Find minimum element in a sorted and rotated array

```
#include <stdio.h>
#include <stdlib.h>

int getMinimumElement(int *arr, int start, int end)
{
	if( end < start)
		return arr[0];

	if(end == start)
		return arr[start];

	int middle = (start + end) / 2;

	if(middle < end && arr[middle + 1] < arr[middle])
		return arr[middle+1];

	if(middle > start && arr[middle] < arr[middle - 1])
		return arr[middle];

	return (arr[end] > arr[middle]) ? getMinimumElement(arr, start, middle-1):
	getMinimumElement(arr, middle+1, end);
}

int main()
{
	int *arr, size;
	printf("Enter size of an array\n");
	scanf("%d", &size);
	//allocate memory for array
	arr = (int *)malloc(size * sizeof(int));

	printf("Enter Array elements ");
	for(int index = 0; index < size; index++)
		scanf("%d", &arr[index]);

	printf("The minimum elment is = %d", getMinimumElement(arr, 0, size-1));
	return 0;
}
```

Time Complexity: O(logn)
Space Complexity: O(logn)