

מוכרו, ו-הנורא מושג ב- X -ו של ערך-

`mid` \leftarrow `getMid`(`arr`, `low`, `high`). *f.f*

$A[\text{mid}] < x$ and $A[\text{mid} + 1] = x$ $\sim 1/2$
 $\text{mid} + 1 - 1/3D$

When this code runs $A[\text{mid}] = x$ at line 1.3

ਇਸ ਕਾਨੂੰਨ ਦੀ ਵਰਤੋਂ ਸੁਣੋ, ਅਤੇ 1.4

፳፻፲፭ ዓ.ም. በ፲፻፲፭ ዓ.ም. ከ፩፭፻፭ ዓ.ም. በ፩፭፻፭ ዓ.ም.

[40, 40, 1, 5, 17, 1] .1

[40, 40, 1] .2

[40, 1] .3

[1, 40] 4

$$[1, 40, 40] \quad 5$$

[5, 17, 1] 6

$$[17, 1] \quad 7$$

[1, 17] 8

[1, 5, 12] 9

$$[1, 1, 5, 17, 40, 40] \cdot 10$$

C 317 2"2N ②

$$a[] = [40, 40, 1, 5, 17, 1] \quad ③$$

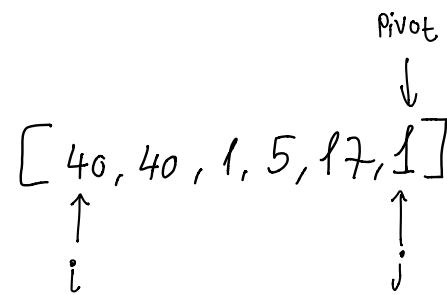
Sort-mergejoin

• የዕለታዊ የሃይል ቤት ስራውን በት ጥሩ

עֲדָם יְמִינָה וְלִפְנֵי נֶשֶׁךְ יְמִינָה

விளைவுகள்

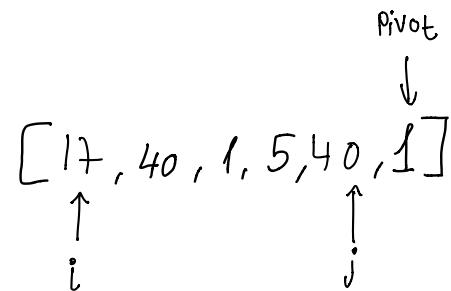
רְאֵת אֶת־בָּנָי בַּיּוֹם
הַז (4)



$$A[i] = 40 \quad A[j] = 1$$

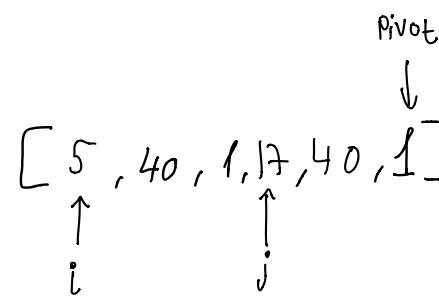
$j-- \Leftarrow \text{Pivot} - f \text{ will be } A[j]$

$$A[j] \rightarrow A[i] \quad i < j$$



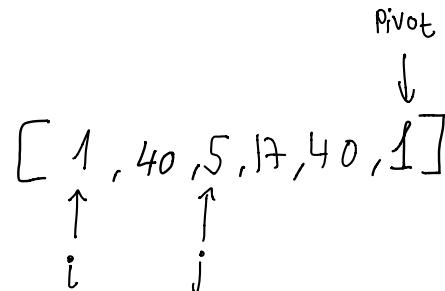
$j^- \in \text{pivot_set}[i]$

$$A[j] \rightarrow A[i] \text{ if } i < j$$



$j^- \in \text{Pivot-Set } \exists i \in A[j]$

$A[j] \rightarrow A[i]$ if $i < j$



$\stackrel{\circ}{j} + \leftarrow \text{pivot} - f \text{ rre } A[i]$

$$A[j] \rightarrow A[i] \text{ if } i < j$$

Pivot
|

[1, 5, 40, 17, 40, 1]

Two hand-drawn black arrows pointing upwards from a horizontal baseline. The arrow on the left is taller than the one on the right.

$j-- \Leftarrow \text{Pivot } - f \text{ in row } A_{ij}$

$$A[j] \rightarrow A[i] \quad f.d_{ij} \quad i < j$$

[1, 5, 40, 17, 40, 1]

↑
j,i

[1, 1, 40, 17, 40, 5]

[40, 17, 40, 5])

\uparrow
 P , j

$j-- \Leftarrow \text{Pivot} - f \text{ nilai } A[j]$

$$A[j] \rightarrow A[i] \text{ for } i < j$$

A stack structure with elements [40, 17, 40, 5]. An arrow labeled *i* points to the first 40, and an arrow labeled *j* points to the second 40.

[5, 17, 40, 40]

۱۱

$$[1, 1, 5, 17, 40, 40]$$

6

$j-- \Leftarrow \text{Pivot} - f \text{ will be } A[j]$

$$A[j] \rightarrow A[i] \text{ if } i < j$$

[40, 17, 40, 5])

Three black arrows pointing upwards, positioned at the top of the page.

: merge סדרה של אוסף און

$$\begin{aligned} p_1 &= \text{lst1} \rightarrow \text{tail} \\ p_2 &= \text{lst2} \rightarrow \text{tail} \end{aligned}$$

Count - (Count == 4) \in 3rd condition 2. ריבועי פולינום ב (lst1, lst2)

While (Count != 4)

: True אם $p_1 \rightarrow \text{key} < p_2 \rightarrow \text{key}$ 2.1

אם $p_1 \rightarrow \text{key}$ ו ערך המצביע על $p_1 \rightarrow \text{key}$ מוגבר Count++

Count++ אם לא $p_1 \rightarrow \text{key} \neq \text{prev}$

$$p_1 = p_1 \rightarrow \text{next} \quad \text{אם לא}$$

ForthLargest רקורסיבית הינה שורה של נードים

: false אם לא $p_1 \rightarrow \text{key} < p_2 \rightarrow \text{key}$

אם $p_2 \rightarrow \text{key}$ ו ערך המצביע על $p_2 \rightarrow \text{key}$ מוגבר Count++

Count++ אם לא $p_2 \rightarrow \text{key} \neq \text{prev}$

$$p_2 = p_2 \rightarrow \text{next} \quad \text{אם לא}$$

ForthLargest רקורסיבית הינה שורה של נードים

Collision detection הינה שורה של נードים שפה שפה

$O(n)$ וריאנט ואנומלי

: 6 ନଫିରେ

target ଲାଗ୍ଜିଂ କରିବାକାରୀ

```
void FourthLargest(const CLIST* A, const CLIST* B)
{
    DATA FourthLargest;
    int count = 0;
    DATA prev = 9999;
    NODE* p1, * p2;
    p1 = A->head.next;
    p2 = B->head.next;
    while(count!=4)
    {
        if (p1->key > p2->key)
        {
            if (prev != p1->key)
                count++;
            prev = p1->key;
            FourthLargest = p1->key;
            p1 = p1->next;
        }
        else
        {
            if (prev != p2->key)
                count++;

            prev = p2->key;
            FourthLargest = p2->key;
            p2 = p2->next;
        }
    }
    printf("\nFourthLargest = %d\n\n", FourthLargest);
}
```

```
Microsoft Visual Studio Debug Console

List1 :
25, 20, 18, -10, -10,

List2:
90, 20, 19, 1, -10,
FourthLargest = 19

C:\Users\asus\OneDrive\Desktop\CLIST\
```