Binary search implementation

Things to ask and consider

- Is the array already sorted or do I have to sort it?
- How big is the input array

Initial algorithm

- check if the start index is less than to the end index
    - if start index is less than the end index
        - find middle index
        - compare middle index with search value
            - if middle index is equal to search value
                - return middle index
            - if middle index is greater than the search value
                - change the end index to middle index and recursive call
            - if middle index is less than the search value
                - change the start index to middle index and recursive call
    - if start index is greater than the end index
        - return -1

Python:HIGHLIGHTED IS WHAT I MISSED

```
#binary search implementation
def binary_search(list, start, end, value):
“””
        does a binary search on a list.

        @params
                list [ ]: the sorted list to search from
                start: the starting index for the list to search from
                end: the ending index for the list to search from
                value: the value to search for in the list
        @returns
                index: the index where the value is at in the list OR -1 for not found
“””
        #Check base case
        if start <= end:
                #find the middle value
                middle = (start+end)//2 +1 ←Not needed
                mid_val = list[middle]

                #compare values
                if mid_val == value:
                        return middle
                elif mid_val > value:
                        return binary_search(list, start, middle-1, value)
                elif mid_val < value:
                        return binary_search(list, middle+1, start end, value)
        else:
                return -1

#Driver code
def main():
    list = [10, 20, 30, 50, 60, 80, 110, 130, 140, 170]
    value = 50
    val_index = binary_search(list,0,len(list)-1,value)

    print(f"The value {value} is at index {val_index}")

    value = 200
    val_index = binary_search(list,0,len(list)-1,value)
    print(f"The value of {value} is too big therefore the index is {val_index}")

    value = -100
```

```
    val_index = binary_search(list,0,len(list)-1,value)
    print(f"The value of {value} is too small therefore the index is {val_index}")


if __name__ == “__main__”:
        main()
```