

עץ סגמנט - Segment Tree

אינטואיציה

נניח ונתון לנו לכל אורך L , את מספר הסדרות באורך L המסתיימות בערך x .
אז כאשר נמשיך לערך הבא במערך arr , העדכון תלוי בידיעת מספר הסדרות באורך $L-1$ המסתיימות ב-
 $x < y$

אלגוריתם

תחילה נגדיר "data" של קטע מסוים (RangeData), ובו נתונים האורך הגדול ביותר עבור תת סדרה עולה בקטע זה, ומספר תתי הסדרות העולות הארוכות ביותר בקטע זה.

נגדיר קודקוד (Range) שמכיר את ערכי המערך arr בעזרת השדות $[range_left, range_right]$ ונתון בו גם שדה RangeData המכיל את המידע של הקטע.
לקודקוד יש גם ילדים בשם $left$, $right$ המייצגים את החצי הימני והשמאלי של הקטע.
הילדים של הקודקוד נוצרים לפי דרישות הצורך.

כעת, השיטה $query(range, key)$ תחזיר לו את הקטע בהינתן קודקוד Range. כאשר המפתח key נמצא מחוץ לטווח הקטע של $range$, אנחנו נחזיר את התשובה.
אחרת - אנחנו נחצה את הקטע ל-2 קטעים ולבסוף נמזג את התוצאה.

מה עושה $merge(r1, r2)$ - ? נניח ששני צמתים מציינים קטעים סמוכים ויש לכל אחד מהם ערכים, כלומר: $r1 = range1.data, r2 = range2.data$.
אז מה יהיה הערך $r = merge(r1, r2)$ - ? אם יש כמה תתי סדרות ארוכות ביותר בקודקוד אחד, אז r יהיה מספר תתי הסדרות האלה. אם ל-2 הקודקודים יש תתי סדרות הארוכות ביותר באורך שווה ביניהם, אז אנחנו צריכים לספור "count" את תתי-הסדרות הללו ב-2 הקודקודים.
נשים לב שלא התייחסנו למקרים בהם נוצרו תתי סדרות ארוכות ביותר מאחר והן מטופלות בשיטה $insert$.

מה עושה $insert(range, key, data)$ - ? אנחנו באופן מחזורי נכניס את ה- key לחצי המתאים לו בקטע בבעלות ה- $range$ ואז אחרי כל הכנסה, ה- $data$ של $range$ המיוחס יכול להשתנות, אז נמזג את ה- $data$ שוב יחד.

לבסוף, באלגוריתם הראשי, לכל מספר במערך arr , אנחנו נבצע $query$ ל- $data$ בעל $length$ ו- $count$ של הקטע המיוחס למספר במערך, וזה יוביל אותנו לספירה "count" של סדרות נוספות באורך $length+1$.
וכך נעדכן את העץ עם הנתונים החדשים.