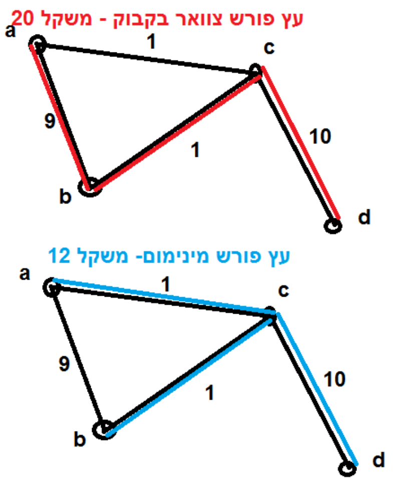
# תכנון אלגוריתמים - 202-1-2041
# בוחן אמצע – 8.05.2015

## שאלה 1 סעיף א




נבחין כי בכל עץ פורש הקשת $(c,d)$ נמצאת, כלומר העץ האדום הוא עץ קשת-כבדה שאיננו עץ פורש מינימום.

## שאלה 1 סעיף ב

יהי $T=(V,E_T)$ עץ פורש של $G$. בלי הגבלת כלליות $u\in S$ ו- $v\in V\backslash S$. מכיוון ש- $T=(V,E_T)$ הוא עץ פורש, אזי יש בו מסלול מ- $u$ ל- $v$. נתחיל ללכת על המסלול מהקודקוד $u$. נביט על הצלע הראשונה $(x,y)$ שהמקור שלה ב- $S$ ויעדה אינו ב- $S$. אם לא קיימת כזו לעולם לא נגיע ל- $v$, שאינו ב- $S$. הראנו שקיימת ב- $T$ קשת $e'=(x,y)$ כך ש- $x\in S$ ו- $y\in V\setminus S$.

## שאלה 1 סעיף ג

יהא $T(V,E_T)$ עץ פורש מינימום של $G$. נטען שהוא עף פורש צוואר בקבוק של $G$. כמובן שהוא עץ פורש. נותר להוכיח שהוא עץ קשת-כבדה. כלומר צריך להוכיח שמשקל הקשת הכבדה בו, נסמנה $e$, קטן או שווה למשקל הקשת הכבדה של העצים הפורשים האחרים של $G$. נניח בשלילה שלא. אזי **קיים** עץ פורש של $G$, נסמנו $T'$, כך שהקשת הכבדה ביותר בו, שנסמנה ב- $e''$, מקיימת $w(e'')<w(e)$.

כעת נעזר בסעיף ב. נביט בגרף $H=(V,E_T\setminus\{e\})$. גרף זה מכיל שני רכיבי קשירות זרים. נסמן את קבוצות הקודקודים שלהם כ- $S$ ו- $V\setminus S$. עפ"י סעיף ב, בעץ הפורש $T'$ קיימת לפחות קשת אחת $e'=(x,y)$ כך ש- $x\in S$ ו- $y\in V\setminus S$. נבחין כי לא יתכן ש- $e'$ נמצאת גם ב $T$, שכן נתון לנו שהורדת $e$ יוצרת שני רכיבי קשירות. אזי, עפ"י משפט 2, הוספת $e'$ ל- $T$ יוצרת מעגל. נבחין כי $e$ נמצאת במעגל, שכן הסרת $e$ לא תפצל כעת את הגרף לשני רכיבי קשירות (מכיוון שבחרנו את $e'$ להיות הקשת שתחבר בין הרכיבים הנוצרים על ידי הסרת $e$). ממשפט 2, $T''=(V,E_T\setminus\{e\}\cup\{e'\})$ הוא עץ פורש של $G$. נבחין כי $w(e')\leq w(e'')<w(e)$ (כי $e''$ קשת כבדה ביותר ב- $T'$). נקבל כי $w(T'')=w(T)-w(w)+w(e')<w(T)$ בסתירה להנחה כי $T$ עץ פורש מינימום של $G$. כלומר, $T$ הוא עף פורש קשת-כבדה של $G$

# שאלה 2 סעיף א

עבור הקלט $(1.5,1.5)$ האלגוריתם החמדן יחזיר פיתרון המכיל סדרה אחת $S_1=(1.5,1.5)$, שערכו $w(S_1)=1.5\cdot 1.5=2.25$. לעומת זאת, קיים פתרון אופטימלי המכיל שתי סדרות $S_1=(1.5), S_2=(1.5)$, פתרון שמשקלו $w(S_1,S_2)=1.5+1.5=3$.
ישנם דוגמאות נגדיות נוספות לדוגמא: $(2,2,1,2,2)$.

# שאלה 2 סעיף ב

נגדיר את $\mathrm{OPT}[i]$ להיות הערך המקסימלי של פתרון חוקי לתת-הבעיה $A_i=(a_1,a_2,\dots,a_i)$.
מיקום הפתרון: $\mathrm{OPT}[n]$.

נוסחת המבנה:

$$\mathrm{OPT}[i]=\begin{cases}0 & i=0\\ \max\limits_{1\le k\le i}\{\mathrm{OPT}[k-1]+\prod\limits_{k\le j\le i}a_j\} & i\neq 0\end{cases}.$$

# שאלה 2 סעיף ג

עבור $i=0$, מתקיים $\mathrm{OPT}[0]=0$ כי הפיתרון היחיד הוא הפיתרון הריק שמשקלו 0. נוכיח את הנוסחה עבור $i>0$.

**<u>ניתוח מקרים</u>:**

עבור תת-הבעיה $(a_1,a_2,\dots,a_i)$ נגדיר לכל $1\le k\le i$ את $M_k$ להיות קבוצת הפתרונות החוקיים כך שתת הסדרה הרציפה האחרונה הינה $(a_k,a_2,\dots,a_i)$.

**<u>כיסוי</u>:**

עבור פתרון חוקי כלשהו $S$ עבור תת-הבעיה $A_i=(a_1,a_2,\dots,a_i)$ קיימת תת-סדרה רציפה בה $a_i$ מופיע. תהי $(a_k,a_{k+1},\dots,a_i)$ תת-הסדרה הנ"ל. מכיון ש-$S$ הינו פתרון חוקי, זוהי תת-סדרה רציפה עם מספרים מהסוף של $A_i$. מכאן ש- $S\epsilon M_k$, לפי הגדרת $M_k$.

**<u>מסקנה לנוסחת המבנה</u>:**

נגדיר את $O^*(M_k)$ להיות הערך המקסימלי של הפתרונות ב-$M_k$. כלומר,
$O^*(M_k)=\max\limits_{S\epsilon S_k} w(S)$.
מ-א' ו-ב' נקבל כי $\mathrm{OPT}[i]=\max_{1\le k\le i}\{O^*(M_k)\}$.

**<u>ניתוח האופטימום של כל קבוצה</u>:**

נוכיח כי לכל $1\le k\le i$ מתקיים $O^*(M_k)=\mathrm{OPT}[k-1]+\prod_{k\le j\le i}a_j$.

<u>כיוון 1: $\geq$</u>

יהי פתרון חוקי $S$ לתת-הבעיה $(a_1,a_2,\dots,a_{k-1})$ כך ש- $w(S)=\mathrm{OPT}[k-1]$. כעת, נסתכל על הפתרון $S^*=S\cup\{(a_k,a_{k+1},\dots,a_i)\}$. $S^*$ הינו פתרון חוקי ל- $A_i=(a_1,a_2,\dots,a_i)$ מכיוון ש- $(a_k,a_{k+1},\dots,a_i)$ הינה תת-סדרה רציפה של- $A_i$ ומכיוון ש- $S$ הינו פתרון חוקי ל-$A_{k-1}$ שאינו מכיל אף איבר מ- $(a_k,a_{k+1},\dots,a_i)$. מכיוון ש- $(a_k,a_{k+1},\dots,a_i)$ הינה תת-סדרה רציפה מהפתרון $S^*$, מהגדרה, $S^*\epsilon M_k$. מכאן ש- $O^*(M_k)\ge w(S^*)$ (ערך פתרון אופטימלי גדול או שווה לערך כל פיתרון חוקי). כמו כן, $w(S^*)=w(S)+w\big((a_k,a_{k+1},\dots,a_i)\big)=\mathrm{OPT}[k-1]+\prod_{k\le j\le i}a_j$. מכאן ש- $O^*(M_k)\ge \mathrm{OPT}[k-1]+\prod_{k\le j\le i}a_j$, כנדרש.

<u>כיוון 2:</u> $\le$

יהי $S\epsilon M_k$ פתרון חוקי לתת-הבעיה $A_i = (a_1, a_2, \dots, a_i)$ כך ש- $w(S) = O^*(M_k)$. אזי, לפי הגדרת $M_k$, תת-סדרה הרציפה האחרונה ב- $S$ היא $(a_k, \dots, a_i)$. הפתרון $S^* = S\backslash\{(a_k, a_{k+1}, \dots, a_i)\}$ הינו פתרון חוקי עבור תת-הבעיה $(a_1, a_2, \dots, a_{k-1})$ ומכאן ש- $w(S^*) \le \mathrm{OPT}[k-1]$. כמו כן, מהגדרת $w$,

$$O^*(M_k) = w(S) = w(S\backslash\{(a_k, a_{k+1}, \dots, a_i)\}) + w\big((a_k, a_{k+1}, \dots, a_i)\big) \le w(S^*) + \prod_{k\le j\le i} a_j$$

ובסה"כ, $O^*(M_k) \le OPT[k-1] + \prod_{k\le j\le i} a_j$. כנדרש.

הוכחנו את 2 הכיוונים ולכן השוויון נכון ולכן נוסחת המבנה נכונה.

# שאלה 2 סעיף ד

נגדיר מערך $D$ חד מימדי שהאינדקסים שלו הם בין 0 ל- $n$. כמו כן, נניח כי $A[0] = 1$.

<u>Iterative_opt_MulSum</u>*(A):*

1. For $i$ = 0 to $n$
   a. $D[i] = 0$
2. For $i$ = 1 to $n$
   a. Mul = $A$[i]
   b. For $j = i$-1 to 0
      i. If ( (Mul+$D[j]$) > $D[i]$)
         1. $D[i]$ = Mul+$D[j]$
      ii. Mul = Mul*$A[j]$
3. Return $D[n]$

<u>זמן ריצה:</u>

2 לולאות, בכל אחת $O(1)$ פעולות. בסה"כ $O(n^2)$ זמן ריצה.

# שאלה 2 סעיף ה

נגדיר שני מערכים $D$ ו- $C$ חד מימדים שהאינדקסים שלהם הם בין 0 ל- $n$, כאשר $C[i]$ יכיל את מספר הפתרונות האופטימליים עבור $(a_1, a_2, \dots, a_i)$. כמו כן, נניח כי $A[0] = 1$.

<u>Iterative_opt_MulSum+NumAnswers</u>*(A):*

1. For $i$ = 0 to $n$
   a. $C$[i] = 0, $D$[i] = 0,
2. For $i$ = 1 to $n$
   a. Mul = $A$[i]
   b. For $j = i$-1 to 0
      i. If ( (Mul+$D[j]$) = $D[i]$ )
         1. $C[i]$ =$C[i]$ +$C[j]$
      ii. If ( (Mul+$D[j]$) > $D[i]$ )
         1. $D[i]$ = Mul+$D[j]$
         2. $C[i]$ =$C[j]$
      iii. Mul = Mul*$A[j]$
3. Return $D[n]$

<u>זמן ריצה:</u>

2 לולאות, בכל אחת $O(1)$ פעולות. בסה"כ $O(n^2)$ זמן ריצה.