**שאלה 1:**

1. אפשרות ג לא אפשרית : משום שלוקחים את הבן השמאלי של 911 ש363 קטן ממנו, ואז את הבן הימני של 240 ש363 גדול ממנו ומגיעים ל912 שלא ייתכן שנמצא בתת ענף הזה משום שהוא גדול מ911 אבל חלק מהתת עץ השמאלי שלו ונמצא במקום של הצמתים שקטנים ממנו.

אפשרות ה לא אפשרית: לוקחים את הבן הימני של 278 ש363 גדול ממנו ומגיעים ל347 שלוקחים את הבן הימני שלו כי 363 גדול ממנו ומגיעים ל621 שלוקחים את הבן השמאלי וכך מגיעים ל299 שלא אפשרי שם משום שהוא נמצא בתת עץ הימני של 347 למרות שהוא קטן ממנוף.

1. 1. האלגוריתמים של חיפוש איבר בעץ ושל הוספת איבר לעץ הם זהים לחלוטין כלומר מספר ההשוואות בהם זהים, כאשר בחיפוש יש השוואה נוספת על האיבר עצמו שזה הפלוס אחד.

2. כן, אם האיבר שנמחק הוא חלק מהמסלול שמוביל לאיבר שמחפשים, מספר ההשוואות ישתנה ומספר הצמתים שנבדקים במהלך החיפוש יהיה שווה למספר הצמתים שנבדקו כאשר ערך זה הוכנס לעץ לראשונה, במידה והערך שנמחק אינו חלק מהמסלול שמוביל לאיבר שמחפשים, משפר ההשוואות לא ישתנה.

3. התשובה תשתנה, משום שבעץ אדום שחור לאחר הכנסה יכול להתבצע fixup שמבצע עוד השוואות.

**שאלה 2:**

1. נחפש פתרון למקרה הכללי:

עבור n=1 קיים רק עץ אפשרי אחד

עבור n=2 קיימות 2 אפשרויות האיבר הגדול הוא שורש והקטן בן שמאלי, או שהקטן שורש והגדול בן ימני

עבור n=3 קיימות חמישה אפשרויות:

האיבר הגדול בשורש ואז מספר האפשרויות לסידור תת עץ בגודל 2 (שהוכחנו שיש 2 אפשרויות)

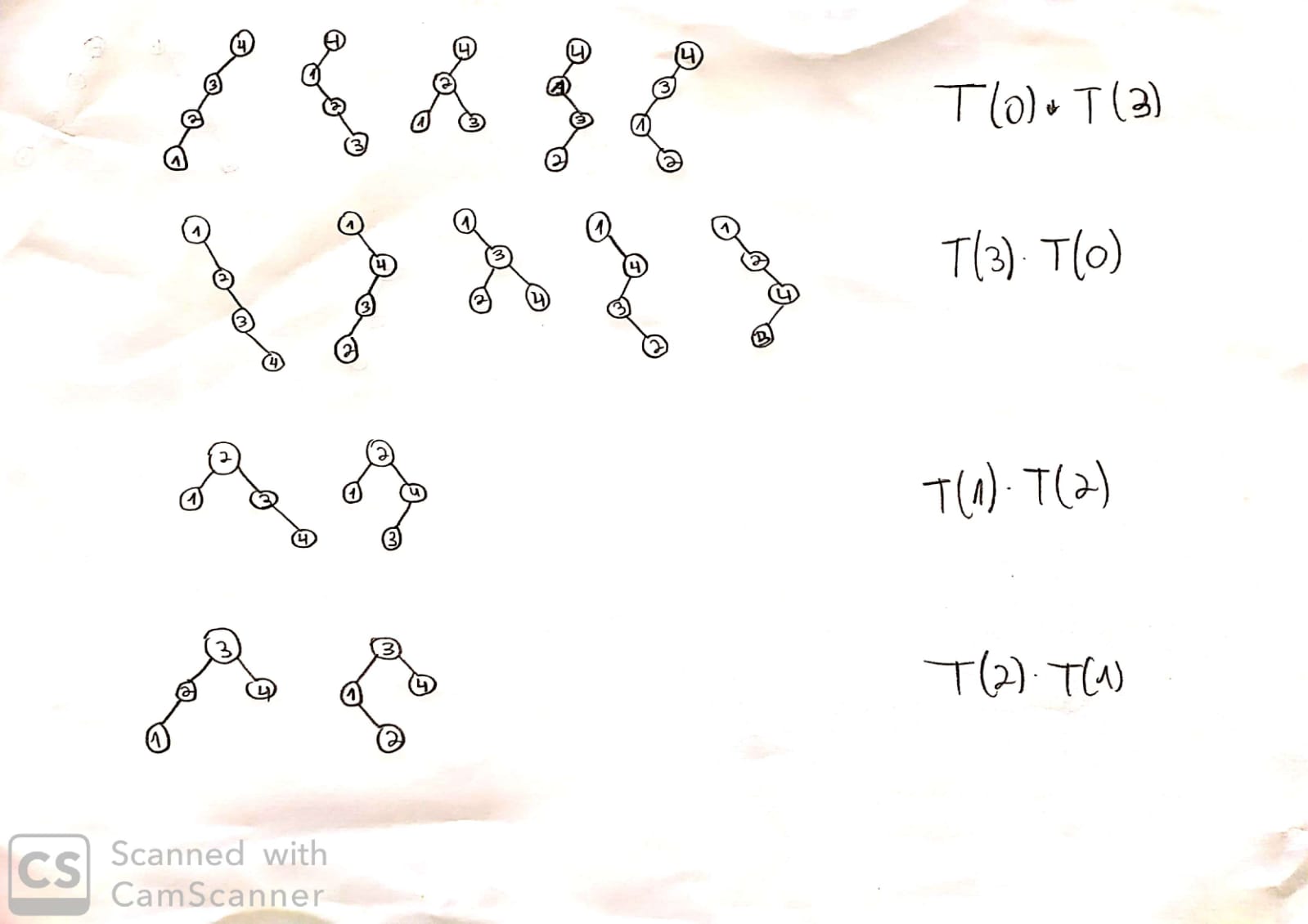
האיבר הקטן בשורש ואז מספר האפשרויות לסידור תת עץ בגודל 2 (שהוכחנו שיש 2 אפשרויות)

האיבר האמצעי בשורש, ואז הגדול בן ימני והקטן בן שמאלי.

אפשר לחשוב על זה בצורה של :

כלומר במקרה הכללי:

*וכך אפשר לחשב את :*



1. ניקח את מספר העצים מסעיף א (14) ונחשב את התוחלת על ידי סכימת הגובה \* ההסתברות לקבלת את הגובה (מספר העצים בגובה הנוכחי מתוך סך כל העצים)

גובה 2: 6 עצים

גובה 3 :8 עצים

ההסתברות לקבל עץ בגובה 2 היא

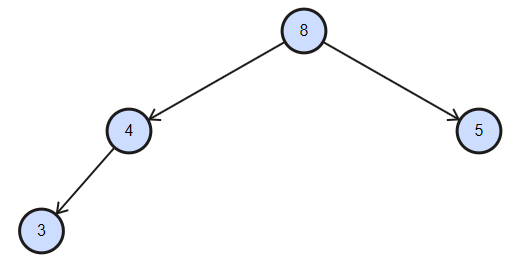
*ההסתברות לקבל עץ בגובה 3 היא*

*ולכן התוחלת תהיה*

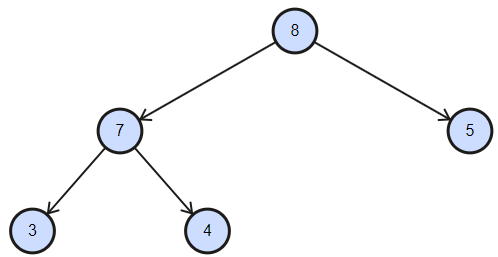
**שאלה 3:**

1. ערימה : לא, נראה דוגמא נגדית:

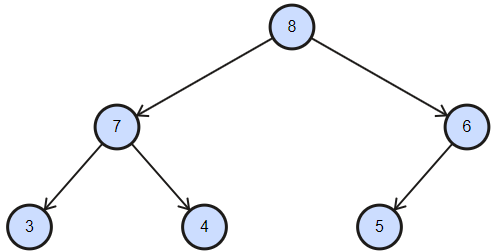
הערימה ההתחלתית:



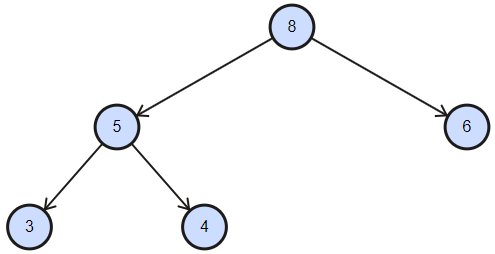
לאחר הכנסה של 7:



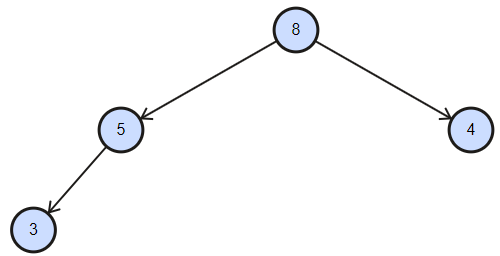
לאחר הכנסה של 6:



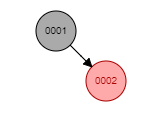
לאחר הוצאה של 7:



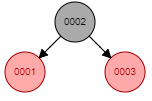
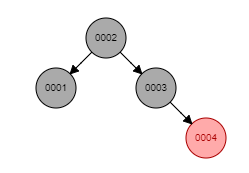
לאחר הוצאה של 6:



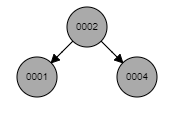
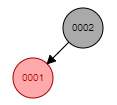
נניח עץ אדום שחור שבו 2 איברים 1,2 ונראה כך:



נכניס 2 איברים 3 ו4 בסדר עולה:

אחרי הכנסת 3:  אחרי הכנסת 4: 

כעת נמחק את 2 האיברים בסדר הגעתם:

אחרי מחיקת 3: אחרי מחיקת 4: 

וזה עץ שונה מהעץ הראשוני.

טבלת גיבוב עם פתרון התנגשויות: כן, אין הכנסה והוצאה של איברים בטבלת גיבוב לא משפיע על סדר או איברים אחרים בטבלה.

עץ חיפוש בינרי: כן, 2 האיברים שיכנסו בהכרח יהיו עלים, ולכן הכנסה ומחיקה שלהם לא תשפיע על שום סדר בעץ משום שאף איבר לא מתחלף.

עץ אדום שחור: לא, נראה דוגמא נגדית

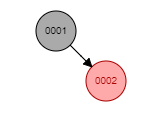
1. ערימה : כן, בדיוק כמו שבהכנסה והוצאה מיד של איבר בערמה הערמה לא תשתנה (משום שאנחנו מבצעים בדיוק את אותם החלפות גם בהכנסה וגם בהוצאה) כך זה קורה בדיוק כאן, לאחר הכנסה של a והכנסה של b ומיד הוצאה של b העץ יחזור בדיוק כמו שהוא היה לאחר הכנסה של a, ואז אפשר להתעלם מb שלא שינה כלום בעץ, והוצאה של a תהיה שקולה להכנסה ומיד הוצאה של a (משום שההכנסה והוצאה מיד של b לא שינתה כלום)

טבלת גיבוב עם פתרון התנגשויות: כן, הכנסה והוצאה של איברים בטבלת גיבוב לא משפיע על סדר או איברים אחרים בטבלה.

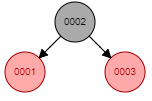
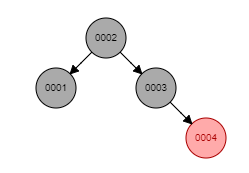
עץ חיפוש בינרי: כן, 2 האיברים שיכנסו בהכרח יהיו עלים, ולכן הכנסה ומחיקה שלהם לא תשפיע על שום סדר בעץ משום שאף איבר לא מתחלף.

עץ אדום שחור: לא, נראה דוגמא נגדית

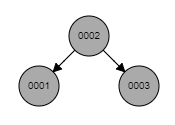
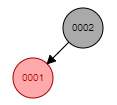
נניח עץ אדום שחור שבו 2 איברים 1,2 ונראה כך:



נכניס 2 איברים 3 ו4 בסדר עולה:

אחרי הכנסת 3:  אחרי הכנסת 4: 

כעת נמחק את 2 האיברים בסדר הגעתם:

אחרי מחיקת 4: אחרי מחיקת 3: 

וזה עץ שונה מהעץ הראשוני.

**שאלה 4:**

במקרה זה כשמורידים ערך C ממסלול בעץ חיפוש בינארי, לכל צומת יש אפשרות שהפרנו את התכונה של התת עץ השמאלי של צומת מהמסלול בגלל שאם החסרנו ערך והמסלול עובר דרך הבן הימני, התכונות מולו נשמרות כי הוא קטן בצורה דומה, בעוד התכונות מול בן השמאלי או יותר נכון מול כל תת העץ הימני יכולות להיות מופרות, כלומר כל האיברים שמשמאל נמצאים במקום בעייתי ביחס לאבא וגם ביחס לאיברים מהמסלול שנמצאים בתת עץ הימני, ולכן נצטרך לעבור על המסלול מהעלה לשורש ובמידה ונזהה צומת שבא מתבצעת הפרה מול התת עץ השמאלי, נעבור על כל הצמתים בתת עץ השמאלי, נמחק אותם ונוסיף אותם אחד אחד בחזרה לעץ.

המקרה הגרוע יהיה במידה והמסלול הוא מסלול ימני (שבו logn איברים לכל היותר) כך שמספר האיברים בתתי עץ השמאליים המפרים יכולים להיות לכל היותר

פעולת ההכנסה עבור כל אחד מהאיברים האלו מתבצעת ב

זמן ריצה: :

1. מעבר על המסלול p מהעלה לשורש ובכל פעם קריאה לmax-heapify ש"תפעפע" (למטה) את האיבר כך שהעץ יקיים וישמור על תכונות הערימה.

זמן ריצה:

1. במקרה של ערמת מינימום הערמה התכונות שלה תמיד ישמרו, משום שהתכונה ש- האב תמיד יהיה קטן מ2 בניו תמיד תתקיים.

במידה ומורידים ערך C קבוע מאב ומהבנים שלו, הפער ביניהם רק יכול לגדול (הפער בין האב לבן שלא במסלול שמחסירים ממנו) אך אף פעם לא יכול לקטן.

כלומר עבור כל איבר במסלול, ההפרש בין האבא שלו אליו יישאר קבוע, וההפרש בין אבא שלו לבן השני (אם קיים) יכול להיות רק יותר גדול, כלומר תכונות הערימה אף פעם לא יופרו, ולכן אין צורך באלגוריתם תיקון.

**שאלה 5:**

נבחר בעץ אדום שחור כשכל איבר בו יכיל את שם (name) זמן הרשמה ייחודי (reg-time) ומצב לימודים ((learn-status תקין/לא תקין.

ובנוסף שדה שיכיל את זמן ההרשמה המקסימלי (max-reg-time) של סטודנט בתת עץ הישיר שלו (כלומר הוא ובניו הישירים) שערכו יהיה:

במידה ומצב הלימודים תקין – זמן ההרשמה המקסימלי המתקבל מזמן ההרשמה שלו ושל הבנים הימני והשמאלי שלו.

אחרת כשמצב הלימודים לא תקין – זמן ההרשמה המקסימלי המתקבל מזמן ההרשמה של הבנים הימני והשמאלי שלו.

הערך החדש שמכיל את זמן ההרשמה המקסימלי של סטודנט בתת עץ הישיר שלו מחושב רק באמצעות האיבר והבנים הישירים שלו ולכן ע"פ משפט 14.1 בעמוד 259 בספר תחזוק שדה זה (בהכנסה ומחיקה לדוגמא) לא ישפיע לרעה על זמן הריצה.

הכנסת סטודנט למבנה:

*שמתואר בעמוד 236 בספר ועדכון השדה* max-reg-time

זמן ריצה: .

מחיקת סטודנט ע"פ מצביע:

*שמתואר בעמוד 242 בספר ועדכון השדה* max-reg-time

זמן ריצה: .

*החזרת סטודנט שנרשם אחרי x:*

*שמתואר בעמוד 218 בספר לעץ חיפוש בינארי ויעבוד באופן דומה גם בעץ אדום שחור*

זמן ריצה: .

*הרשמה תקינה הבאה אחרי זמן t:*

*תיאור האלגוריתם: נממש שגרה רקורסיבית שמזכירה מעט חיפוש איבר בעץ חיפוש בינרי, והפעם עם בדיקת השדה שהוספנו.*

*תחילה במידה ו* max-reg-time*של השורש גדול מt – לא קיים איבר העונה על הדרישה ויוחזר NIL*

*כעת אם קיים לאיבר הנוכחי בן שמאלי וגם השדה* max-reg-time *שלו גדול מt נבצע קריאה רקורסיבית לבן השמאלי.*

*אחרת אם מצב הלימודים של האיבר הנוכחי תקין וגם זמן ההרשמה שלו גדול מt נחזיר את האיבר הנוכחי*

*אחרת נבצע קריאה רקורסיבית לתת עץ הימני.*

*כאשר הקריאה תתבצע על ידי*