**גיליון יבש מספר 2 - מבני נתונים 234218 חורף 2020/21  
  
מגישים:   
נועם יעקבי - 316600782   
עידו קולמן - 316465202   
  
  
  
שאלה 1**

1. הטענה אינה נכונה:  
   כפי שלמדנו בהרצאה, הכנסת איבר לרשימת דילוגים רנדומלית מתבצעת בצורה הבאה:   
   מציאת האיבר או המיקום בו אמור להיות האיבר ברשימה המקושרת התחתונה המכילה את כל האיברים, ולאחר מכן הטלות מטבע לגבי יצירת צומת חדש מעל הרמה הנוכחית.  
   כאשר אנו מסירים איבר מרשימת הדילוגים אנו מסירים אותו מכל הרמות בהן הוא מופיע.   
   כלומר ייתכן מצב כי לפני ההסרה האיבר הופיע ב-3 רמות, הסרנו אותו ולכן הוא לא מופיע ברשימה, וכשאנו מוסיפים אותו מחדש הוא נכנס לאותו מקום ברמה התחתונה ביותר, אך מכיוון שכמות הרמות בהן הוא מופיע נבחרת בהטלה יכול להיות שכעת הוא מופיע רק ב-2 רמות ולא ב-3 כמו לפני כן,  
   ולכן פעולת ואחריה לא בהכרח אינן משנות את S.
2. הטענה אינה נכונה:  
   מכיוון ששואלים האם זה נכון עבור כל רשימת דילוגים, נפריך עבור רשימת דילוגים ששומרת בהתחלת הרשימה צומת שמפתחו וצומת בסוף שמפתחו .  
   עבור המקרה בו עשינו פעולת , אז נוסף לקומה הנמוכה ביותר והתחיל להטיל מטבע לגבי יצירת צמתים חדשים מעליו, נניח מצב בו הוא היחיד שנמצא ברמה הכי גבוהה,  
   ולאחר פעולת הצומת הוסר מהרשימה אך נותרה הרמה הגבוהה ביותר (אשר נמצאים בה רק ו-), קומה אשר לא הייתה לפני פעולות אלה, ולכן פעולות אלה עלולות לשנות את S.
3. הטענה אינה נכונה:  
   כפי שלמדנו בהרצאה, עץ פיבונאצ'י בגובה הוא עץ AVL תקין המקיים ,  
   ועץ מלא בגובה הוא עץ AVL תקין המקיים .  
   נבחר את להיות עץ פיבונאצ'י ואת להיות עץ מלא, שניהם בגובה .  
   נשיב לב כי מתקיים:
4. הטענה אינה נכונה:  
   ניקח שני עצי 2-3 בני צמתים,   
   נבחר בעץ A להיות עץ בגובה המקסימלי האפשרי, כאשר לכל צומת יהיו 2 בנים, ולכן כפי שלמדנו בהרצאה הגובה של עץ זה יהיה ,  
   ונבחר בעץ B להיות העץ בגובה המינימלי האפשרי, כאשר לכל צומת יהיו 3 בנים, ולכן כפי שלמדנו בהרצאה הגובה של עץ זה יהיה .  
   כעת נבדוק למה שווה :  
     
     
   כלומר קיבלנו כי עבור דוגמה זו ולכן בפרט .
5. הטענה נכונה:  
   נשים לב כי על כל צומת נעבור לכל היותר 3 פעמים, פעם אחת כאשר נגיע אליו למעלה, פעם שנייה כאשר נרד לתת העץ השמאלי שלו ואז נחזור אליו, ופעם שלישית כאשר נרד לתת העץ הימני ונחזור אליו.  
   כלומר, עבור כל פעולות נעבור על כל צומת פעמים (מינימום 1, לכל היותר 3) כך שמתקיים:  
     
   כלומר, יש פה פעולות שמתבצעות בזמן ,   
   ולכן הסיבוכיות המשוערכת תהיה .

**שאלה 3**

1. מבנה הנתונים שנשתמש בו הוא עץ 2-3, כאשר המפתח של כל עלה הוא מספר היום המתאים לו וכן נשמור בעלה גם מספר החולים שביום זה, כל צומת שומר את מספר החולים המקסימלי עבור כל תת עץ שלו. *- נאתחל עץ 2-3 ריק, נאתחל את מקסימום החולים עבור כל תת עץ של השורש להיות 0,  
   כפי שלמדנו אתחול עץ 2-3 ריק נעשה בסיבוכיות של .  
     
    – נכניס למבנה הנתונים שלנו שהוא עץ 2-3 עלה שהמפתח שלו הוא וישמור במידע שלו את בתור מספר הימים שלו.   
   כאשר נבצע מעבר על העץ נבדוק בכל צומת, לפני שאנחנו נכנסים לתת העץ המתאים לצורך הכנסת המידע, האם גדול ממספר החולים המקסימלי בתת עץ זה.   
   אם הוא גדול ממנו, נשנה את המקסימום של תת העץ בצומת להיות (בהנחה שבדקנו לפני כך שלא קיים יום בעל מזהה , אחרת נשמור את המקסימום הקודם ונעדכן חזרה במידה ומצאנו כבר יום בעל מזהה ),  
   אחרת לא נשנה את מקסימום זה.   
   כפי שלמדנו בהרצאה סיבוכיות הכנסה לעץ 2-3 היא ומכיוון שאנו עושים מספר פעולות סופי בכל צומת סיבוכיות הפעולה הכוללת היא .  
     
    – אנו מחפשים את היום במבנה הנתונים שלנו, לאחר שמצאנו אותו שומרים את מספר החולים שבו ומתחילים לבצע חיפוש הפוך בעץ שלנו ולעלות חזרה למעלה.  
   עבור כל צומת שעוברים בו בודקים את מקסימום החולים עבור כל תת עץ שקטן מהיום .  
   אם מצאנו תת עץ שהמזהה שלו קטן מ- שיש בו מקסימום גדול יותר מאשר (מספר החולים של ) מחפשים את העלה המקסימלי שמספר החולים בו גדול מ-(חיפוש לפי מידע המקסימום בצומת) ומחזירים אותו.  
   אם לא מצאנו, ממשיכים לעלות למעלה ומבצעים אותה בדיקה עד שמוצאים.   
   במידה והגענו לשורש ולא מצאנו אף יום שקטן ממנו אבל גרוע ממנו נחזיר .  
   כפי שלמדנו חיפוש היום בעץ 2-3 נעשה בסיבוכיות , ובחיפוש ההפוך במקרה הכי גרוע נעבור על גובה העץ פעמיים ונבצע מספר פעולות קבוע בכל צומת בסיבוכיות כוללת של ,  
   ולכן הסיבוכיות הכוללת של הפעולה היא .*
2. בדומה לסעיף א' גם כאן נשתמש בעץ 2-3, כאשר המפתח של כל עלה הוא מספר היום המתאים לו וכן נשמור בעלה גם מספר החולים שביום זה, ובכל צומת שומרים מידע על תתי העצים שלו: בדומה לסעיף א' נשמור את מקסימום החולים בכל תת עץ וכן נשמור את מספר הימים בכל תת עץ של הצומת.  
     
   *- נאתחל עץ 2-3 ריק,   
   נאתחל את מקסימום החולים ואת מספר הימים עבור כל תת עץ של השורש להיות 0,  
   כפי שלמדנו אתחול עץ 2-3 ריק ועשיית מספר פעולות סופי בשורש נעשים בסיבוכיות של .  
     
    – נכניס למבנה הנתונים שלנו שהוא עץ 2-3 עלה שהמפתח שלו הוא וישמור במידע שלו את בתור מספר הימים שלו.   
   כאשר נבצע מעבר על העץ נבדוק בכל צומת, לפני שאנחנו נכנסים לתת העץ המתאים לצורך הכנסת המידע, האם גדול ממספר החולים המקסימלי בתת עץ זה.   
   אם הוא גדול ממנו, נשנה את המקסימום של תת העץ בצומת להיות (בהנחה שבדקנו לפני כך שלא קיים יום בעל מזהה , אחרת נשמור את המקסימום הקודם ונעדכן חזרה במידה ומצאנו כבר יום בעל מזהה ),  
   אחרת לא נשנה את מקסימום זה.   
   כאשר נכנסים לתת עץ של צומת מגדילים את קאונטר הימים שלו באחד.  
   כפי שלמדנו בהרצאה סיבוכיות הכנסה לעץ 2-3 היא ומכיוון שאנו עושים מספר פעולות סופי בכל צומת סיבוכיות הפעולה הכוללת היא .  
     
    – אנו מחפשים את הימים ו- במבנה הנתונים שלנו, לאחר שמצאנו אותם שומרים את מספר החולים שבשניהם ומתחילים לבצע חיפוש הפוך בעץ שלנו ולעלות חזרה למעלה.  
   נבצע חיפוש הפוך משני העלים ו- עד שנגיע לצומת המשותף הראשון שלהם (או האחרון מלמעלה), כלומר לצומת ברמה הכי נמוכה ששניהם מגיעים אליה לפני שהם מתפצלים לתת עצי שונים.  
   לפני תחילת החיפוש נאתחל קאונטר של ימים בין ו- להיות 0 ונבדוק אותו בסוף.  
   עבור החיפוש ההפוך מ-נבדוק בכל צומת שעוברים בו את מקסימום החולים עבור כל תת עץ שגדול מהיום , וכן את מספר הימים ששמורים בכל תת עץ כזה ונוסיף אותם לקאונטר הימים שלנו.  
   בצורה דומה יתבצע החיפוש ההפוך עבור , רק שנבדוק את כל תתי העצים הקטנים ממנו במקום את הגדולים כמו ב-.  
   אם מצאנו תת עץ שהמזהה שלו בין ו- שיש בו מקסימום גדול יותר מאשר (מספר החולים של ) מחזירים בפעולה.  
   אם לא מצאנו, ממשיכים לעלות למעלה עד לצומת המשותף הראשון ומבצעים אותה בדיקה עד שמוצאים.   
   במידה והגענו לצומת המשותף ולא מצאנו אף יום גרוע יותר נבדוק את קאונטר הימים שלנו,  
   אם הוא שווה נחזיר , עבור כל תוצאה אחרת של הקאונטר נחזיר .  
   כפי שלמדנו חיפוש יום בעץ 2-3 נעשה בסיבוכיות , ומכיוון שחיפשנו שני ימים זה נעשה בסיבוכיות .   
   בחיפוש ההפוך במקרה הכי גרוע נעבור על גובה העץ פעמיים ונבצע מספר פעולות קבוע בכל צומת בסיבוכיות כוללת של .  
   לכן הסיבוכיות הכוללת של הפעולה היא .*

**שאלה 5**

*מבנה הנתונים שלנו יהיה מורכב משלושה עצים ששומרים מחרוזות, עץ עבור המילים הימניות במחרוזת, עץ עבור המילים השמאליות במחרוזות ועץ ששומר את כל המחרוזת כולל המקף. לכל צומת בעצים אלו יהיו בנים כמספר האותיות בא"ב הנתון* Σ *ועוד אחד (עבור בדיקה שלנו) ומידע על כמה מחרוזות עוברות בכל בן של הצומת.  
  
כל צומת מחזיק מערך ששומר מצביעים לבנים שלו, אחד עבור כל אות בא"ב וכן קאונטר עבור מספר המחרוזות,  
בהנחה כי ובהנחה כי כל צומת תיראה כך:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *@* | *z* | *...* | *...* | *c* | *b* | *a* | *Node\** |
| *1* | *0* | *...* | *...* | *2* | *0* | *1* | *Counter* |

*- נשמור את הפרמטר m ונאתחל שלושה עצים ריקים שכאלה,   
כפי שלמדנו אתחול עץ ריק ועשיית מספר פעולות סופי בשורש נעשים בסיבוכיות של .  
  
 – נכניס לעצים שלנו את שתי המילים הללו, לעץ המילים השמאליות נכניס את , לעץ המילים הימניות נכניס את , ולעץ המשותף נכניס את .  
הכנסת לעץ מתבצעת בצורה הבאה: מהשורש נמשיך לתת העץ שמתאים לאות הראשונה של המילה ונוסיף אחד לקאונטר של אות זו, כל נמשיך עבור כל אות של המילה עד שנגיע לסופה.   
כשהגענו לסוף המילה נכניס @ אחרי האות האחרונה על מנת לסמן שיש מילה שהיא בדיוק המסלול מהשורש ועד ל-@ (נראה כי זה לא הכרח כאשר אנחנו גם כך משתמשים בקאונטר של האותיות אך מקל על המימוש והרעיון).  
מכיוון שאנו עוברים על צמתים בדיוק כמספר האותיות במילה, ומבצעים מספר פעולות סופי בכל צומת,  
סיבוכיות ההכנסה של המילה לעץ נעשית בסיבוכיות כוללת של .  
באותה צורה תתבצע ההכנסה של המילה וכן המילה , בסיבוכיות ו-   
בהתאמה ולכן הסיבוכיות הכוללת של הפעולה תהיה .  
  
 – נבדוק האם המילה נמצאת בעץ המשותף שלנו, במידה ולא נמצאת נחזיר , אחרת נבצע חיפוש עבור בעץ המילים השמאליות ו-בעץ המילים הימניות.  
עבור חיפוש המילה בעץ השמאלי, בצומת ה-i שנעבור בה נסכום את כל הקאונטרים של האותיות הקטנות מהאות ה-i של המילה , כך שאנו בודקים מה מספר המילים הקטנות ממש לקסיקוגרפית מהמילה בכל העץ של המילים השמאליות.  
באותה צורה נבדוק עבור מה מספר המילים הגדולות ממנה לקסיקוגרפית בעץ המילים הימניות.  
לאחר שתי בדיקות אלו נבדוק האם מספר המילים שקטנות מ-הוא לפחות והאם מספר המילים שגדולות מ-גם הוא לפחות . במידה וכן נחזיר , אחרת נחזיר .  
מכיוון שאנו עוברים על מספר צמתים כמספר האותיות במילה המשותפת, בדיקה האם המילה נמצאת בעץ המשותף לוקחת ,   
בחיפוש המילה וסכימת הקאונטרים של האותיות הקטנות ממנה בכל צומת, נעבור ב-צמתים ונבצע פעולות בכל צומת. מכיוון ש- הוא מספר קבוע וידוע נקבל כי הבדיקה עבור המילה נעשית ב-  
בצורה דומה בדיקת נעשית ב-ולכן הסיבוכיות הכוללת של כל הפעולה תהיה .*