Debugging parallel programs is simpler than normal sequential programs.

F

In order to solve the maximum finding problem by a parallel algorithm with $T(n) = O(1)$, we need work load $W(n) = \Omega(n^2)$ in return.

F

Recall the discussion about the Maximum Finding Problem (that is, to find the maximum among n numbers in an array), Common CRCW memory strategy is used to assure $T(n) = O(1)$ for the parallel algorithm. Actually, we can also apply Arbitrary CRCW memory strategy to keep $O(1)$ time complexity. Now let us consider a new memory strategy, namely the **C**oncurrent **R**ead **O**wner **W**rite (CROW). It means that each memory has an official "owner". Only the "owner" can write to the corresponding memory. Then there is no parallel algorithm that can solve the problem with $T(n) = O(1)$ using CROW memory strategy.

◉ T ○ F

答案正确：2 分 创建提问

The Merging problem is to merge two non-decreasing arrays A(1), A(2), ..., A(n) and B(1), B(2), ..., B(n) into another non-decreasing array C(1), C(2), ..., C($2n$). To solve it in parallel, we turn it into a Ranking problem. That is, to compute RANK(A(i),B) and RANK(B(i),A) for every $1 \leq i \leq n$, where RANK(e,S) is the position of e in S. The following psuedo-code is for solving the Ranking problem parallely.

```
for Pi, 1<=i<=n  pardo
    RANK(A(i),B) := BS(A(i),B)
    RANK(B(i),A) := BS(B(i),A)
```

where `BS(e,S)` is to find the position of `e` in `S` by **binary search**.

Which of the following gives the time and work load of the algorithm?

◉ A. $T(n) = O(\log n), W(n) = O(n \log n)$

○ B. $T(n) = W(n) = O(n \log n)$

○ C. $T(n) = W(n) = O(n)$

○ D. $T(n) = O(\log n), W(n) = O(n)$

答案正确：3 分 创建提问