

תרגיל בית מספר 2

(להגשה עד 14/12/2021 בזוגות)

שאלה 1

נגדיר וריאציה לבעיית הסכומים החלקיים (Subset Sum Problem): נתונה קבוצה $A = \{a_1, a_2, \dots, a_n\}$ של n מספרים שלמים (לאו דווקא חיוביים) ומספר מטרם שלם S . האם קיימת תת-קבוצה $Q \subseteq A$ שסכומה בדיוק S ?

א. הציעו אלגוריתם יעיל לפתרון הבעיה הנ"ל. על האלגוריתם להחזיר תשובה בוליאנית, ובמקרה של תשובה חיובית גם תת-קבוצה Q המתאימה.

ב. נתחו סיבוכיות של האלגוריתם המוצע.

ג. הריצו את האלגוריתם המוצע על הקלט הבא: $A = \{9, -3, 2, 4, -5, -1\}$, $S = 7$. מה התשובה שמתקבלת (כולל תת-קבוצה אם קיימת)?

שאלה 2

נתון טקסט T באורך n ותבנית P באורך m .

תארו אלגוריתם יעיל ככל הניתן, המחשב לכל מספר שלם $1 \leq k \leq n$, את המספר הרישיות (לא ריקות) של P , המהוות סיפא של $T[1..k]$.

הוכיחו את נכונות האלגוריתם וחשבו את סיבוכיותו.

שאלה 3

א. הדגימו את פעולת האלגוריתם שבונה אוטומט לחיפוש מחרוזות על התבנית הבאה: $P = \text{BABBBABCAB}$. הראו את תוכן טבלת המצבים.

ב. הדגימו את פעולת האוטומט הנ"ל על הטקסט הבא: $T = \text{ABABABABBABCAA}$ (14 אותיות). (צינו באיזה מצב יהיה האוטומט אחרי קליטת כל תו ואילו הודעות ידפסו האלגוריתם)

ג. הציגו את הוקטורים π (state) ו- i שיחשב האלגוריתם KMP על הדוגמא הנ"ל.

שאלה 4

תלמידים כתבו את התוכניות הבאות לבעיית התאמת המחרוזות. קבעו לכל תוכנית:

א. מה סיבוכיותה (במקרה הגרוע)?

ב. האם היא:

- מוצאת את כל המופעים
 - (או) מוצאת לפחות מופע אחד, אם יש
 - (או) עלולה שלא למצוא אף מופע, למרות שיש?
- נמקו תשובתכם (על ידי דוגמאות, או הסברים, לפי הצורך).

PROGRAM 1

```
// Input: text T[1..n], pattern P[1..m]
i ← 1; j ← 1
while i ≤ n do {
    if P[j] = T[i] then
        if j = m then
            print "Pattern occurs with shift" i - m + 1
            j ← 1
        else
            j ← j + 1
    else
        j ← 1
    i ← i + 1
}
```

PROGRAM 2

```
// Input: text T[1..n], pattern P[1..m]
i ← 1; j ← 1
while i ≤ n do {
    if P[j] = T[i] then
        if j = m then
            print "Pattern occurs with shift" i - m + 1
            j ← 1; i ← i - m + 2
        else
            j ← j + 1; i ← i + 1
    else
        j ← 1; i ← i + 1
}
```

PROGRAM 3

```
// Input: text T[1..n], pattern P[1..m]
i ← 1; j ← 1; k ← 1
while k ≤ n - m + 1 do {
    if P[j] = T[i] then
        if j = m then
            print "Pattern occurs with shift" k
            k ← k + 1; j ← 1; i ← k
        else
            j ← j + 1; i ← i + 1
    else
        k ← k + 1; j ← 1; i ← k
}
```

בהצלחה!