**תרגיל בית 1 מבני נתונים**

**id1: 322520255**

**name1: Itamar Ben Nun**

**username1: itamarbennun**

**id2: 316061787**

**name2: Tal Malka**

**username2: talmalka2**

**חלק מעשי**

Class AVLNode

|  |  |  |
| --- | --- | --- |
| פעולה | תיאור | סיבוכיות |
| \_\_init\_\_() | בנאי – יוצר צומת ריק. |  |
| is\_real\_node() | בודק האם צומת הוא וירטואלי או לא ע"י בדיקה של המפתח. |  |
| search\_helper(k) | מחפש מפתח בתת העץ של הצומת עליה פועל, ומחזיר את מספר הקשתות של החיפוש +1, ואת הצומת – או בן וירטואלי אם היא לא קיימת. |  |

Class AVLTree

|  |  |  |
| --- | --- | --- |
| פעולה | תיאור | סיבוכיות |
| \_\_init\_() | בנאי – יוצר עץ ריק |  |
| search(k) | מחפש מפתח בעץ שעליו פועל, ומחזיר את מספר הקשתות של החיפוש +1, ואת הצומת – או None אם היא לא קיימת. משתמש בsearch\_helper. |  |
| finger\_search(k) | מחפש מפתח בעץ שעליו פועל החל מהמקסימום, ומחזיר את מספר הקשתות של החיפוש +1, ואת הצומת – או None אם היא לא קיימת. משתמש ב-finger\_search\_helper. |  |
| finger\_search\_helper(k) | מחפש מפתח בעץ שעליו פועל החל מהמקסימום, ומחזיר את מספר הקשתות של החיפוש +1, ואת הצומת – או בן וירטואלי אם היא לא קיימת. משתמש בsearch\_helper. |  |
| successor(x) | מחזיר את היורש של צומת כלשהו. |  |
| insert(k, v) | מכניס צומת לעץ, משתמש בsearch\_helper על מנת למצוא את מקום ההכנסה, מבצע איזון לעץ במידת הצורך ומעדכן מינימום, מקסימום, וגודל. |  |
| rebalance(x) | מבצע איזון לעץ החל מהצומת שמקבל. |  |
| rotate\_left(x) | מבצע גלגול שמאלה – החלפת פוינטרים. |  |
| rotate\_right(x) | מבצע גלגול ימינה – החלפת פוינטרים. |  |
| finger\_insert(k, v) | מכניס צומת לעץ החל מהמקסימום, משתמש ב-finger\_search\_helper על מנת למצוא את מקום ההכנסה, מבצע איזון לעץ במידת הצורך ומעדכן מינימום, מקסימום, וגודל. |  |
| delete(x) |  |  |
| join(t, k, v) |  |  |
| split(x) |  |  |
| set\_min() |  |  |
| set\_max() |  |  |
| avl\_to\_array() |  |  |
| max\_node() |  |  |
| size() |  |  |
| get\_root() |  |  |
| insert\_root(k, v) |  |  |
| insert\_de\_facto(x, e, k, v) |  |  |

**חלק ניסויי**

1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מס"ד** | **עלות איזון במערך ממוין** | **עלות איזון במערך ממוין-הפוך** | **עלות איזון במערך מסודר אקראית** | **עלות איזון במערך עם היפוכים סמוכים אקראיים** |
| **1.** | 430 | 430 | 383.05 | 423.95 |
| **2.** | 873 | 873 | 776.2 | 867.3 |
| **3.** | 1760 | 1760 | 1570.4 | 1738.35 |
| **4.** | 3535 | 3535 | 3162.1 | 3498.05 |
| **5.** | 7086 | 7086 | 6329.25 | 7016.8 |
| **6.** | 14189 | 14189 | 12653.7 | 14037.9 |
| **7.** | 28396 | 28396 | 25338.85 | 28056.95 |
| **8.** | 56811 | 56811 | 50719.1 | 56153.5 |
| **9.** | 113642 | 113642 | 101372.7 | 112336.65 |
| **10.** | 227305 | 227305 | 202890.7 | 224675.3 |

כפי שלמדנו בשיעור, החסם העליון למספר הגלגולים *קטן/שווה 2, החסם העליון למספר ה-*promotions *הוא ולכן מספר גלגולים זניח. כלומר החסם העליון התיאורטי על סך עלויות האיזון הוא .*

2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מס"ד** | **מספר היפוכים במערך ממוין** | **מספר היפוכים במערך ממוין-הפוך** | **מספר היפוכים במערך מסודר אקראית** | **מספר היפוכים במערך עם היפוכים סמוכים אקראיים** |
| **1.** | 0 | 24531 | 12237.55 | 110.9 |
| **2.** | 0 | 98346 | 49575.25 | 221.3 |
| **3.** | 0 | 393828 | 195797.9 | 444.6 |
| **4.** | 0 | 1576200 | 786370.55 | 888.1 |
| **5.** | 0 | 6306576 | 3156659.35 | 1781.3 |

3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מס"ד** | **עלות חיפוש במערך ממוין** | **עלות חיפוש במערך ממוין-הפוך** | **עלות חיפוש במערך מסודר אקראית** | **עלות חיפוש במערך עם היפוכים סמוכים אקראיים** |
| **1.** | 222 | 2695 | 2215.15 | 287.65 |
| **2.** | 444 | 6273 | 5251.0 | 581.05 |
| **3.** | 888 | 14317 | 12371.15 | 1159.15 |
| **4.** | 1776 | 32181 | 28420.45 | 2330.3 |
| **5.** | 3552 | 71461 | 64050.55 | 4657.6 |
| **6.** | 7104 | 157125 | 142773.7 | 9313.95 |
| **7.** | 14208 | 342661 | 310491.3 | 18640.2 |
| **8.** | 28416 | 742149 | 682672.75 | 37316.45 |
| **9.** | 56832 | 1597957 | 1475418.7 | 74597.15 |
| **10.** | 113664 | 3423237 | 3167710.1 | 149222.45 |

4.

I. לכל ,  *מסמן את מספר האיברים לפני האינדקס ה- כך שלכל מתקיים , כלומר הגדרת ההיפוך. ולכן סך ההיפוכים הכולל הוא סכום ההיפוכים לכל איבר.*

II. בfinger\_search- מתחילים מהאיבר המקסימלי, עד שמגיעים לשורש או לצומת עם מפתח קטן/שווה מהמפתח שאנו מחפשים, והתת-עץ של הצומת הזה מכיל את כל האיברים שגדולים מהמפתח ה-, כלומר  *איברים. הגובה של העץ הזה חסום ע"י , ובהתאם גם עלות ההכנסה של המפתח ה-. במקרה שבו עלות ההכנסה היא , ולכן עבור סדרת הכנסות של איברים, עלות ההכנסה היא:*

III. לפי א"ש הממוצעים מתקיים:

*הביטוי מייצג את ממוצע ההיפוכים, כלומר ה- הממוצע עבור האיבר ה-, לכן עלות החיפוש הממוצעת היא כפי שראינו בסעיף* II*, תהליך ההכנסה של האיבר נחלק ל-2, כאשר בשלב הראשון מחפשים את תת-העץ שמכיל את האיברים, שהוא בממוצע , ולאחר מכן מחפשים בתת-העץ הזה, כלומר העלות לוגריתמית ביחס למספר האיברים בו - , ולכן עבור סדרה של הכנסות, העלות היא .*

IV. נבחין כי במערך ממוין, מתקיים , ולכן עלות החיפוש היא , שמשקף את תוצאות הניסוי בעמודה הימנית.

במערך הפוך, מתקיים *, ולכן עלות החיפוש היא , מתנהג אסימפטוטית דומה לתוצאות העמודה השנייה עד כדי קבוע .*

במערך אקראי, מספר ההחלפות הוא כחצי ממערך הפוך, ולכן תוצאות הניסוי דומות לשל מערך הפוך.

במערך בו יש חילופים אקראיים, מספר ההחלפות דומה למספר האיברים, ולכן תוצאות הניסוי דומות לשל מערך ממוין.