**צומת**

פונקציות מחלקה:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| אינדקס | פונקציה | תיאור הפונקציה | פונקציות עזר בשימוש | סיבוכיות זמן ריצה | הסבר לסיבוכיות |
|  | \_\_init\_\_(self, key, value) |  |  |  | O(1) |
|  | get\_key(self) |  |  |  | O(1) |
|  | get\_value(self) |  |  |  | O(1) |
|  | get\_left(self) |  |  |  | O(1) |
|  | get\_right(self) |  |  |  | O(1) |
|  | get\_parent(self) |  |  |  | O(1) |
|  | get\_height(self) |  |  |  | O(1) |
|  | get\_size(self) |  |  |  | O(1) |
|  | set\_key(self, key) |  |  |  | O(1) |
|  | set\_value(self, value) |  |  |  | O(1) |
|  | set\_left(self, node) |  |  |  | O(1) |
|  | set\_right(self, node) |  |  |  | O(1) |
|  | set\_parent(self, node) |  |  |  | O(1) |
|  | set\_height(self, h) |  |  |  | O(1) |
|  | set\_size(self, s) |  |  |  | O(1) |

פונקציות עזר:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| אינדקס | פונקציה | תיאור הפונקציה | פונקציות עזר בשימוש | סיבוכיות זמן ריצה | הסבר לסיבוכיות |
| 1 | create\_leaf\_with\_virtual\_nodes(key, value) | יוצר עלה בעץ שבניו וירטואליים |  | O(1) |  |
| 2 | compute\_balance\_factor(self) | מחשב את הBF של הצומת |  | O(1) |  |
| 3 | is\_empty\_node(node) | בודק האם הצומת היא None או צומת וירטואלי |  | O(1) |  |
| 4 | is\_leaf(self) | בודק האם הצומת הוא עלה |  | O(1) |  |
| 5 | has\_one\_child(self) | בודק האם לצומת ילד אחד בלבד |  | O(1) |  |
| 6 | is\_left\_child\_of\_parent(node) | בודק האם הצומת היא הבן השמאלי האב שלו |  | O(1) |  |
| 7 | update\_parents\_child(self, old\_child, new\_child) | מחליף בין old\_child לnew\_child ביחס לאב של old\_child |  | O(1) |  |
| 8 | is\_real\_node(self) | בודק אם הצומת הוא "אמיתי" או וירטואלי |  | O(1) |  |

**עץ**

פונקציות מחלקה:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| אינדקס | פונקציה | תיאור הפונקציה | פונקציות עזר בשימוש | סיבוכיות זמן ריצה | הסבר לסיבוכיות |
|  | \_\_init\_\_(self) |  |  |  |  |

פונקציות עזר:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| אינדקס | פונקציה | תיאור הפונקציה | פונקציות עזר בשימוש | סיבוכיות זמן ריצה | הסבר לסיבוכיות |
| 9 | set\_root(self, node) |  |  | O(1) |  |
| 10 | rotate\_right(self, old\_root: AVLNode) |  | 7 | O(1) |  |
| 11 | rotate\_left(self, old\_root: AVLNode) |  | 7 | O(1) |  |
| 12 | rotate\_left\_then\_right(self, old\_root: AVLNode) |  | 10,11 |  |  |
| 13 | rotate\_right\_then\_left(self, old\_root: AVLNode) |  | 10,11 |  |  |
| 14 | find\_successor\_for\_node\_with\_two\_childs(self, node: AVLNode) |  | 3 | O(logn) כגובה העץ |  |
| 15 | find\_parent\_for\_insert(self, key) |  | 8 | O(logn) כגובה העץ |  |
| 16 | find\_parent\_with\_illegal\_balance\_factor(self, node: AVLNode) |  | 2 | O(logn) כגובה העץ |  |
| 17 | fix\_tree\_of\_illegal\_root(self, illegal\_root) |  | 2, 10, 11, 12, 13 | O(1) |  |
| 18 | fix\_tree(self, node\_with\_illegal\_balance\_factor) |  | 16, 17 | O(logn) כגובה העץ |  |
| 19 | physical\_insert(self, leaf\_for\_insert: AVLNode) |  | 15 | O(logn) כגובה העץ |  |
| 20 | physical\_delete(self, node: AVLNode) |  | 8, 6 | O(1) |  |
| 21 | replace\_node\_in\_tree(self, old\_node: AVLNode, new\_node: AVLNode) |  | 9, 7 | O(1) |  |
| 22 | avl\_to\_array\_rec(self, node: AVLNode, array) |  | 3, 22 | O(n) |  |
| 23 | create\_tree(self, root) |  | 9 | O(1) |  |
| 24 | join\_trees\_with\_equal\_heights(self, t1, t2, x) |  |  | O(1) |  |
| 25 | join\_trees\_left\_tree\_is\_smaller(self, t1, t2, x) |  | 18 | O(logn) כגובה העץ |  |
| 26 | join\_trees\_right\_tree\_is\_smaller(self, t1, t2, x) |  | 18 | O(logn) כגובה העץ |  |
| 27 | join\_tree\_with\_array\_of\_node\_tuples(self, array\_of\_node\_tuples) |  | קורא פונקציה join, 23 | O(logn) כגובה העץ |  |

פונקציות שלהם:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| אינדקס | פונקציה | תיאור הפונקציה | פונקציות עזר בשימוש | סיבוכיות זמן ריצה | הסבר לסיבוכיות |
| 1 | search(self, key) |  | 8 | O(logn) כגובה העץ |  |
| 2 | insert(self, key, val) |  | 1, 9, 18, 19 | O(logn) כגובה העץ |  |
| 3 | delete(self, node: AVLNode) |  | 4, 5, 20, 14, 21, 18 | O(logn) כגובה העץ |  |
| 4 | avl\_to\_array(self) |  | 22 | O(n) |  |
| 5 | size(self) |  |  | O(1) |  |
| 6 | join(self, tree, key, val) |  | 1, 8, 24, 25, 26 | O(logn) כגובה העץ |  |
| 7 | split(self, node: AVLNode) |  | 23, 6, 27 | O(logn) כגובה העץ |  |
| 8 | rank(self, node) |  |  | O(logn) כגובה העץ |  |
| 9 | select(self, i) |  |  | O(logn) כגובה העץ |  |
| 10 | get\_root(self) |  |  | O(1) |  |