


- SEARCH
- INSERT
- DELETE

} The 3 main operations that we perform on data structures
} How fast can we do these?

- SEARCH
 - INSERT
 - DELETE
- } The 3 main operations that we perform on data structures
How fast can we do these?
-

linked list




?

• SEARCH

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

linked list



• SEARCH

$O(n)$

• INSERT

?

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

linked list



• SEARCH

$O(n)$

• INSERT

$O(1)$ [$O(n)$ sorted]

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

linked list



• SEARCH

$O(n)$

• INSERT

$O(1)$ [$O(n)$ sorted]

• DELETE

?

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

linked list



• SEARCH

$O(n)$

• INSERT

$O(1)$ [$O(n)$ sorted]

• DELETE

$O(n)$

search + $O(1)$

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

array



linked list



• SEARCH

?

$O(n)$

• INSERT

$O(1)$ [$O(n)$ sorted]

• DELETE

$O(n)$
search + $O(1)$

• SEARCH } The 3 main operations that we perform on data structures
• INSERT }
• DELETE } How fast can we do these?

array



linked list



• SEARCH

?
by key

$O(n)$

?
by index

• INSERT

$O(1)$ [$O(n)$ sorted]

• DELETE

$O(n)$
search + $O(1)$

- SEARCH
 - INSERT
 - DELETE
- } The 3 main operations that we perform on data structures
How fast can we do these?
-

array



linked list



- SEARCH $O(n) \rightarrow [O(\log n) \text{ sorted}]$
 - ↳ by key
 - $O(1)$
 - ↳ by index

$O(n)$

- INSERT ?

$O(1) [O(n) \text{ sorted}]$

- DELETE ?

$O(n)$
search + $O(1)$

- SEARCH
 - INSERT
 - DELETE
- } The 3 main operations that we perform on data structures
How fast can we do these?
-

array



linked list



- SEARCH $O(n) \rightarrow [O(\log n) \text{ sorted}]$
 - ↳ by key
 - $O(1)$
 - ↳ by index
- INSERT $O(n)$ $O(1) [O(n) \text{ sorted}]$
- DELETE $O(n)$ $O(n)$
 - search + $O(1)$ search + $O(1)$

• SEARCH
 • INSERT
 • DELETE

} The 3 main operations that we perform on data structures
 } How fast can we do these?

array



linked list



• SEARCH $O(n) \rightarrow [O(\log n) \text{ sorted}]$
 ↳ by key
 $O(1)$
 ↳ by index

$O(n)$

• INSERT $O(n)$
 • DELETE $O(n)$

} $[O(1) \text{ amortized}]$
 } $[\text{if not sorted}]$

search + $O(1)$

$O(1) [O(n) \text{ sorted}]$

$O(n)$

search + $O(1)$

• SEARCH
 • INSERT
 • DELETE

} The 3 main operations that we perform on data structures
 } How fast can we do these?

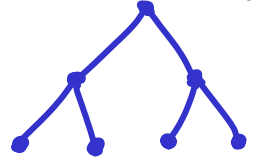
array



linked list



BST (RB, AVL, etc)



?

?

?

• SEARCH $O(n) \rightarrow [O(\log n) \text{ sorted}]$
 ↳ by key
 $O(1)$
 ↳ by index

$O(n)$

• INSERT $O(n)$
 • DELETE $O(n)$

$[O(1) \text{ amortized}]$
 $[\text{if not sorted}]$

search + $O(1)$

$O(1) [O(n) \text{ sorted}]$

$O(n)$

search + $O(1)$

• SEARCH
 • INSERT
 • DELETE

} The 3 main operations that we perform on data structures
 } How fast can we do these?

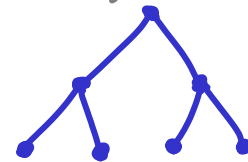
array



linked list



BST (RB, AVL, etc)



• SEARCH $O(n) \rightarrow [O(\log n) \text{ sorted}]$
 ↳ by key
 $O(1)$
 ↳ by index

$O(n)$

$O(\log n)$

• INSERT $O(n)$
 • DELETE $O(n)$

} $[O(1) \text{ amortized}]$
 } $[\text{if not sorted}]$

search + $O(1)$

$O(1) [O(n) \text{ sorted}]$

$O(\log n)$

$O(n)$

$O(\log n)$

search + $O(1)$