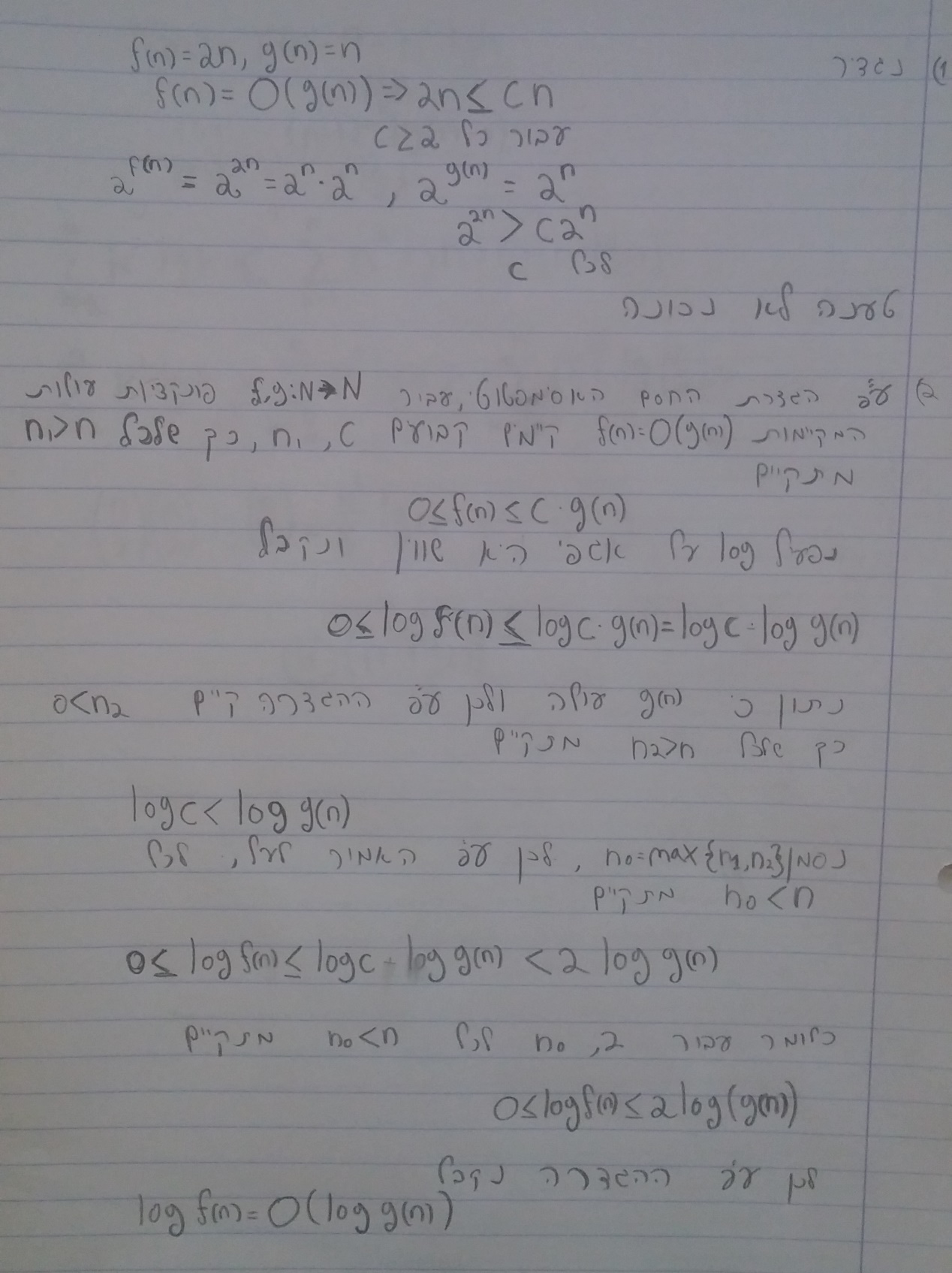
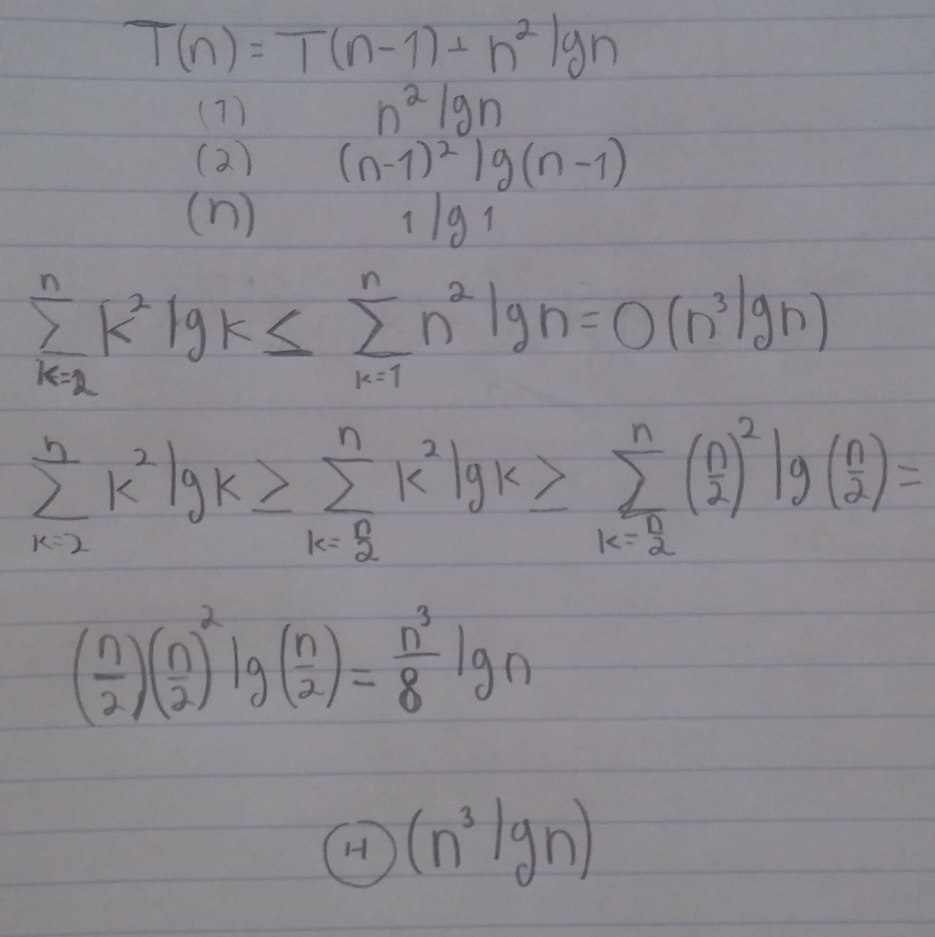
שאלה 1

א-



ב-



שאלה 2

הרעיון הוא לספור את ההיפוכים תוך כדי ביצוע השגרה MERGE

(תיאור של השגרה מופיע בעמודים 10-11 בספר הלימוד, ובתרגיל 1.3-2 בספר היה צריך לכתוב את הפסידו-קוד של השגרה(.

באופן כללי, בעת ביצוע השגרה MERGE משווים איברים מהתת-מערך השמאלי לאיברים

בתת-מערך הימני,ומציבים את הקטן מביניהם בתת-מערך הממוזג. כאשר משווים איבר *A*[*i*]

הנמצא בתת-מערך השמאלי לאיבר  *A*[*j*] הנמצא בתת-מערך הימני יש שתי אפשרויות:

אם *A*[*i*] ≤ *A*[*j*] אז זוג האינדקסים אינו מהווה היפוך. (*i*, *j*)

לעומת זאת, כאשר *A*[*i*] > *A*[*j*] אז זוג האינדקסים (*i*, *j*) מהווה היפוך. יתרה מזאת, גם זוג

האינדקסים (*x*, *j*) לכל *i +* 1 ≤ *x* ≤ *q* מהווה היפוך.

נוסיף לאלגוריתם מיון-מיזוג משתנה גלובלי בשם count שיאותחל לאפס.

כאשר *A*[*i*] > *A*[*j*] נוסיף לשגרה MERGE את השורה.count ← count + *q* - *i* + 1 :

מכיוון שהפעולה שהוספנו מתבצעת בזמן קבוע, זמן הריצה של מיון-מיזוג לא משתנה.

שאלה 3

א –

מבנה הנתונים היה מחסנית אשר נוסף לה השדה nextmin. ומשתנה נוסף d.

- insert : מהספר, טריוויאלי. כמו PUSH.

-minKey : בכל פעם שמוספים איבר חדש אשר קטן מהמינימום הנוכחי, מוספים את (מצביע) המינימום הנוכחי לשדה nextmin באיבר החדש, ומעדכן את הmin הנוכחי להצביע לאיבר החדש.

-deleyeLast: מבוסס על פעולת המחיקה שבספר, בתוספת של בזמן מחיקת הmin צריך להגדיר את min, להצביע על nextmin ואז למחוק.

mult-: האיבר הנוסף,d, יאותחל ב1, ובכל שימוש של MULT, הוא יוכפל בעצמו.

בזמן הכנסה, האיבר הנכנס יחולק בd.

באמת החזרה האיבר המוחזר יוכפל בd.

ב – לא אפשרי, מכיוון שאנחנו לא ידועים מה האיבר הנכנס, לא נוכל לא להסיק כי הוא ממוין, מכך בכל פעולת הכנסה נצטרך לבצע HEAPIFY, ולכן עלות ההכנסה תעלה.

גם אם נצור מבנה נתונים נוסף ונקשר בנהיים, כדי שהמבנה הנותנים הנוסף גם היה ממוין, עלות ההכנסה שלו גם תהיה גדול יותר מ O(1).

שאלה 4

1. מערך לדוגמה **{ 20, 10, 5, 1, 7, 15, 30, 25, 35, 32, 40 }**

האיבר הראשון במערך הוא השורש של העץ, טריוויאלי מתוכנות סריקה תחילית, במקרה זה 20,נגדיר אותו כשרוש העץ, וניצור שני טווחים, מינימום ומקסימום אשר יתחמו את הבנים שלו השורש, כאשר באתחול הראשון המינימום יאתחל להיות המינימום הכי קטן שנתמך והמקסימום יאתחל להיות מקסימום הכי גדול שנתמך, כדי שהשורש יכול לקבל כל ערך בטווח האפשרי.

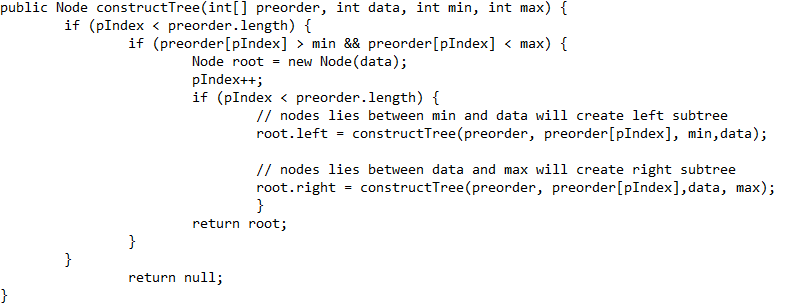
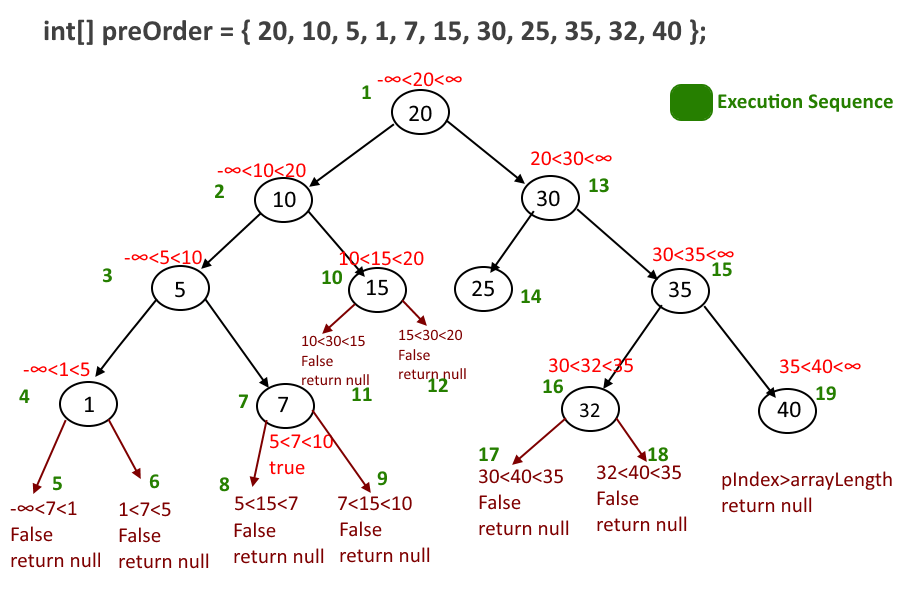
כאשר נגדיר את הבן השמאלי של הצומת (לדוגמה השורש 20), צרכים להגדיר שערך המקסימלי שלו היה איבר הצומת (במקרה זה 20).

* אם האיבר הבא במערך נמצע בטווח הזה של המערך, נגדיר אותו להיות הבן השמאלי.
* אחרת, הוא חייב להיות הבן הימני, וכך אלה

נעבור על זה באופן רקורסיבי.

1. לא, בעץ בינארי רגיל, לדוגמה 7 יכול להיות הבן השמאלי של הערך 1, לכן לא ניתן היה לדעת את מיקמו המדויק בעץ.

דוגמה לקוד של שאלה 4-א



שאלה 5

נשתמש בעץ אדום שחור מורחב, כל הפרטים טריוויאליים.