פתרון ממ"ן 11 – סמסטר 2020ג

פתרון שאלה 2

אלגוריתם המקבל עץ בינרי המייצג ביטוי אריתמטי ומחשב את ערך הביטוי:

חשב-ביטוי (T)

(1) אם T הוא עלה, אז החזר את info(T).

(2) אם ל-T יש רק בן שמאלי, אז בצע:

(2.1) קרא ל- **חשב-ביטוי** עם left(T) והצב את הערך המוחזר ב-X;

(2.2) החזר את X–.

(3) אם ל-T יש רק בן ימני, אז בצע:

(2.1) קרא ל- **חשב-ביטוי** עם right(T) והצב את הערך המוחזר ב-Y;

(2.2) החזר את Y–.

(4) קרא ל- **חשב-ביטוי** עם left(T) והצב את הערך המוחזר ב-X;

(5) קרא ל- **חשב-ביטוי** עם right(T) והצב את הערך המוחזר ב-Y;

(6) אם info(T) = '+', אז החזר את X + Y.

(7) אם info(T) = '–', אז החזר את X – Y.

(8) אם info (T) = '×', אז החזר את X × Y.

(9) אם info (T) = '/', אז החזר את X / Y.

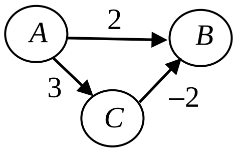
פתרון שאלה 3

א. הרצת האלגוריתם של דייקסטרה על הגרף שבעמוד 91 בספר:

מס' האיטרציה	הצומת הנבחר	העדכונים המתבצעים
1	*A*	$\lambda(C) = 5, \lambda(D) = 3, \lambda(G) = 14$
2	*D*	$\lambda(E) = 10, \lambda(G) = 9$
3	*C*	$\lambda(E) = 8, \lambda(F) = 7$
4	*F*	$\lambda(B) = 14$
5	*E*	$\lambda(B) = 13$
6	*G*	–
7	*B*	–

ב. האלגוריתם של דייקסטרה אינו מתאים לגרף בעל קשתות שליליות, כי אי אפשר יהיה לסמוך על כך שהערך $\lambda(V)$ עבור איזשהו צומת V הוא אכן המרחק הקצר ביותר מצומת המקור ל-V.

דוגמה נגדית:



האלגוריתם יתחיל בצומת A ויציב $\lambda(B) = 2$ ו- $\lambda(C) = 3$. כעת האלגוריתם יבחר את צומת B ויוציא אותו מקבוצת הצמתים שמרחקם מ-A טרם נקבע סופית. נקבל שאורך המסלול הקצר ביותר מ-A ל-B הוא 2, אבל המסלול A-C-B הוא באורך 1.

פתרון שאלה 4

א. האלגוריתם יחשב לכל סוג פריט את v_i / w_i (השווי ליחידת משקל).

לאחר מכן הוא יתחיל למלא את התרמיל בפריטים מהסוג שעבורו v_i / w_i הוא מקסימלי.
אם יישאר מקום בתרמיל הוא יעבור לסוג הבא עפ״י השווי ליחידת משקל וכך הלאה.
הערה: פסאודו-קוד של אלגוריתם חמדני לפתרון הגרסה המקורית של הבעיה מופיע בפתרון שאלה 15 בפרק 4 במדריך הלמידה.

ב. הקלט לבעיה:

N = 5, W = 70

q = [3, 1, 4, 3, 2]

w = [10, 20, 25, 8, 7]

v = [15, 42, 30, 16, 18]

נסמן $m_i = v_i / w_i$ ($1 \le i \le N$). הערכים המתקבלים עבור הקלט הנתון:

$m_1 = 15/10 = 1.5$

$m_2 = 42/20 = 2.1$

$m_3 = 30/25 = 1.2$

$m_4 = 16/8 = 2$

$m_5 = 18/7 = 2.57$

הפריטים שהאלגוריתם יבחר:

2 פריטים מסוג 5 – 14 ק״ג, 36 ₪

1 פריטים מסוג 2 – 20 ק״ג, 42 ₪

3 פריטים מסוג 4 – 24 ק״ג, 48 ₪

1.2 פריטים מסוג 1 – 12 ק״ג, 18 ₪

בסה״כ – 70 ק״ג, 144 ₪

פתרון שאלה 5

א. הסבר הנוסחה: עלות החיתוך הראשון היא בכל מקרה $a_j - a_i$.

אח"כ מחפשים את המינימום על פני k ‏($i < k < j$) של סכום העלויות המינימליות של ביצוע החיתוכים בחלק השמאלי ובחלק הימני.

תנאי העצירה של הנוסחה הוא $\text{Cost}(a_i, a_{i+1}) = 0$.

ב. נכתוב אלגוריתם תכנון דינמי המחשב את העלות הכוללת המינימלית.

הקלט לאלגוריתם הוא מערך A[0..n+1] המכיל את n+2 הנקודות שעל המוט.

כדי לשמור את הפתרונות לתת-בעיות, האלגוריתם משתמש במערך דו-ממדי C[0..n, 0..n+1].

גודל המערך הוא $(n+1) \times (n+2)$.

(1) עבור i המקבל את הערכים 0 עד n בצע:

(1.1) $C[i, i+1] \leftarrow 0$

(2) עבור h המקבל את הערכים 2 עד n+1 בצע: /* h מציין את הגודל של תת-בעיה */

(2.1) עבור i המקבל את הערכים 0 עד n+1-h בצע:

(2.1.1) $j \leftarrow i+h$

(2.1.2) $C[i, j] \leftarrow A[j] - A[i] + \min_{i<k<j}\{C[i, k] + C[k, j]\}$

(3) החזר את C[0, n+1].

הלולאה בשורה (2) רצה על הגודל של התת-בעיות.

התת-בעיה הקטנה ביותר היא בגודל 2, והפתרון שלה הוא אורך המוט שבין הנקודות i ו- j (עבור שתי התת-בעיות בגודל 1 מתקיימים תנאי העצירה).

הפתרונות לתת-בעיות שהולכות וגדלות מחושבים באמצעות הנוסחה הרקורסיבית, על-פי הפתרונות שחושבו כבר לתת-בעיות הקטנות יותר.

התת-בעיה הגדולה ביותר היא בגודל n+1 והפתרון לתת-בעיה זו הוא הפתרון המבוקש.

זמן הריצה (לא נדרש לנתח אותו): $O(n^3)$