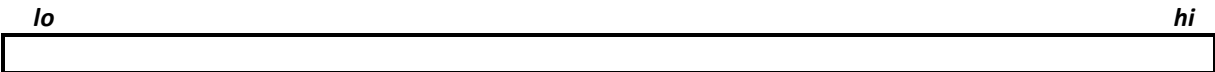


# Items		# Checks
1	?	1
3	1 ? 1	2
7	3 ? 3	3
15	7 ? 7	4
31 15 ? 15	5
63 31 ? 31	6
127 63 ? 63	7
255 127 ? 127	8
511 255 ? 255	9
1023 511 ? 511	10
2047 1023 ? 1023	11
4095 2047 ? 2047	12
8191 4095 ? 4095	13
16383 8191 ? 8191	14
32767 16383 ? 16383	15
65535 32767 ? 32767	16

BINARY SEARCH

Pre-Condition: Items are stored in sorted order

Current search range delimited by indexes *lo* & *hi*

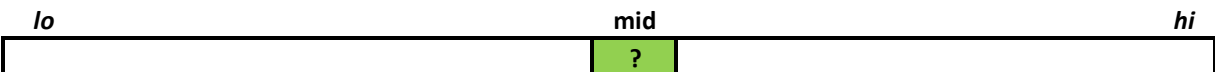


Initialization: *lo* = 0, *hi* = #items - 1

lo > *hi* ? FAILURE, return -1

mid = (*lo* + *hi*) / 2

Case: target == list[mid] ? SUCCESS, return mid



Case: target > list[mid] ? CONTINUE high, *lo* = mid + 1



Case: target < list[mid] ? CONTINUE low, *hi* = mid - 1

