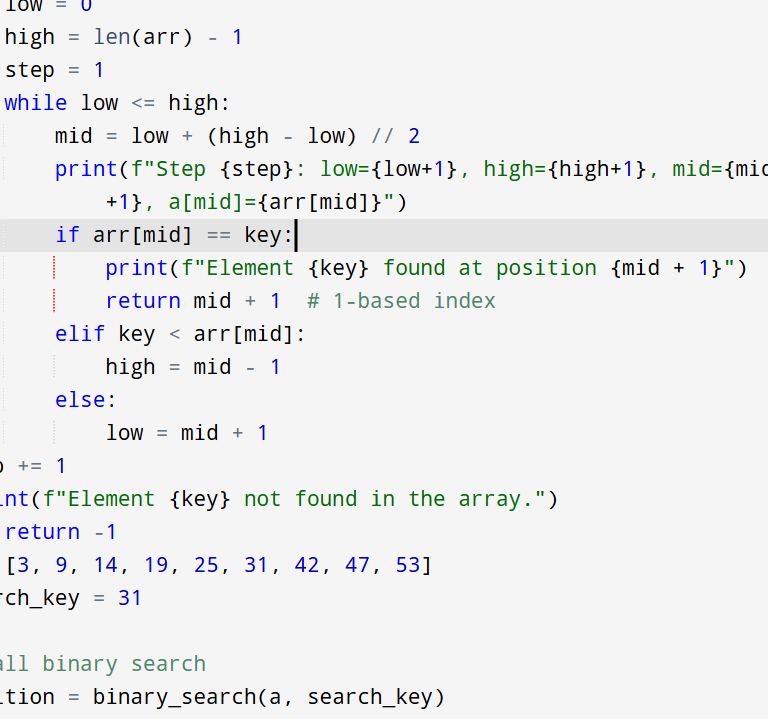
**3.8 BINARY SEARCH ALGORITHM**

**Aim**: To write a Python program to execute binary search algorithm

**Algorithm:**

1. Start
2. Input: sorted array a[0...N-1], search key x
3. Initialize:
4. low = 0
5. high = N - 1
6. While low <= high:
7. Compute mid = (low + high) // 2
8. If a[mid] == x, return mid + 1 (for 1-based index)
9. If x < a[mid], search left (high = mid - 1)
10. If x > a[mid], search right (low = mid + 1)
11. If not found, return -1
12. End

**Program:**

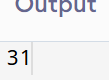
****

**Input:**

a = [3, 9, 14, 19, 25, 31, 42, 47, 53]

search\_key = 31

**Output:**

****

**Result:** Thus the program is successfully executed and the output is verified.

**Performance analysis:**

|  |  |
| --- | --- |
| Time Complexity | O(log n) |

|  |  |
| --- | --- |
| Space Complexity | O(1) |

|  |  |
| --- | --- |
|  |  |