# ממן 13

אלגוריתמיקה 20290

16.08.2019

## 

[**ממן**](#_6wpwt0gra8n2) **13 1**

[שאלה](#_940e35q6racc) 1 2

[סעיף א](#_31ovr3cl7qcs) 2

[סעיף ב](#_ltswlaw8ut2r) 2

[שאלה](#_ed05ibzwm0d) 2 3

[שאלה](#_exodmta504w2) 3 4

[סעיף א](#_6iaeb5j168b2) 4

[סעיף ב](#_ym5x672tozzj) 4

[שאלה](#_vy1nqebxf77b) 4 5

[סעיף א](#_ei23srty5lpz) 5

[סעיף ב](#_l01jlxpupv8j) 5

[סעיף ג](#_huzpb0l9bnnh) 5

[שאלה](#_kfp26wqosedr) 5 6

[סעיף ב](#_4vgcrht3bpfs) 6

[סעיף ג](#_1vjfpoy9btgg) 6

[שאלה](#_ugm38nmwezeb) 6 7

## 

## שאלה 1

### סעיף א

אתאר אלגוריתם נאיבי לפתרון בעית הסכום החלקי.  
הקלט- רשימה מקושרת A ומספר k

|  |
| --- |
| SubsetSum(A, k):  if k=0:  return true  else: if k<0:  return false  for i ← 1 to size(A) // if A is empty  x ← pop(A, i) // remove the i number from the list and return it’s value  if SubsetSum(A, k-x):  return true  return false // no subset was found |

הסבר:

אם k=0 ברור כי קיימת תת קבוצה ריקה, נמשיך עבור k גדול מ- 0.

אם קיימת תת קבוצה B של A כך שסכום איברי B שווה k, אזי קיים ב- B איבר שערכו x כך ששאר איברי B משלימים אותו ל- k, כלומר, שאר איברי B סכומם k-x. אבל B ללא האיבר שערכו x היא גם כן תת קבוצה של A (ללא אותו איבר), לכן אם היא קיימת יחזור ערך אמת.

זמן ריצה:

זמן הריצה במקרה הגרוע ביותר מושפע מגודל הקבוצה A (למשל המקרה בו המספר k גדול מסכום כל איברי הקבוצה)

נסמן ב- n את גודל הקבוצה A.

האלגוריתם עובר בלולאה לינארית על איברי A, ובמקרה הגרוע בודק לכל איבר את כל תת האפשרויות, כלומר אלגוריתם נאיבי זה בודק את כל תת הקבוצות האפשריות של A, ומחזיר אמת אם סכומן משלים ל- k.

זמן הריצה של אלגוריתם כזו הוא מכיוון שיש תתי קבוצות, ולכל תת קבוצה כזו מתבצע חישוב לינארי (מעבר על ענף העץ)

### סעיף ב

ניתן לחשוב על תהליך האלגוריתם הנאיבי כבניית עץ, לשורש יש n בנים (כמס' איברי הקבוצה הנתונה), הבן ה- i מייצג את תת העץ בו האיבר ה- i שייך לתת הקבוצה הנבדקת.

באמצעות שימוש במטבע קסם, האלגוריתם יכול לבחור ללכת רק במסלולים המובילים לבנים שמייצגים איברים השייכים לתת הקבוצה שיוצרת את הסכום הנדרש.  
בדומה לדוגמה ממדריך הלמידה בעמ' 124:

כל אחד מהאיברים בקבוצה הנתונה יכול להיות שייך לתת הקבוצה או לא שייך לתת הקבוצה- האלגוריתם יטיל את מטבע הקסם עבור כל משתנה כדי להחליט איזה ערך שייכות לתת לו. לאחר שהאלג' יקבע בדרך זו מהם האיברים השייכים לתת הקבוצה, הוא יסכום אותם, אם סכום איברי תת הקבוצה שווה למספר הנתון, האלג' יחזיר אמת, אחרת יחזיר שקר.

## שאלה 2

הוכחת שלמות בNP מתחלקת לשני שלבים.

בשלב הראשון יש להוכיח כי הבעיה בNP ובשלב השני להראות רדוקציה פולינומיאלית מבעיית NPC אחרת.

שייכות לNP:

ברור כי בהינתן מסמך אישור- שלוש השמות מספקות- ניתן לבדוק אותן בזמן לינארי, כלומר בדיקת מסמך האישור נעשית בזמן פולינומיאלי, לכן הבעיה בNP.

רדוקציה פולינומיאלית מבעיית SAT לבעיה הנתונה:

בהינתן קלט לבעית SAT- פסוק בתחשיב הפסוקים, נבצע בו רדוקציה פולינומיאלית, נוסיף עם קשר "או" את הפסוק , כלומר נקבל את הפסוק: .

לפסוק שהתקבל יש לפחות 3 השמות מספקות אם ורק אם לפסוק ישנה השמה מספקת אחת לפחות.

כיוון ראשון: אם לפסוק ישנה השמה מספקת אז לפסוק ישנן לפחות 3 השמות מספקות:

ברור כי הפסוק המתקבל מסופק על ידי ההשמה המספקת את , כמו כן אם מקבל ערך "אמת", ואם מקבל ערך "אמת".

כיוון שני: אם ל- יש 3 השמות מספקות אז לפסוק ישנה השמה מספקת:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| T | T | T | F | F | T |
| T | T | F | F | T | T |
| T | F | T | T | F | T |
| T | F | F | F | F | T |
| F | T | T | F | F | F |
| F | T | F | F | T | T |
| F | F | T | T | F | T |
| F | F | F | F | F | F |

מארבעת השורות התחתונות בטבלה רואים בבירור כי אילו ל אין השמה מספקת, לפסוק יש 2 השמות מספקות בלבד.

מכאן, כיוון שהבעיה הנתונה שייכת לNP, וכיוון שישנה רדוקציה פולינומיאלית מבעית NPC ידועה (SAT), גם הבעיה הנתונה היא בעית NPC.

## שאלה 3

### סעיף א

לא, בעיית הקליקות עבור k קבוע היא פולינומיאלית, הבעיה k-clique שייכת לNP רק כאשר k לא נתון (שזו בעצם בעיית מציאת הקליקה המקסימלית).

לגרף עם n צמתים ישנם פחות מ- תתי גרפים המכילים 5 צמתים (כאלו שאולי הם קליקה בגודל 5), ובכל אחד מהם צריך לבדוק קיום של קשתות, כלומר הבעיה פתירה בזמן פולינומיאלי () עבור k=5, לכן מהוכחתו של הפרופ' אי אפשר להסיק כי P=NP.

### 

### סעיף ב

כן, אם הפרופ' הצליח למצוא פתרון פולינומי לבעית ה הרי כי ניתן להסיק מכך ש- P=NP.

בעית הצביעה היא בעיית NPC לכל (עבור 0 ו-1 זה טריוויאלי לחלוטין, ועבור 2 זו בעיית הגרף הדו צדדי שהיא כמובן לינארית למספר הקשתות), גם כאשר המספר הוא חלק מהקלט.  
לכן פתרון בעיה זו (שהיא כאמור בעית NPC- עמוד 173 בספר) בזמן פולינומיאלי בהחלט מוכיח כי P=NP.

## שאלה 4

### סעיף א

בדומה להסבר משאלה 3 סעיף ב', כאשר k נתון (קבוע) ניתן לפתור את הבעיה בזמן פולינומיאלי.

הרעיון הוא לבדוק עבור כל תת גרף האם הוא קליקה.

מבנה אלגוריתם אפשרי יהיה בנוי מ- k לולאות מקוננות, כאשר כל אחת רצה על רשימה של צמתי הגרף מהאינדקס של הלולאה החיצונית לה ועד הסוף.  
בתוך הלולאה הפנימית ביותר, תמיד יהיו לנו k אינדקסים מכל הלולאות החיצוניות, אשר מייצגים תת קבוצה של k צמתים. כעת נבדוק עבור כל זוג מתוך k הצמתים האלה אם קיימות קשת, במידה ולא, נמשיך לתת הקבוצה הבאה.

הנה דוג' של האלג' עבור k=4:

|  |
| --- |
| 4-Clique(V,E):  n ← |V|  if n<4:  return false  for a ← 1 to n:  for b ← a to n:  for c ← b to n:  for d ← c to n:  if  :  return true  return false // no subset was found |

\* ככל הנראה בכתיבה אלגוריתם כזה עבור k גדול יותר, ניתן למצוא שיטה טובה יותר מבדיקת קוניונקציה של כל הזוגות, אך אין פעולה זו משפיעה על פולינומיאליות האלגוריתם, שכן מתבצעות בה פעולות בלבד, ואם k קבוע אז גם פעולה זה היא במשך זמן קבוע.

### סעיף ב

ברור כי לא קיים אלגוריתם פולינומי לבעיה כאשר k הוא חלק מהקלט, כפי שמוצג במדריך הלמידה בעמוד 131.

לעומת זאת, אם k לא חלק מהקלט, אלה תלוי בצורה "קבועה" ב-n אז כן ניתן לבנות אלגוריתם פולינומי לבעיה זו. למשל עבור k=n, הבעיה היא בעצם "האם הגרף הנתון הוא קליקה", ובעיה זו כן פתירה פולינומיאלית, קל לעבור על כל זוג מצמתי הגרף ולבדוק קיום קשת. אלג' כזה ירוץ בזמן .

באופן דומה אם ההפרש הקבוע הוא k=n-1 ניתן לעבור לינארית על צמתי הגרף ולשלוח לאלג' הקודם כל פעם את הגרף פחות צומת אחת (וכל הקשתות שלה), אלג' זה ירוץ בזמן , וכן הלאה לכל הפרש קבוע בין k ל- n.

לעומת זאת, עבור k=n/2 להערכתי אין אלג' מתאים, אך אין לי הכלים להראות זאת.

### סעיף ג

ניתן לעבור על צמתי הגרף בלולאה לינארית וכל פעם להוציא צומת מהגרף, ולבדוק באמצעות השגרה הנתונה אם עדיין יש קליקה כזו בגרף. אם עדיין יש, אז הצומת הנוכחית לא חלק מהקליקה, נמחק אותה ונמשיך הלאה. אם לאחר הוצאת הצומת אין יותר קליקה כזו בגרף, אז צומת זו כן שייכת לקליקה ולכן נחזיר אותה לגרף ונמשיך הלאה.

(כמובן שבעת הוצאת צומת מהגרף נוציא גם את כל הקשתות המחוברות לצומת זו. בצורת ייצוג מתאימה של גרף ניתן לעשות זאת בזמן קבוע גם כן).

לבסוף נישאר עם k צמתים שכולם חלק מהקליקה.

## שאלה 5

### סעיף ב

הבעיה שייכת ל- NP, נראה זאת על ידי מסמך האישור: המספר m.

ברור שבהינתן מסמך אישור כזה, ניתן לבדוק בזמן קבוע אם m בין a ל- b ואם הוא גורם של n (כלומר אם ).

### סעיף ג

אם אפשר לפתור ביעילות את בעיית המשוואות הריבועיות, אשר היא בעית NPC, אז P=NP, מכאן שקיימת שגרה הפותרת את הבעיה מסעיף ב' בזמן פולינומיאלי.

אראה כי ניתן להשתמש בשגרה זו כדי לפרק מספרים גדולים לגורמים, נפרק את המספר n:

נעביר לשגרה הנ"ל את המספר n ואת המספרים 2 ו- n-1 בתור a ו- b, אם קיים גורם, אז נעביר רק חצי מטווח המספרים הזה (כלומר בין 2 ל- n/2 או בין n/2 ל- n-1), באחד מהם קיים הגורם. כך נוכל לצמצם את החיפוש ולמצוא את הגורם הראשון ב , זאת בדומה לחיפוש בינארי אחר הגורם.

כעת, למציאת גורמים נוספים אפשר לחלק את המספר n בגורם שכבר מצאנו, ולחזור על הפעולה לחיפוש הגורמים הנוספים.

ברור כי הגורם הקטן ביותר הוא 2, לכן בכל שלב נחלק את n ב- 2 לכל הפחות, מכאן שנבצע פעולה זו לכל היותר log n פעמים.

אורך הקלט הוא ייצוג של מספר n, כלומר log n, זהו גם סדר הגודל של הקלט לבעיה מסעיף ב', לכן מכיוון שהיא פולינומית לאורך הקלט, עבור k קבוע כלשהו מתקיים שהיא רצה בזמן מכאן שמציאת הגורמים מתבצעת גם כן בזמן סביר .

## שאלה 6

האלג' המוצע ברמז (משאלה 4 בממ"ן 12) מקבל קבוצה ומחזיר את כל הסכומים החלקיים שלה בסדר ממוין בזמן לקבוצה בגודל n.

על מנת לפתור את בעית הסכום החלקי בזמן , נחלק את הקבוצה הנתונה לשתי קבוצות שוות בגודלן, אין משמעות לצורת החלוקה. כל חלק נעביר לאלג' המוצע ברמז. עד כאן אנחנו בסיבוכיות הזמן הנדרשת.

כעת יש בידינו 2 מערכים ממוינים באורך , נעבור עליהם בצורה לינארית כך:

באחד המערכים נתחיל מהסוף ובשני מההתחלה, בכל איטרציה נבדוק אם סכום האיברים שווה לסכום מהקלט (המספר שאנחנו מחפשים). אם כן, אז קיימת תת קבוצה (איחוד תתי הקבוצות שהרכיבו את 2 המספרים). אם לא- אם הסכום קטן מידי, נקדם את המערך השני (זה שבו התחלנו מהאיבר הקטן), אם הסכום גדול מידי, נקדם את המערך הראשון (בו מתקדמים אחורה, כלומר נקטין את האינדקס במערך בו התחלנו מהאיבר הגדול).

בכל איטרציה לפחות אינדקס אחד מתקדם, לכן המעבר לינארי (על אורך סכום המחרוזות, אבל זה הכפלה בקבוע ולא משפיע על סיבוכיות הזמן).

לסיכום, יצירת המערכים בזמן , מעבר על המערכים בזמן , סה"כ בדיקה האם קיימת תת קבוצה המשלימה לסכום מסוים בזמן כנדרש.