Hyp. Let P(n) be the assertion that circularBinarySearch works for all inputs right-left+1 = n

Base Case: when n=1 then right=left=m so t is found in A[m] as the function works correctly.

Inductive Step: We assume circularBinarySearch works as long as right-left+1 ≤ k and must prove right-left+1 ≤ k+1 in 3 cases: A[m]=t, A[m]>t, A[m]<t.

Case 1: A[m]=t
As the function works this will return m, the index of t.

Case 2: A[m]>t
This half of the array is either sorted or circularly sorted. Thus t must lie to the left of m. If the recursive call works correctly so shall this one. So n = (m-1)-left+1 = $\lfloor \frac{right+left}{2} \rfloor$ - left + 1 if right+left is even then n = $\frac{right+left}{2} - \frac{2left}{2} - 1 + 1 = \frac{right-left}{2} \leq$ right-left+1. When right+left is odd n = $\frac{right+left-1}{2} - \frac{2left}{2} - 1 + 1 = \frac{right-left-1}{2} \leq$ right-left+1. As right-left+1 ≤ k then the recursive call is made on interval 0 to k and is correct by Hyp. Thus circularBinarySearch must also work for interval k+1.

Case 3: A[m]<t
This is symmetrical to the previous case. Thus n = right-(m+1)+1 = right-($\lfloor \frac{right+left}{2} \rfloor$ +1)+1 so when right+left is even n = right - $\frac{right-left}{2} - 1 + 1 = \frac{2right}{2} - \frac{right-left}{2} = \frac{right+left}{2}$. When odd n = $\frac{right}{2} - \frac{right-left+1}{2} - 1 + 1$ = $\frac{right-left+1}{2} \leq$ right-left+1 as well $\frac{right-left}{2} \leq$ right-left+1 ≤ k. So the recursive call is made on interval 0 to k and is correct by Hyp. Thus circularBinarySearch must also work for interval k+1.

∴ as the inductive step works in all cases we can see circularBinarySearch works in all cases.