Task 05

(Index of arrays starts from 1 as is given by the problem.)

We need an intermediate array C for this task, where

$$C[k] := \sum_{i=1}^{k} A[i].$$

As $A[]$ is a boolean array consisting of 0s and 1s, it is easy to notice, if $A[k] == 1$, $B[C[k]]$ should be k.

Note that the LHS of the formula above is in a form of prefix sum, the main task becomes computing the prefix sum in $O(\log n)$.

Prefix Sum Steps:

1. Fill B with undefined;
2. Split A into $\log n$ parts. $A_k := A[k \log n + 1, .., (k+1) \log n]$; Denote B_k, C_k in the same way;
3. Do in parallel $C_k \leftarrow \text{sequential\_prefix\_sum}(A_k)$; ($O(\log n)$)
4. Let $\text{last}(C_k) :=$ the last element of $C_k, \text{offset}_k := \sum_{i=1}^{k} \text{last}(C_k)$;
5. Do in parallel $C_k \leftarrow C_k + \text{offset}_{k-1}$; ($O(\log n)$)

The total time complexity of Prefix Sum Steps is $O(\log n)$.

Compute Final Solution:

Do in parallel $B[C[j]] \leftarrow j$, if $A[j] == 1$. ($O(\log n)$)

All in all, the overall time complexity is $O(\log n)$.■

An Example of How It Works:

$$A = \{0, 1, 1, 1, 0, 1\}, n = 6, p = n/(\log n) \approx 3.$$

1. $A_1 = \{0, 1\}, A_2 = \{1, 1\}, A_3 = \{0, 1\}$;
2. $C_1 = \{0, 1\}, C_2 = \{1, 2\}, C_3 = \{0, 1\}$;
3. $\text{offset}_1 = 1, \text{offset}_2 = 1 + 2 = 3$;
4. $C_2 \leftarrow C_2 + \text{offset}_1 = \{2, 3\}$;
5. $C_3 \leftarrow C_3 + \text{offset}_2 = \{3, 4\}$;
6. The complete C is $\{0, 1, 2, 3, 3, 4\}$;
7. Compute the final solution $B \leftarrow \{2, 3, 4, 6, \text{undefined}...\}$.