

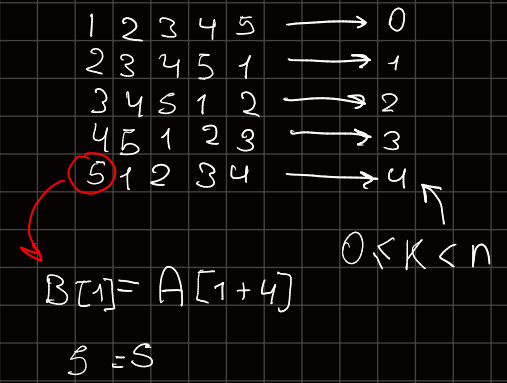
Ejercicio 8. Given an array $A[1..n]$, a k -rotation of A is an array $B[1..n]$ such that

$$B[i] = \begin{cases} A[i+k] & i+k \leq n \\ A[(i+k) \bmod n] & \text{otherwise} \end{cases}$$

For example, if $A = [3, 6, 9, 10]$, a 2-rotation of A is $B = [9, 10, 3, 6]$.

Consider the following problem. Input: A k -rotation B of an array sorted in ascending order of distinct elements. Output: The number k . For example, if $B = [9, 10, 3, 6]$, the algorithm should return the value 2.

Design a $\Theta(\lg n)$ worst-case time algorithm for the problem. Write the pseudocode of the algorithm. Write a recurrence for the worst-case of this algorithm. Verify with the master theorem.



Algo (B, l, r) // Devuelve la rotación de $B[l, r]$

if ($l == r$)
return 0

$m = \lfloor \frac{l+r}{2} \rfloor$

if $A[l] < A[m+1]$

return Algo($B, m+1, r$)

← coincide

else

return Algo(B, l, m)

+ $\frac{r-l+1}{2}$

← desfase

