אלגוריתמיקה (20290)

סמסטר ב98 – מועד א2 – פתרון הבחינה

שאלה 1

א. כן, האלגוריתם פותר את הבעיה. האלגוריתם בודק באופן שיטתי את כל התת-קבוצות בגודל k של V. אם קיים ב-G קליק בגודל k או יותר, האלגוריתם ימצא את הקליק (או תת-קבוצה בגודל k של צמתי הקליק, שהיא, כמובן, גם כן קליק) ויחזיר "כן".

ב. מס' האפשרויות לבחור k צמתים מתוך n הוא: $C_n^k = \dfrac{n!}{(n-k)!\,k!}$.

כדי לבדוק אם תת-קבוצה בגודל k היא קליק, צריך לבדוק אם כל שני צמתים בקבוצה מחוברים ע"י קשת. כלומר, מס' הבדיקות שצריך לבצע הוא: $C_k^2 = \dfrac{k!}{(k-2)!\,2!}$.

סה"כ: $\dfrac{n!}{(n-k)!k!} \cdot \dfrac{k!}{(k-2)!2!} = \dfrac{n!}{(n-k)} \cdot \dfrac{1}{2(k-2)!}$

כדי להראות שהאלגוריתם איננו פולינומיאלי נחשב את C_n^k כאשר k שווה ל- n/2:

$$C_n^{n/2} = \frac{n!}{(n/2)!\,(n/2)!} = \frac{n(n-1)\ldots(n/2+1)}{n/2(n/2-1)\ldots1} \geq 2^{n/2} = (\sqrt{2})^n$$

שאלה 2

1. לא נכון. הבעיה היא כריעה, אך היא איננה בהכרח לא סבירה (יתכן שיש גם אלגוריתם סביר שפותר אותה).
2. זו בעיה פתוחה (לא ידוע אם NP שווה ל- co-NP)
3. לא נכון. קרוב לודאי שבעיית הפריקות איננה שלמה ב-NP, ולכן גם אם יימצא פתרון פולינומיאלי לבעיית הפריקות, לא נוכל להסיק מכך ש- P = NP.
4. נכון. לכל בעיה ב NP קיים אלגוריתם אקספוננציאלי הפותר אותה.
5. נכון. למשל, בעיית מחסום הדרכים שייכת ל-EXP ואינה שייכת ל-P.

שאלה 3

א. המחלקה RP מוגדרת בספר (עמ' 310).

ב. האלגוריתם המתואר בספר באיור 11.1 הוא אלגוריתם אקראי, שפותר את בעיית הפריקות בזמן פולינומיאלי.

האלגוריתם מקבל כקלט מספר N.

אם N איננו פריק (כלומר, N הוא ראשוני), האלגוריתם לא יצליח למצוא עד מאשר לפריקותו של N, ולכן האלגוריתם יחזיר "לא" בהסתברות 1.

אם N הוא פריק, יש סיכוי גדול מ- ½ שהאלגוריתם ימצא עד מאשר לפריקותו של N, ולכן האלגוריתם יחזיר "כן" בהסתברות שגדולה מ- ½.

קיומו של אלגוריתם כזה מהווה הוכחה לכך שבעיית הפריקות שייכת ל-RP.

שאלה 4

האלגוריתם המבוקש יתקבל על-ידי ביצוע שינוי אחד באלגוריתם שבעמ' 138 בספר:

שורות (2.2) ו- (2.3) יתבצעו במקביל.

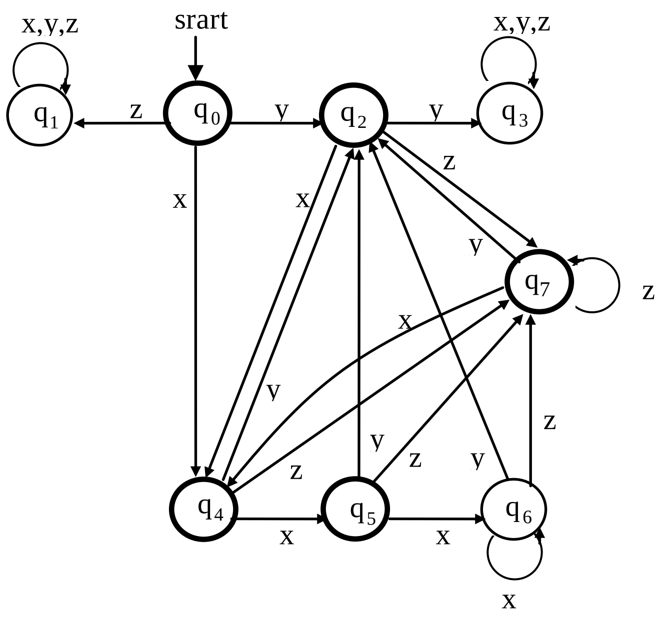
סיבוכיות הזמן המקבילית של האלגוריתם היא $O(\log n)$.

בכל פעם שמתבצעת שורה (2) התחום שבו מחפשים את המינימום והמקסימום מתחלק לשני חלקים והחיפוש בכל אחד מהם מתבצע במקביל. לכן נוסחת הנסיגה היא $C(n) = C(n/2) + 2$ והפתרון שלה הוא $C(n) = O(\log n)$.

האלגוריתם משתמש ב- n/2 מעבדים (בהנחה ש-n הוא חזקה שלמה של 2).

זהו מספר המעבדים שיידרש ברמה התחתונה של הרקורסיה.

שאלה 5



המצבים המקבלים הם $\{q_0, q_2, q_4, q_5, q_7\}$.