

find

find.c

- prompts for numbers to fill the haystack
- searches the haystack for a needle
 - ▣ calls sort and search, functions defined in helpers.c

TODO

- search
 - ▣ return true if value is found in haystack
 - ▣ return false if value is not in haystack
- sort
 - ▣ sort the values[] array

search

- linear search: you can do better!
 - $O(n)$ → slow
 - Can search any list
- binary search
 - $O(\log n)$ → fast
 - can only search sorted lists

binary search

1	3	6	9	10	14	16	17	21
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]

binary search

1	3	6	9	10	14	16	17	21
---	---	---	---	----	----	----	----	----

[0]

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]



left



middle



right

binary search

1	3	6	9	10	14	16	17	21
---	---	---	---	----	----	----	----	----

[0]

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]

↑
left

↑
right

binary search

1	3	6	9	10	14	16	17	21
---	---	---	---	----	----	----	----	----

[0]

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]



left



right

binary search

1	3	6	9	10	14	16	17	21
---	---	---	---	----	----	----	----	----

[0]

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]



left



middle



right

binary search

1	3	6	9	10	14	16	17	21
---	---	---	---	----	----	----	----	----

[0]

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]

↑
left
middle

↑
right

binary search: pseudocode

```
while length of list > 0
    look at middle of list
    if number found, return true
    else if number higher, search left
    else if number lower, search right
return false
```