תיעוד תרגיל מעשי 2-מבני נתונים

המגישים:

שם: עמיד גאנם. ת.ז_315525311

שם: נור חג'. תז_314997602

ניתוח זמן ריצה של ההפונקציות:

arrayToHeap_הסיבוכיות היא O(n).

hasChild_הסיבוכיות היא O(1).

isHeap_הסיבוכיות היא O(n*d) כי משווים כל צמת עם d הבנים שלו.

Parent_(O(1 כי פשוט מחשבים ומחזירים אינדיקס.

Child_(O(1 כי פשוט מחשבים ומחזירים אינדיקס.

Insert_(O(logn כי משווים האיבר שהכנסנו לערימה לאבא שלו וממשיכים עד שלא יהיה תנאי הערימה מופר.

Heapify_up_(O(logn נשווה האיבר לאבא ואז ממשיכים לכל היותר log(n) פעמים עד השורש.

swapWithParent_(O(1 כי משנים שני מצבעים ומעדכנים שני שדות(השדות של pos) לכן מס' קבוע של פעולות ולכן נקבל O(1).

Delete_Min_(O(d*$\log_d$n) כי קוראים לפונקציה heapify_down שאני אנתח אותה בהמשך.

Heapify_down_(O(d*$\log_d$n) קוראים לפונקציה minChild_numOfComps כדי לקבל האינדיקס של הבן המינימלי, וזה לוקח O(d) ואז אם הבן קטן מאבא אז מחליפים, ממשיכים באופן הזה עד שלא מפירים תנאי ערימה או עד שנגיע לעלה, העומק הוא O($\log_d$n) לכן בסה"כ O(d*$\log_d$n).

minChild_numOfComps_(O(d עוברים בלולאה על הבנים של צמת נתון, בודקים כל פעם מי גדול מהבן הראשון ואז מעכנים את המינימום. זה לוקח O(1) זמן ויש d איטירציות לכן סה"כ O(d).

Get_Min_(O(1 כי פשוט מחזירים מצבע ל-array[0].

Decrease_Key_(O($\log_d$n) כשמקטינים את ערכו של ה-Item אז אולי הפרנו את תנאי הערימה לכן קוראים ל-heapify_up ואז סה"כ לוקח O($\log_d$n).

Delete_(O((d+1)*$\log_d$n) קוראים ל-Decrease-key שלוקחת O($\log_d$n) ואז ל-Delete-Min שלוקחת(O(d*$\log_d$n לכן סה"כ O((d+1)*$\log_d$n).

DHeap_Sort_(O(n*d*$\log_d$n) כי קודם מעדכנים השדה array שיכיל Items עם key מתאים לפי המערך הנתון ואז קוראים ל- arrayToHeap שלוקחת O(n) ואז עושים n פעמים Get_Min ו- Delete_Min (בלולאה) לכן סה"כ:

O(n+n+(n*d*$\log_d$n)) = O(n*d*$\log_d$n).

מדידות:

```
1st stage (DHeapSort):
        m=1,000:
                d=2: 16281
                d=3: 16149
                d=4: 17115

        m=10,000:
                d=2: 229530
                d=3: 222918
                d=4: 225718

        m=100,000:
                d=2: 2947933
                d=3: 2807723
                d=4: 2867315
2nd stage (DecreaseKey):
        x=1:
                d=2: 100000
                d=3: 100000
                d=4: 100000
        x=100:
                d=2: 152931
                d=3: 130739
                d=4: 123057
        x=1000:
                d=2: 299686
                d=3: 211147
                d=4: 180057
```