

CS50 for MBAs

Algorithms

Last Time

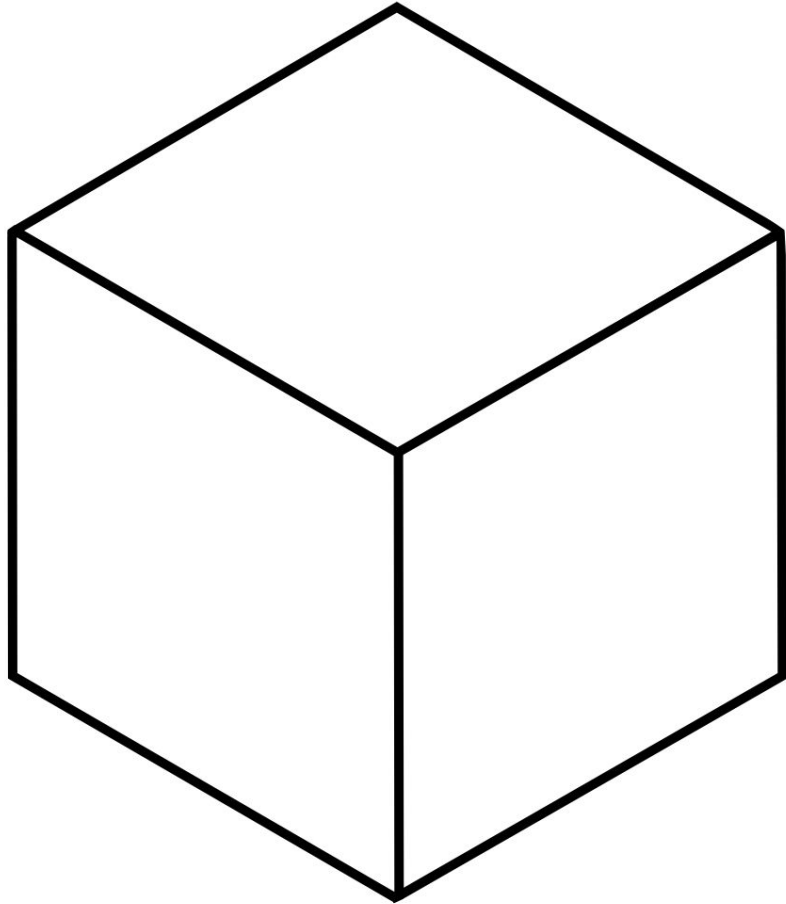
- representation
- algorithms, pseudocode
- functions, variables, conditions, loops
- ...

This Time

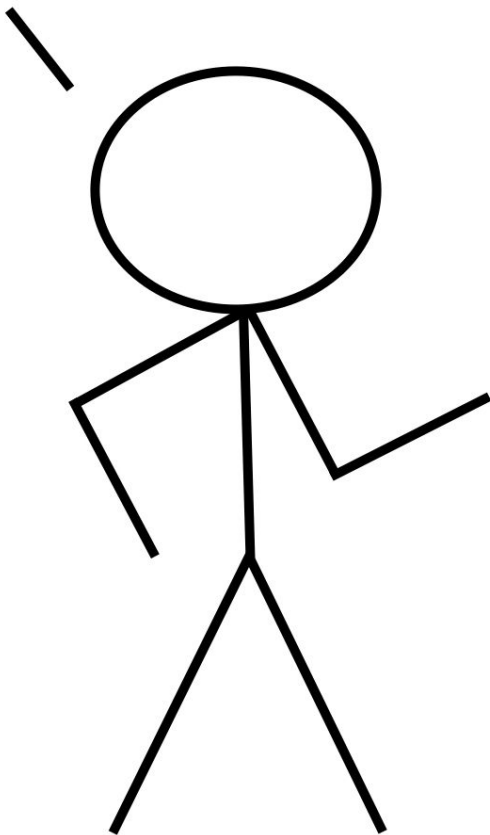
- abstraction
- correctness, efficiency
- searching, sorting

shorts

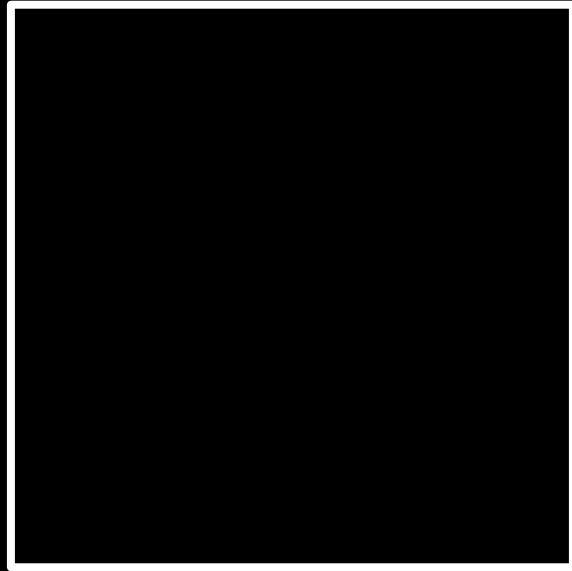
abstraction



Hi



input →



→ output



algorithms





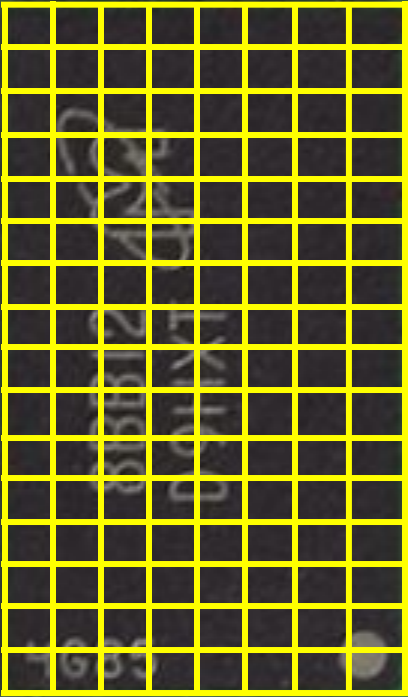
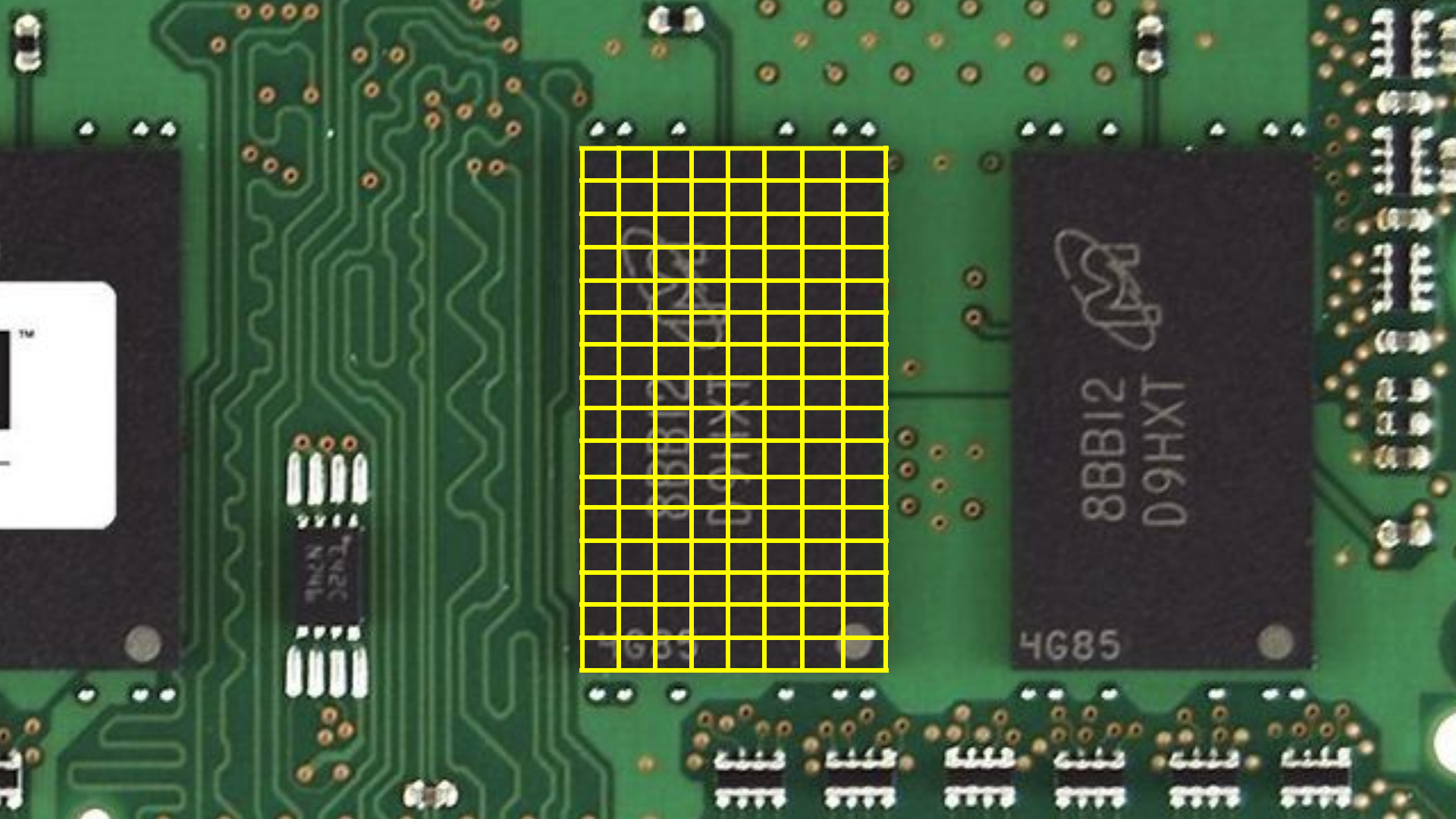
8BB12
D9HXT

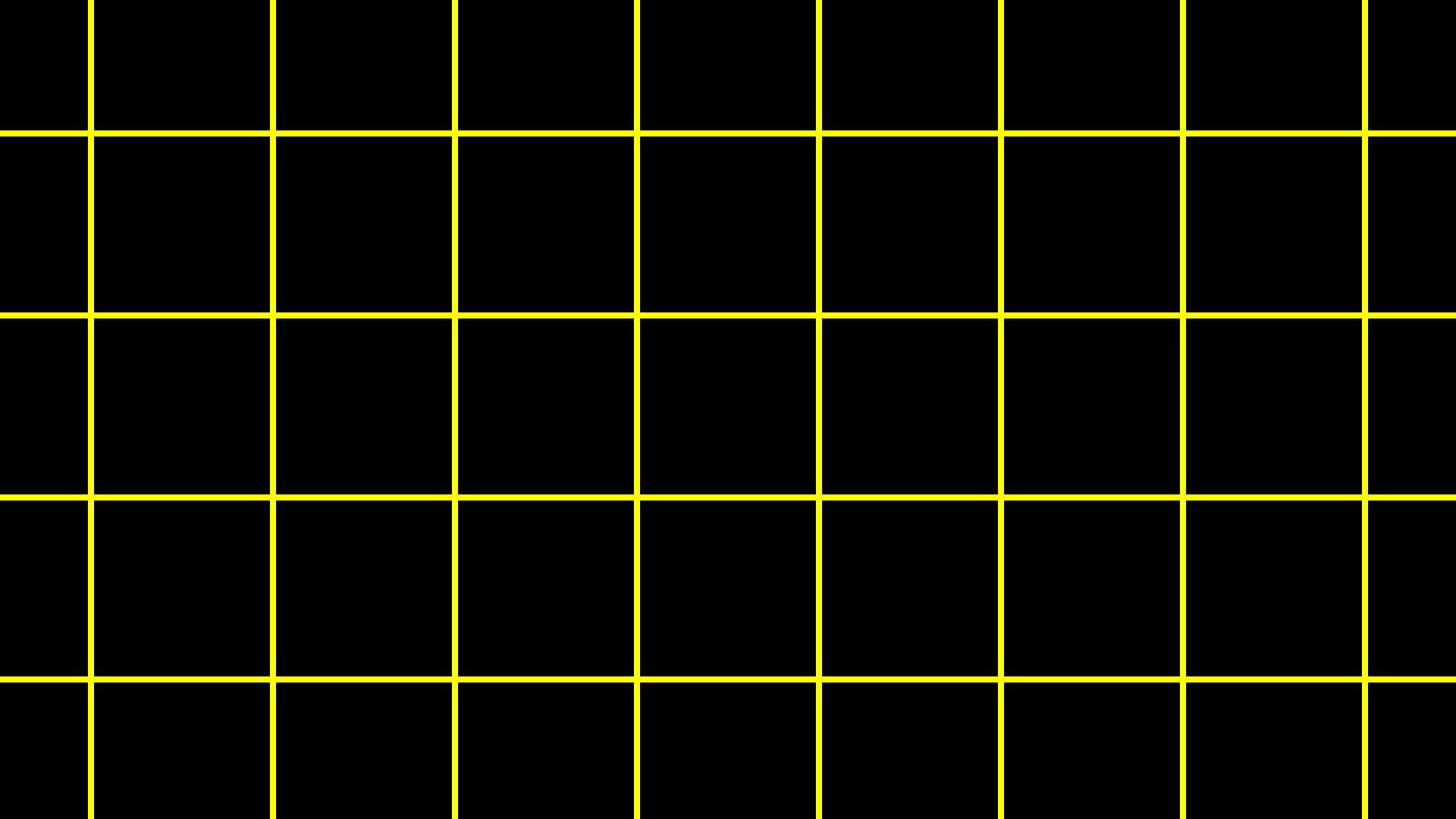
4G85

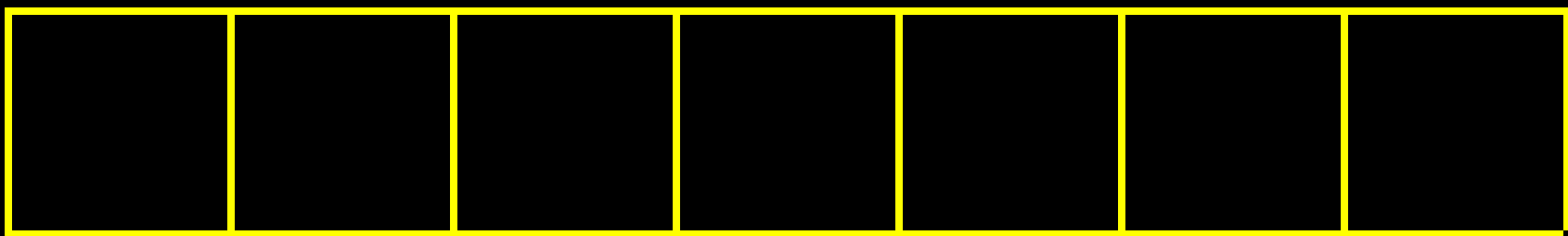


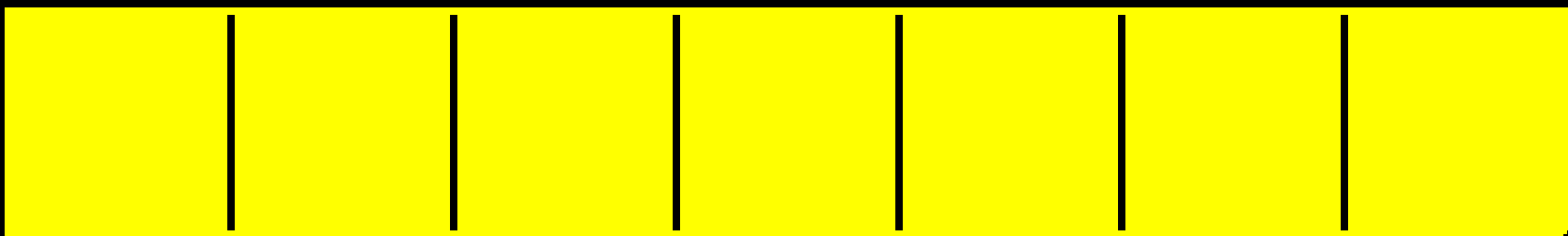
8BB12
D9HXT

4G85









linear search

```
For i from 0 to n-1
    If i'th element is 50
        Return true
Return false
```

binary search

If middle item is 50

 Return true

Else if $50 < \text{middle item}$

 Search left half

Else if $50 > \text{middle item}$

 Search right half

If no items

 Return false

If middle item is 50

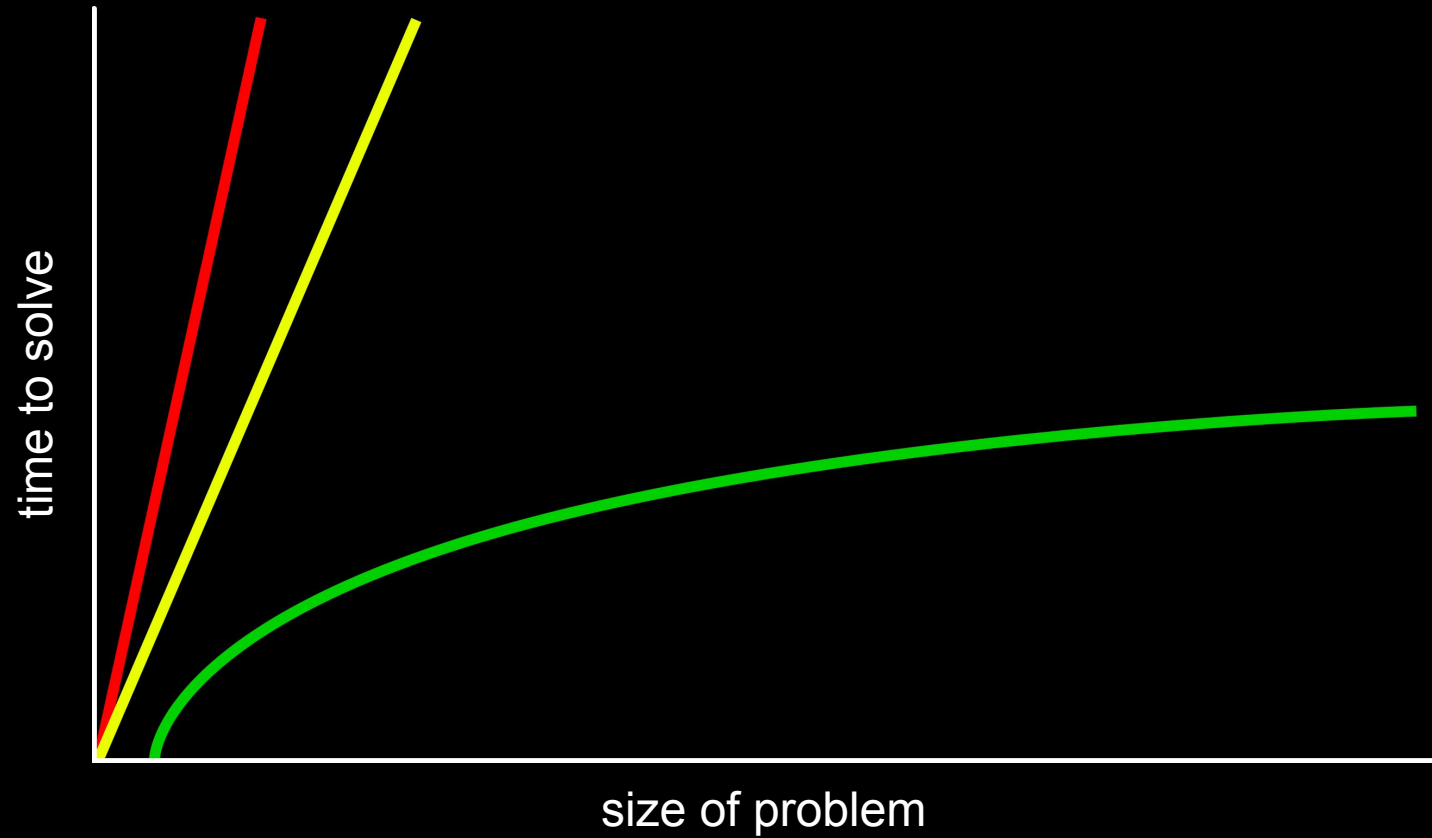
 Return true

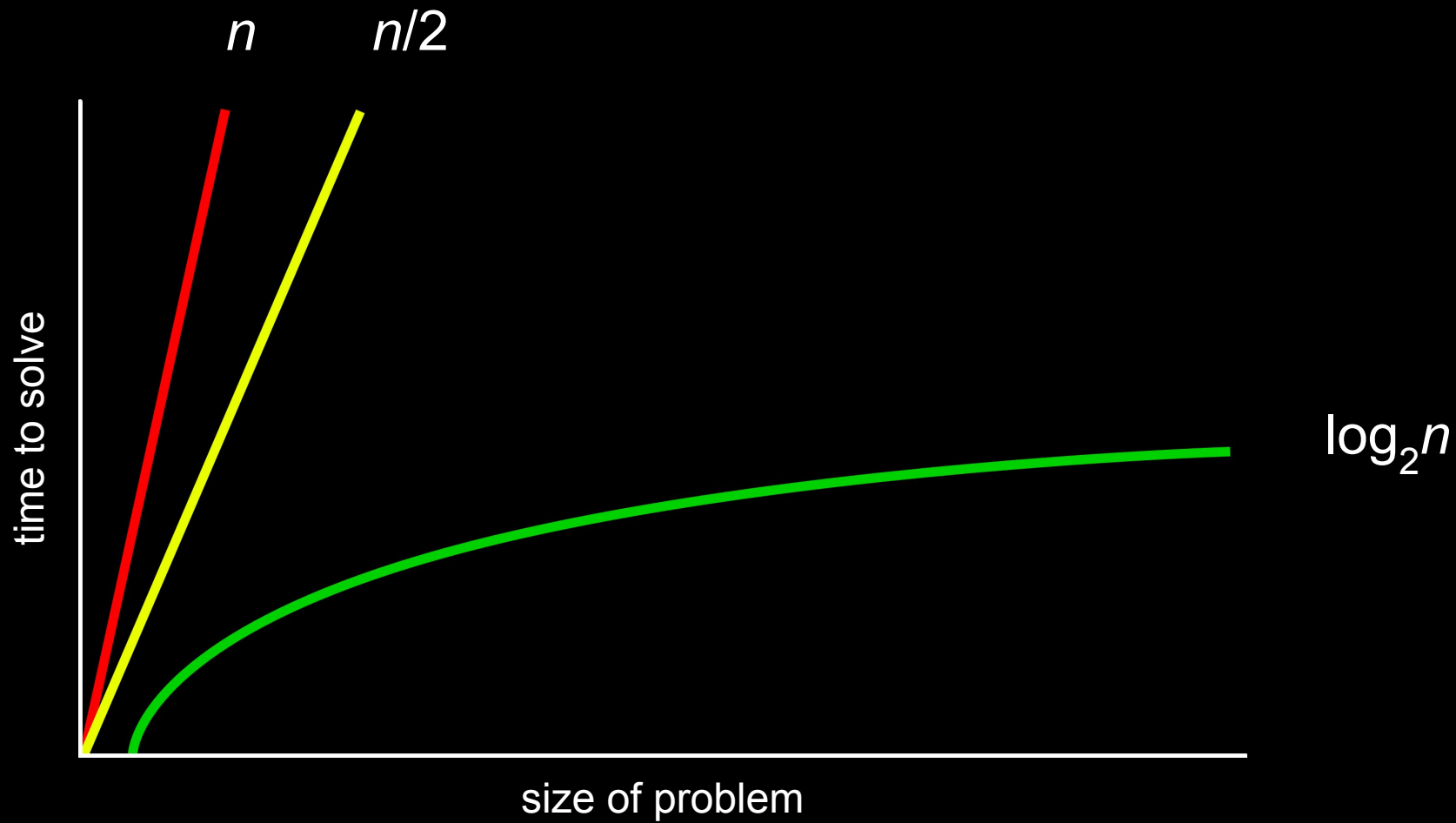
Else if $50 < \text{middle item}$

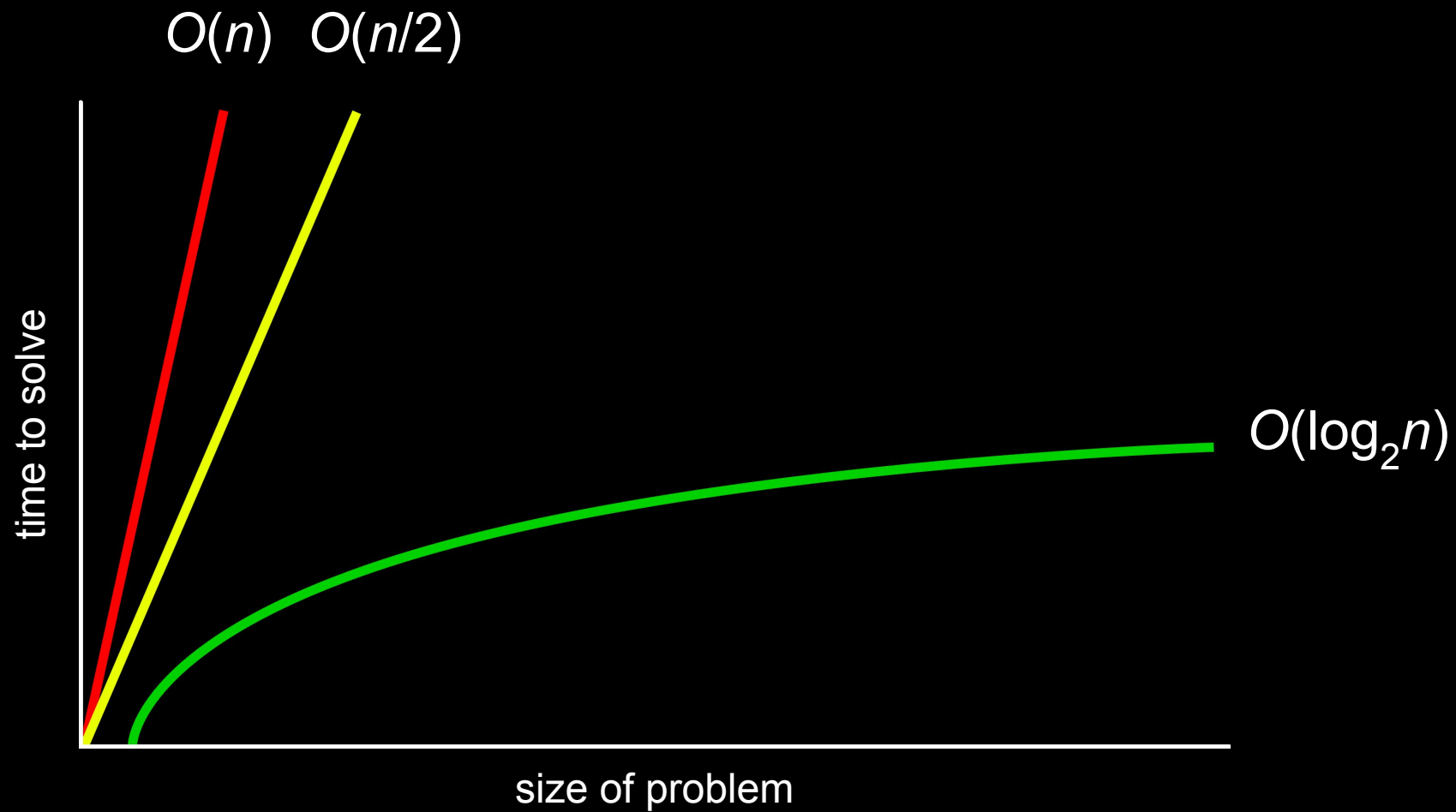
 Search left half

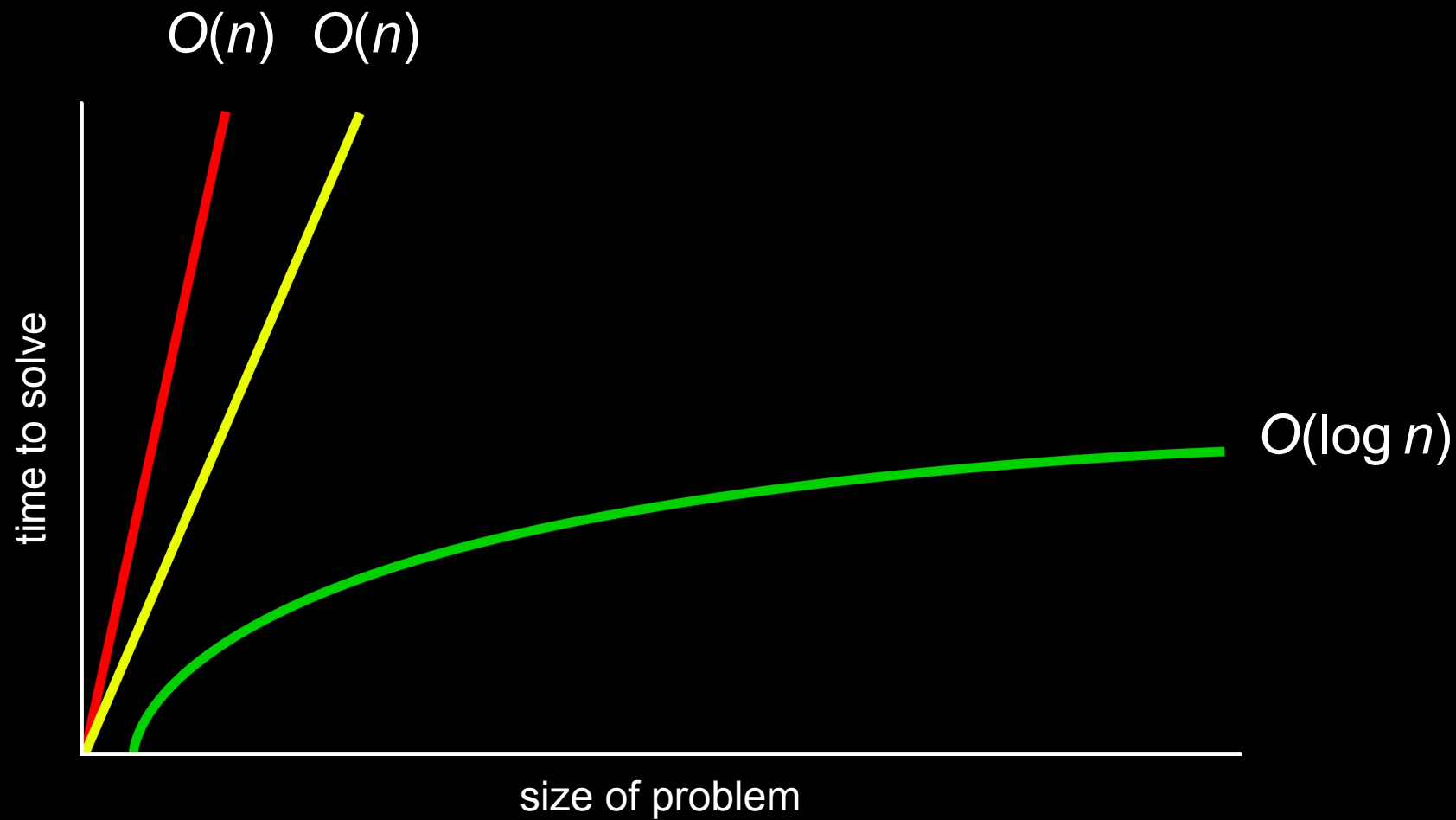
Else if $50 > \text{middle item}$

 Search right half









$O(n^2)$

$O(n \log n)$

$O(n)$

$O(\log n)$

$O(1)$

$O(n^2)$

$O(n \log n)$

$O(n)$ linear search

$O(\log n)$ binary search

$O(1)$

$\Omega(n^2)$

$\Omega(n \log n)$

$\Omega(n)$

$\Omega(\log n)$

$\Omega(1)$

$\Omega(n^2)$

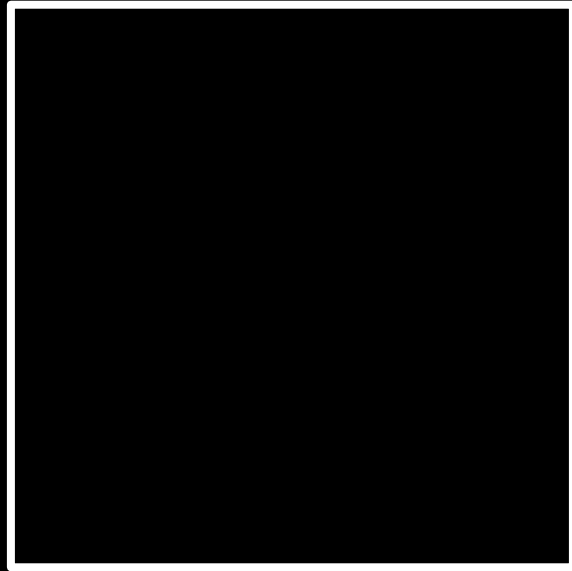
$\Omega(n \log n)$

$\Omega(n)$

$\Omega(\log n)$

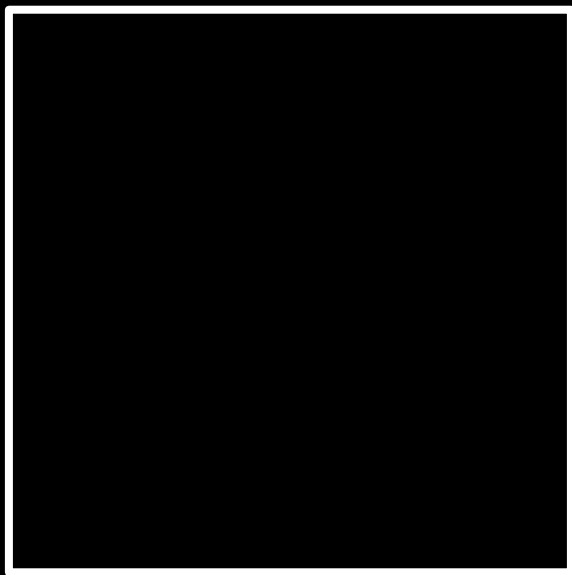
$\Omega(1)$ linear search, binary search

input →



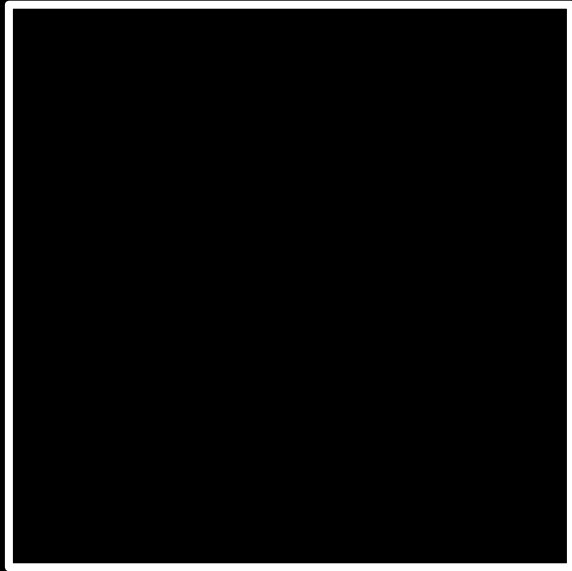
→ output

unsorted →



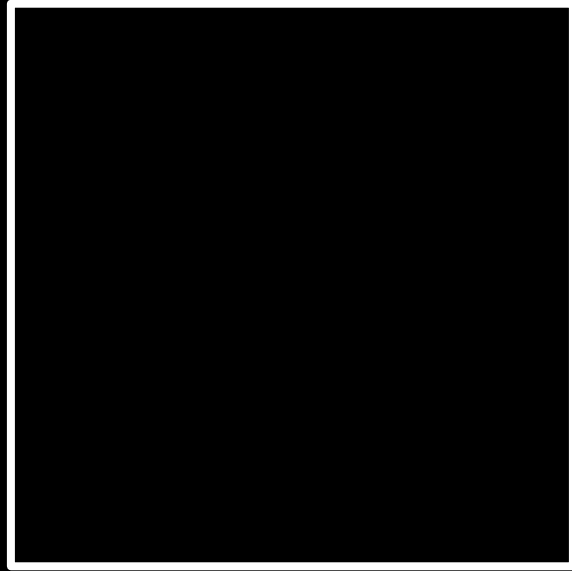
→ output

unsorted →



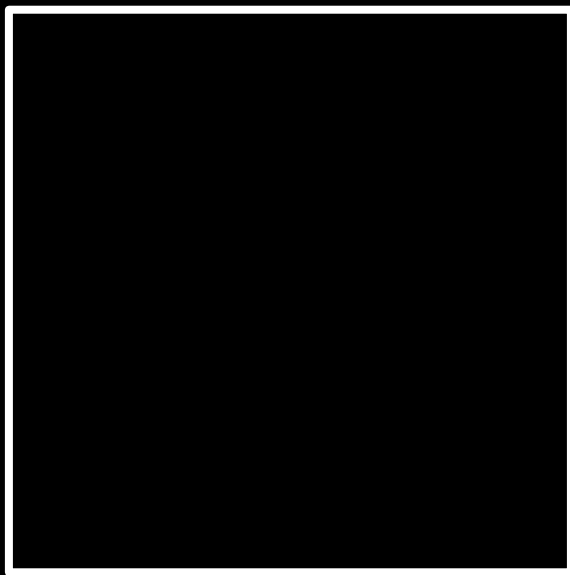
→ sorted

15 23 16 8 42 50 4



sorted

15 23 16 8 42 50 4



4 8 15 16 23 42 50

6 3 8 5 2 7 4 1

bubble sort

Repeat $n-1$ times

For i from 0 to $n-2$

 If i 'th and $i+1$ 'th elements out of order

 Swap them

$$(n - 1) \times (n - 1)$$

$$(n - 1) \times (n - 1)$$

$$n^2 - 1n - 1n + 1$$

$$(n - 1) \times (n - 1)$$

$$n^2 - 1n - 1n + 1$$

$$n^2 - 2n + 1$$

$$(n - 1) \times (n - 1)$$

$$n^2 - 1n - 1n + 1$$

$$n^2 - 2n + 1$$

$$O(n^2)$$

$O(n^2)$ bubble sort

$O(n \log n)$

$O(n)$

$O(\log n)$

$O(1)$

$\Omega(n^2)$ bubble sort

$\Omega(n \log n)$

$\Omega(n)$

$\Omega(\log n)$

$\Omega(1)$

6 3 8 5 2 7 4 1

selection sort

For i from 0 to $n-1$

 Find smallest item between i 'th item and last item

 Swap smallest item with i 'th item

$$n + (n - 1)$$

$$n + (n - 1) + (n - 2)$$

$$n + (n - 1) + (n - 2) + \dots + 1$$

$$n + (n - 1) + (n - 2) + \dots + 1$$

$$n(n + 1)/2$$

$$n + (n - 1) + (n - 2) + \dots + 1$$

$$n(n + 1)/2$$

$$(n^2 + n)/2$$

$$n + (n - 1) + (n - 2) + \dots + 1$$

$$n(n + 1)/2$$

$$(n^2 + n)/2$$

$$n^2/2 + n/2$$

$$n + (n - 1) + (n - 2) + \dots + 1$$

$$n(n + 1)/2$$

$$(n^2 + n)/2$$

$$n^2/2 + n/2$$

$$O(n^2)$$

$O(n^2)$ selection sort

$O(n \log n)$

$O(n)$

$O(\log n)$

$O(1)$

$\Omega(n^2)$ selection sort

$\Omega(n \log n)$

$\Omega(n)$

$\Omega(\log n)$

$\Omega(1)$

bubble sort

Repeat $n-1$ times

 For i from 0 to $n-2$

 If i 'th and $i+1$ 'th elements out of order

 Swap them

Repeat until no swaps

 For i from 0 to $n-2$

 If i 'th and $i+1$ 'th elements out of order

 Swap them

$\Omega(n^2)$ selection sort, bubble sort

$\Omega(n \log n)$

$\Omega(n)$

$\Omega(\log n)$

$\Omega(1)$ linear search, binary search

$\Omega(n^2)$ selection sort

$\Omega(n \log n)$

$\Omega(n)$ bubble sort

$\Omega(\log n)$

$\Omega(1)$ linear search, binary search



recursion

```
1  Pick up phone book
2  Open to middle of phone book
3  Look at page
4  If person is on page
5      Call person
6  Else if person is earlier in book
7      Open to middle of left half of book
8      Go back to line 3
9  Else if person is later in book
10     Open to middle of right half of book
11     Go back to line 3
12 Else
13     Quit
```

```
1 Pick up phone book
2 Open to middle of phone book
3 Look at page
4 If person is on page
5     Call person
6 Else if person is earlier in book
7     Open to middle of left half of book
8     Go back to line 3
9 Else if person is later in book
10    Open to middle of right half of book
11    Go back to line 3
12 Else
13    Quit
```

```
1 Pick up phone book
2 Open to middle of phone book
3 Look at page
4 If person is on page
5     Call person
6 Else if person is earlier in book
7     Open to middle of left half of book
8     Go back to line 3
9 Else if person is later in book
10    Open to middle of right half of book
11    Go back to line 3
12 Else
13    Quit
```

```
1 Pick up phone book
2 Open to middle of phone book
3 Look at page
4 If person is on page
5     Call person
6 Else if person is earlier in book
7     Search left half of book
8
9 Else if person is later in book
10    Search right half of book
11
12 Else
13    Quit
```

```
1 Pick up phone book
2 Open to middle of phone book
3 Look at page
4 If person is on page
5     Call person
6 Else if person is earlier in book
7     Search left half of book
8 Else if person is later in book
9     Search right half of book
10 Else
11     Quit
```













merge sort

If only one item

Return

Else

Sort left half of items

Sort right half of items

Merge sorted halves

If only one item

Return

Else

Sort left half of items

Sort right half of items

Merge sorted halves

7 4 5 2 6 3 8 1

7

4

5

2

6

3

8

1

7

4

5

2

6

3

8

1

7

4

5

2

6

3

8

1

7

4

5

2

6

3

8

1

7

4

5

2

6

3

8

1

7

5

2

6

3

8

1

4

5 2 6 3 8 1

4 7

5

2

6

3

8

1

4

7

5

2

6

3

8

1

4

7

5

2

6

3

8

1

4

7

5 2 6 3 8 1

4 7

5

6

3

8

1

4

7

2

6 3 8 1

4 7 2 5

6 3 8 1

4	7	2	5
---	---	---	---

6 3 8 1

4 7

5

2

6 3 8 1

7

5

2 4

6 3 8 1

7

2 4 5

6 3 8 1

2 4 5 7

6 3 8 1

2 4 5 7

6

3

8

1

2

4

5

7

6

3

8

1

2

4

5

7

6

3

8

1

2

4

5

7

6 3 8 1

2 4 5 7

6

8

1

3

2

4

5

7

8 1

3 6

2 4 5 7

8 1

3 6

2 4 5 7

8

1

3

6

2

4

5

7

8



3

6

2

4

5

7

8	1
---	---

3 6

2 4 5 7

8

3

6

1

2

4

5

7

3

6

1

8

2

4

5

7

3	6	1	8
---	---	---	---

2 4 5 7

2 4 5 7

1

3 6

8

2

4

5

7

1

3

6

8

2

4

5

7

1

3

6

8

2

4

5

7

1

3

6

8

2

4

5

7

1

3

6

8

2 4 5 7

3 6 8

1

4 5 7

3 6 8

1 2

4 5 7

6 8

1 2 3

1 2 3 4

5 7

6 8

7

6 8

1 2 3 4 5

7

8

1

2

3

4

5

6

1 2 3 4 5 6 7

8

1

2

3

4

5

6

7

8

7	4	5	2	6	3	8	1
---	---	---	---	---	---	---	---

7	4	5	2	6	3	8	1
---	---	---	---	---	---	---	---

4	7	2	5	3	6	1	8
---	---	---	---	---	---	---	---

7	4	5	2	6	3	8	1
---	---	---	---	---	---	---	---

4	7	2	5	3	6	1	8
---	---	---	---	---	---	---	---

2	4	5	7	1	3	6	8
---	---	---	---	---	---	---	---

7	4	5	2	6	3	8	1
---	---	---	---	---	---	---	---

4	7	2	5	3	6	1	8
---	---	---	---	---	---	---	---

2	4	5	7	1	3	6	8
---	---	---	---	---	---	---	---

1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---

$O(n^2)$

$O(n \log n)$ merge sort

$O(n)$

$O(\log n)$

$O(1)$

$\Omega(n^2)$

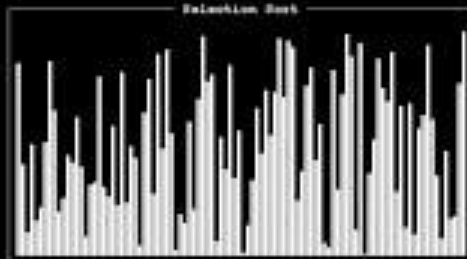
$\Omega(n \log n)$ merge sort

$\Omega(n)$

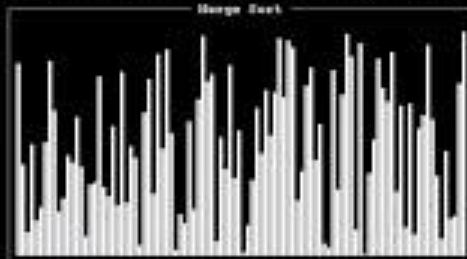
$\Omega(\log n)$

$\Omega(1)$

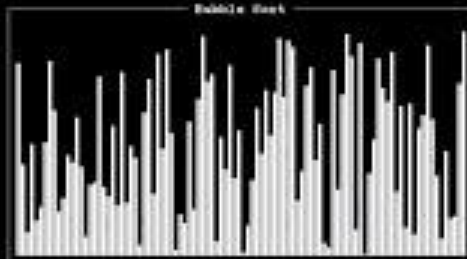
Salmon Run



Harpe Run



Public Run



Assignment 1

Office Hours

Lab

CS50 for MBAs

Algorithms