

```
def binary_search(L, v):
    """ (list, object) -> int

    Precondition: L is sorted from smallest to largest, and
    all the items in L can be compared to v.

    Return the index of the first occurrence of v in L, or
    return -1 if v is not in L.

    >>> binary_search([2, 3, 5, 7], 2)
    0
    >>> binary_search([2, 3, 5, 5], 5)
    2
    >>> binary_search([2, 3, 5, 7], 8)
    -1
    """

    b = 0
    e = len(L) - 1

    while b <= e:
        m = (b + e) // 2
        if L[m] < v:
            b = m + 1
        else:
            e = m - 1

    if b == len(L) or L[b] != v:
        return -1
    else:
        return b

if __name__ == '__main__':
    import doctest
    doctest.testmod()
```