגוריתם</u>
נמיין לפי רווחיות יורדת .

נסמן את המשימות $\{1 \dots n\}$ ונניח ש 1 הכי רווחית ו n הכי פחות.

נשתמש בשני מערכים $d[0 \dots\dots n]$ ו $j[0 \dots . n]$.

בשניהם התא 0 משמש לביקורת.

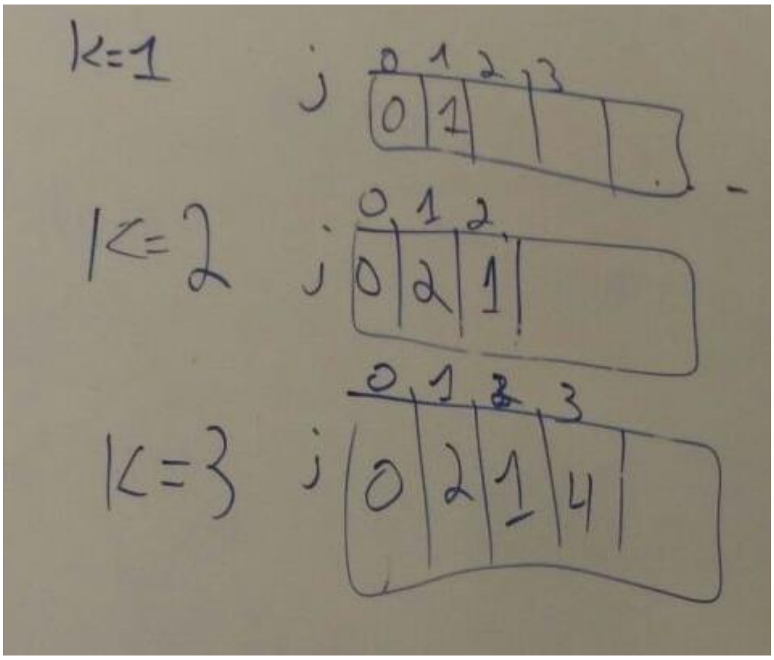
במערך d נחזיק את הדדליין.

ובמערך j נבנה תוך כדי האלג' את התזמון.

```
sequence1(d[0...n],j[0...n]
{
    d[0] <- j[0]<-0
    k <- j[1] <- 1
    for i <- 2...n
    {
        r <- k
        while(d[j[r]] > max(d[i],r))
            r <- r-1
        if(d[i] > r)
        {
            for(m=k ; m>=r+1 ; m--)
                j[m+1] <- j[m]
            j[r+1] <- i
            k <- k+1
        }
    }
    return k,j[1....k]
}
```

דוגמה

i	1	2	3	4	5	6
g_i	20	15	10	7	5	3
d_i	3	1	1	4	2	3



סיבוכיות

מיון- $O(nlogn)$

עבור החלק השני הקלט הגרוע ביותר הוא כשהדדליינים ממויינים בסדר יורד, ואז נדחוף את כולם ימינה.

יתנהג כמו סדרה חשבונית- $O(n^2)$

סה"כ $\mathbf{O(n^2)}$

שיטה שניה

למה 2

תהי J קבוצה של n משימות.

J פיזיבילית אם ורק אם השיטה הבאה מניבה תזמון שניתן לביצוע:

1. מנתחיל מתזמון ריק j
2. לכל משימה i נשבץ אותה באינדקס $1 \leq t \leq d_i$ המקסימלי הפנוי ב j .

הוכחה

→ טריוויאלי מההגדרה של קבוצה פיזיבילית .

← תהי J קבוצה פיזיבילית, ונניח בשלילה שהשיטה שתוארה לא נותנת תזמון שניתן לביצוע .

כלומר קיימת משימה i שכאשר רוצים לשבץ אותה אין תא פנוי לשבץ .

נסמן ב $r = d_i$.

מכיוון שיש n משימות ו n תאים , חייב להיות תא פנוי.

נסמן ב s את התא הפנוי הראשון. נשים לב ש $s > r$.

מכך שהשיטה משבצת תמיד במקום המקסימלי הפנוי .

נובע שהמשימות ששובצו בתאים $1 \ldots s-1$ בעלות דדליין קטן מ S .

כלומר יש$s-1$ משימות עם דדליין קטן או שווה $s-1$.

וגם הדדליין של i קטן או שווה ל $s-1$.

כלומר יש בJ כמות של S משימות עם דדליין קטן שווה ל $s-1$ בסתירה לכך ש J פיזיבילית.

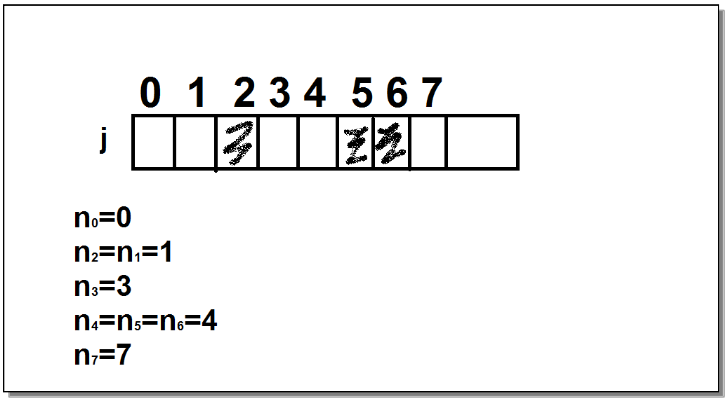
ולכן השיטה מניבה תזמון שניתן לביצוע, כנדרש.

רעיון האלגוריתם

סימונים והגדרות

$$n_t = \max\{\, k: 0 \le k \le t \ , \quad \text{התא ה}k\text{ פנוי} \,\}$$

דוגמה



אם $n_i = n_j$ i ו j באותה הקבוצה, לכל קבוצה k נסמן האיבר המינימלי בקבוצה $F(k)$.

K הוא שם הקבוצה

$$A = \{0\} \quad F(\boldsymbol{A}) = 0$$

$$B = \{1,2\} \quad F(B) = 1$$

$$C = \{3\} \quad F(C) = 3$$

$$D = \{4,5,6\} \quad F(D) = 4$$

$$E = \{7\} \quad F(E) = 7$$

נשתמש במבנה נתונים שמחזיק קבוצות זרות .בשלב הראשון כל איבר נמצא בקבוצה לבד

קימות שתי פעולות:

Find(x)- מקבלת איבר x ומחזירה את התוית של הקבוצה (השם) שהאיבר שייך אליה

Merge(a,b)- מקבלת שתי תויות a,b של קבוצות וממזגת אותו לקבוצה אחת שהתוית שלה היא אחת משתי התויות המקוריות (לא יודעים מי)

ניתן לבצע את פעולת החיפוש בסיבוכיות של $O(\log n)$

ואת פעולת המיזוג בסיבוכיות של$O(1)$

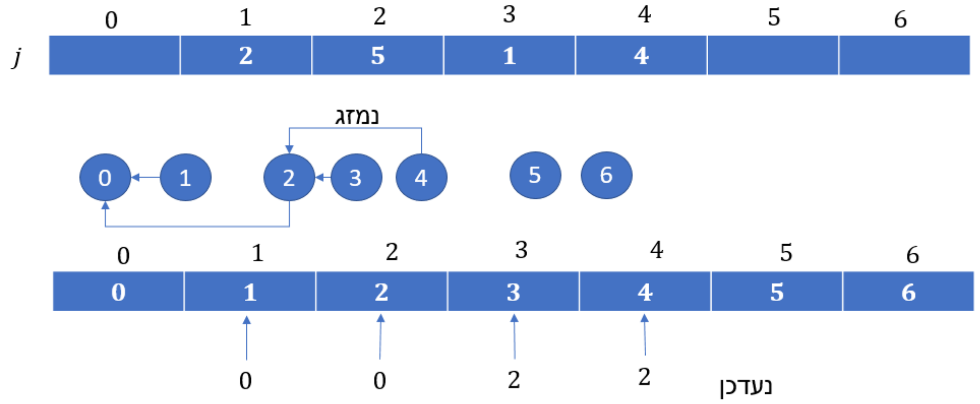
המימוש עצמו צפוי להלמד לקראת סוף הקורס

<u>שלבי האלגוריתם</u>

1. מתחילים מתזמון ריק j וכל אחד מהמיקומים $0 \ldots n$ מאותחל לקבוצה לבד . $F(i) = i$
2. עוברים על המשימות לפי רווחיות יורדת.
   עבור משימה עם דדליין d .
   א. נחפש את הקבוצה ש d שייך אליה ונסמן אותה ב k.
   ב. נסתכל על $F(k)$
   - אם $F(k) = 0$ נמשיך למשימה הבאה.
   - אחרת, נשבץ אותה ב $F(k)$.
   - נחפש את הקבוצה ש $F(k) - 1$ שייך אליה - נסמן אותה ב L .
   - נמזג את L ו K
   - נעדכן את $F(k)$ להיות $F(L)$

<u>דוגמה</u>

i	1	2	3	4	5	6
g_i	20	15	10	7	5	3
d_i	3	1	1	4	2	3



<u>סיבוכיות</u>

מיון לפי רווח -$O(nlogn)$

יש לכל היותר $2n$ *find* שניתן לעשות כל אחת בסיבוכיות$O(\log n)$

כלומר $O(nlog\ n)$

וn פעולות *merge* שניתן לעשות כל אחת בסיבוכיות $O(1)$

כלומר $O(n)$

שאר הקוד - $O(n)$

סה"כ $O(nlogn)$.