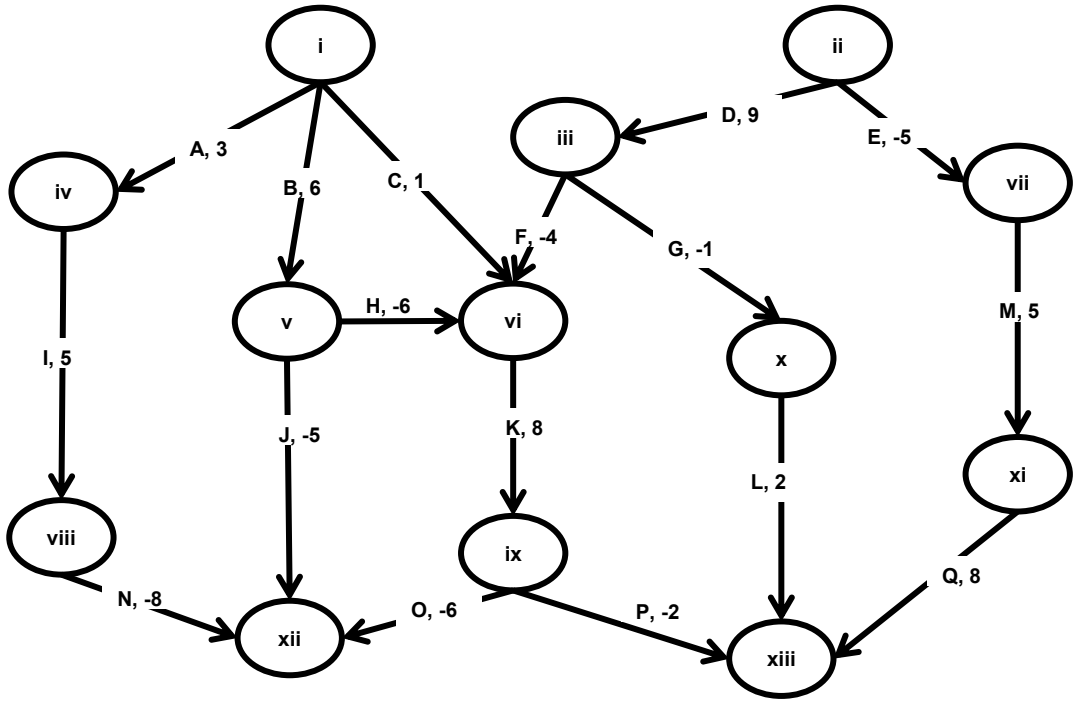
Genome 540: Discussion Section Class - 11

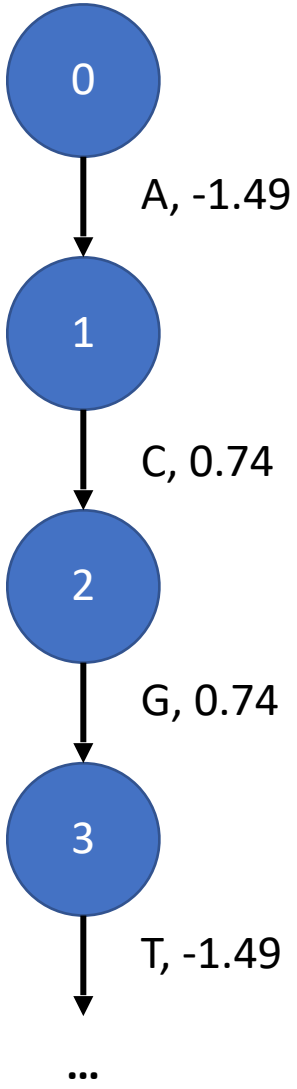
Chengxiang Qiu

HW4 questions?

- Part 1



- Part 2



Checking if you match the template

- When testing your code on the example, run ‘diff’ between your output and the sample output

```
> diff your_output.txt example_output.txt
```

- The only differences should be the header.

Diff Example

file1.txt:

```
I need to go to the store.
I need to buy some apples.
When I get home, I'll wash the dog.
```

file2.txt:

```
I need to go to the store.
I need to buy some apples.
Oh yeah, I also need to buy grated cheese.
When I get home, I'll wash the dog.
```

```
diff file1.txt file2.txt
```

Output:

```
2a3
> Oh yeah, I also need to buy grated cheese.
```

Diff Example

file1.txt:

```
I need to go to the store.
I need to buy some apples.
When I get home, I'll wash the dog.
```

file2.txt:

```
I need to go to the store.
I need to buy some apples.
Oh yeah, I also need to buy grated cheese.
When I get home, I'll wash the dog.
```

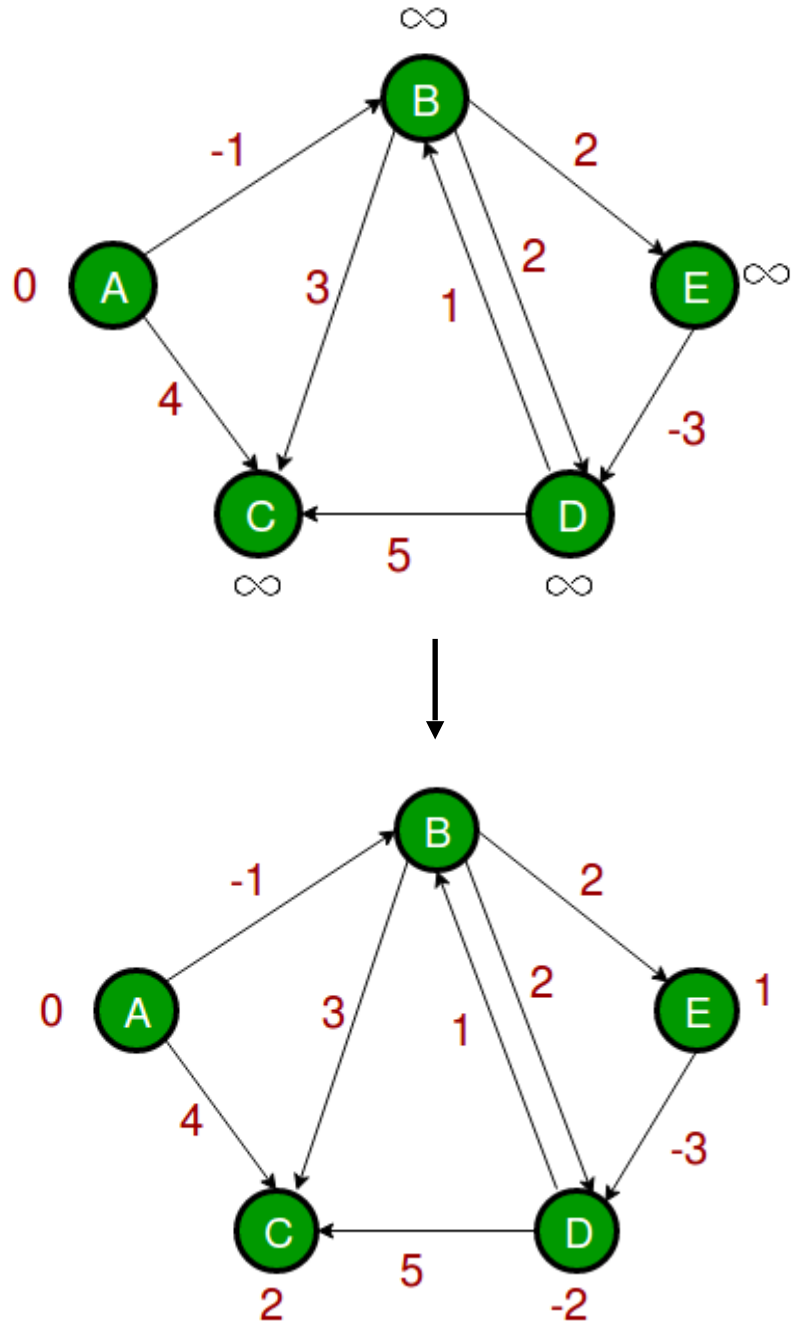
```
diff –y file1.txt file2.txt
```

Output:

```
I need to go to the store.              I need to go to the store.
I need to buy some apples.                  I need to buy some apples.
                                          > Oh yeah, I also need to buy grated cheese.
When I get home, I'll wash the dog.     When I get home, I'll wash the dog.
```

Time complexity

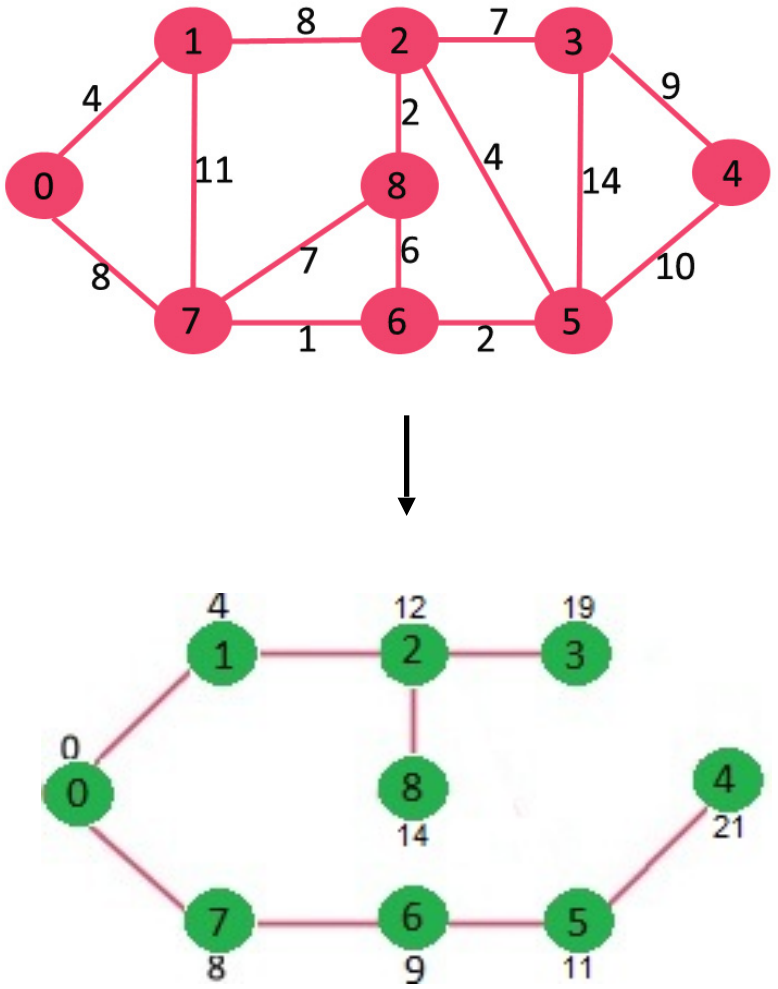
- Bellman-Ford algorithm (for given start vertex)
  - Choose source node and set distance to 0
  - Set distance to all other nodes to infinity
  - For **each** edge (u,v), if v's distance can be reduced by taking that edge, update v's distance
  - Cycle through all edges in this way |V|-1 times
  - (can also check for negative-weight cycle with one extra iteration)



O(V*E)

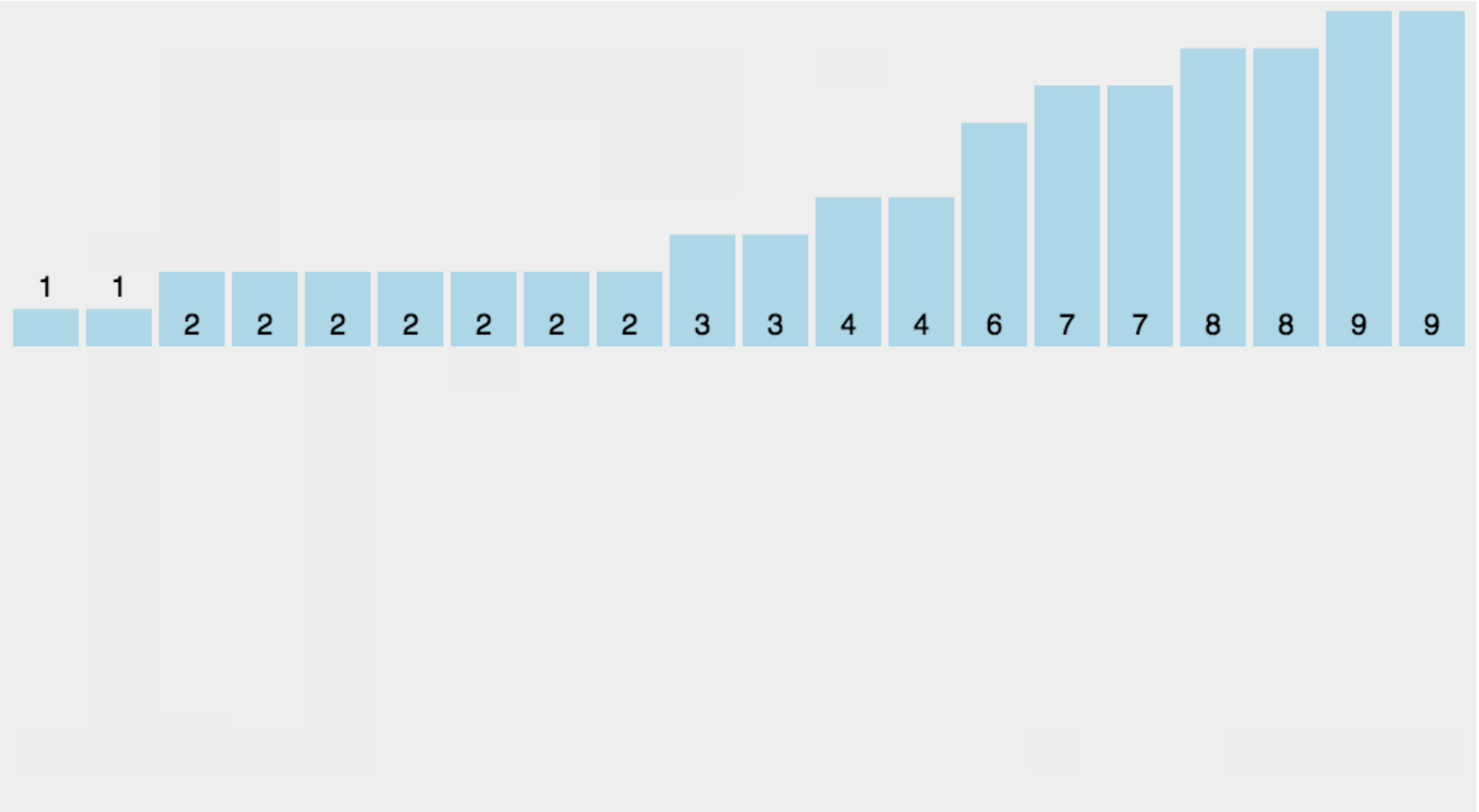
Time complexity

- Dijkstra's algorithm (for given start vertex)
  - Choose source node and set distance to 0
  - Set distance to all other nodes to infinity
  - Set source node to **current**
  - Make distance offers to all **unvisited** neighbors, which are accepted if they're less than the previous best offer
  - Mark current as **visited** (it will never be updated again)
  - Select unvisited neighbor with smallest distance, set it to current, and repeat



$O(V^2)$

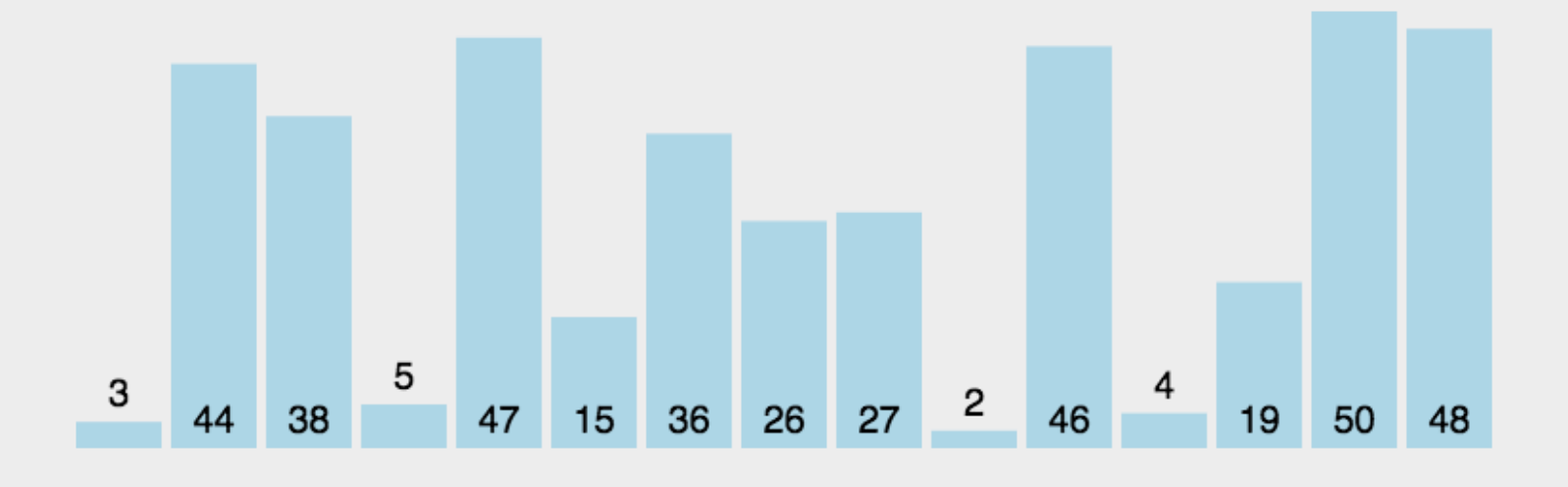
Sorting algorithms and their time complexities



Bucket Sort

O(M + N)

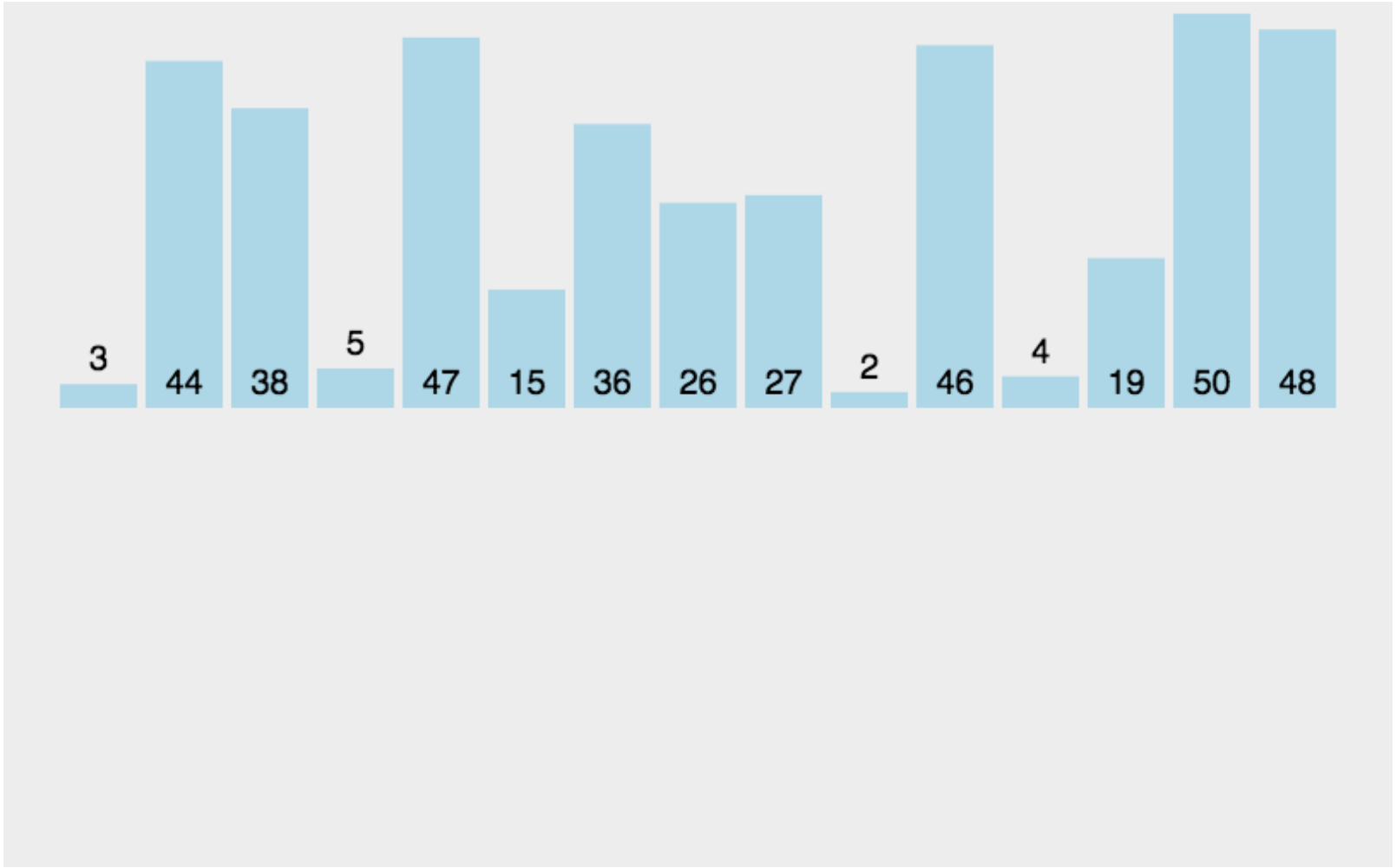
Sorting algorithms and their time complexities



Bubble Sort

$O(N^2)$

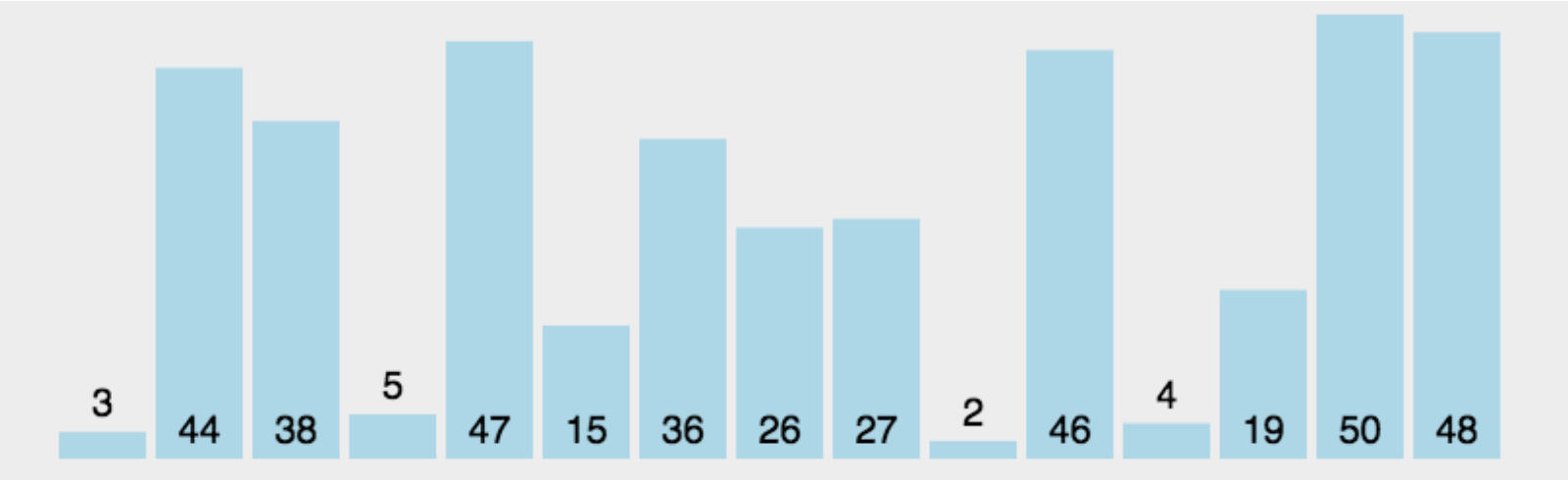
Sorting algorithms and their time complexities



Insertion Sort

$O(N^2)$

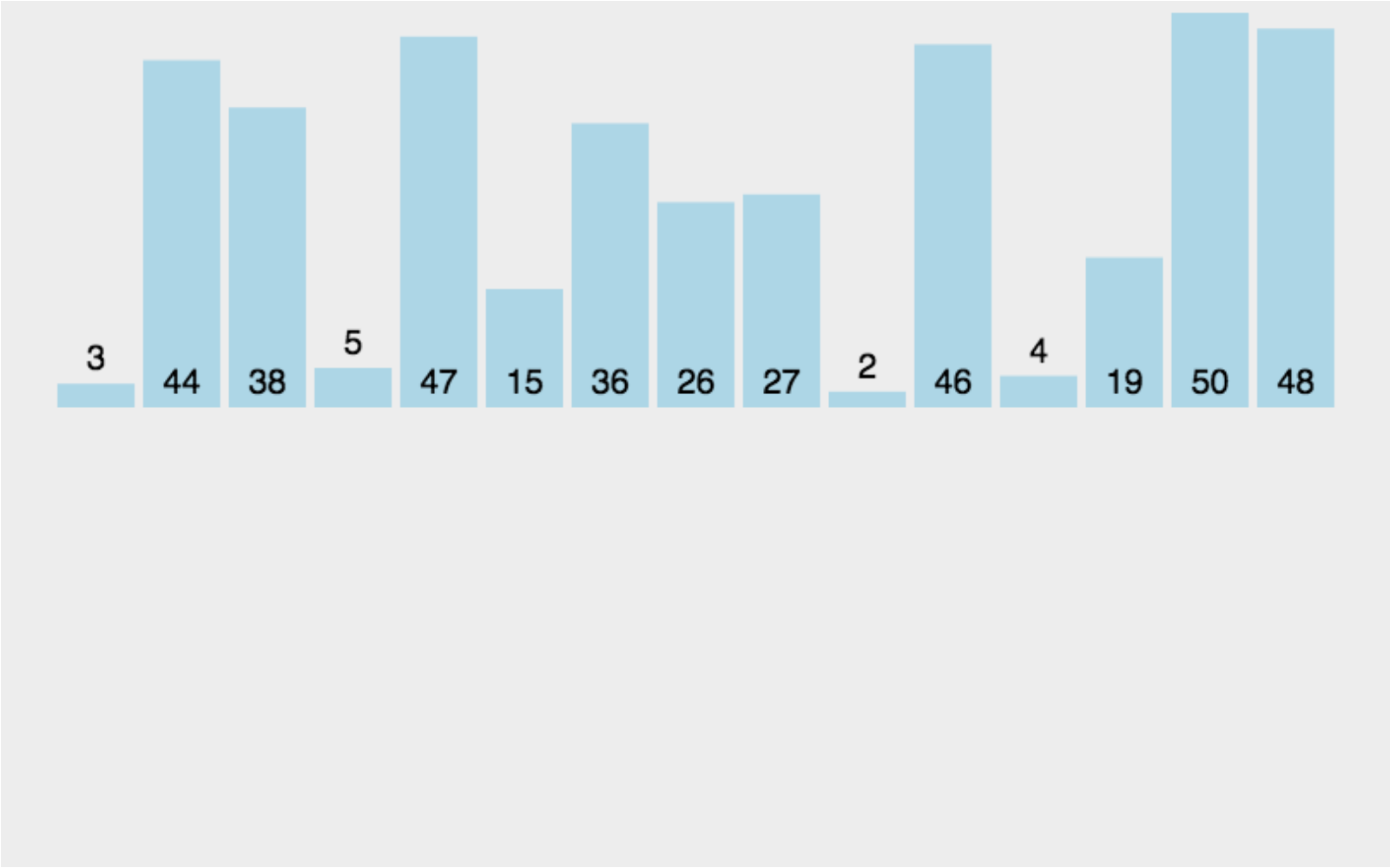
Sorting algorithms and their time complexities



Selection Sort

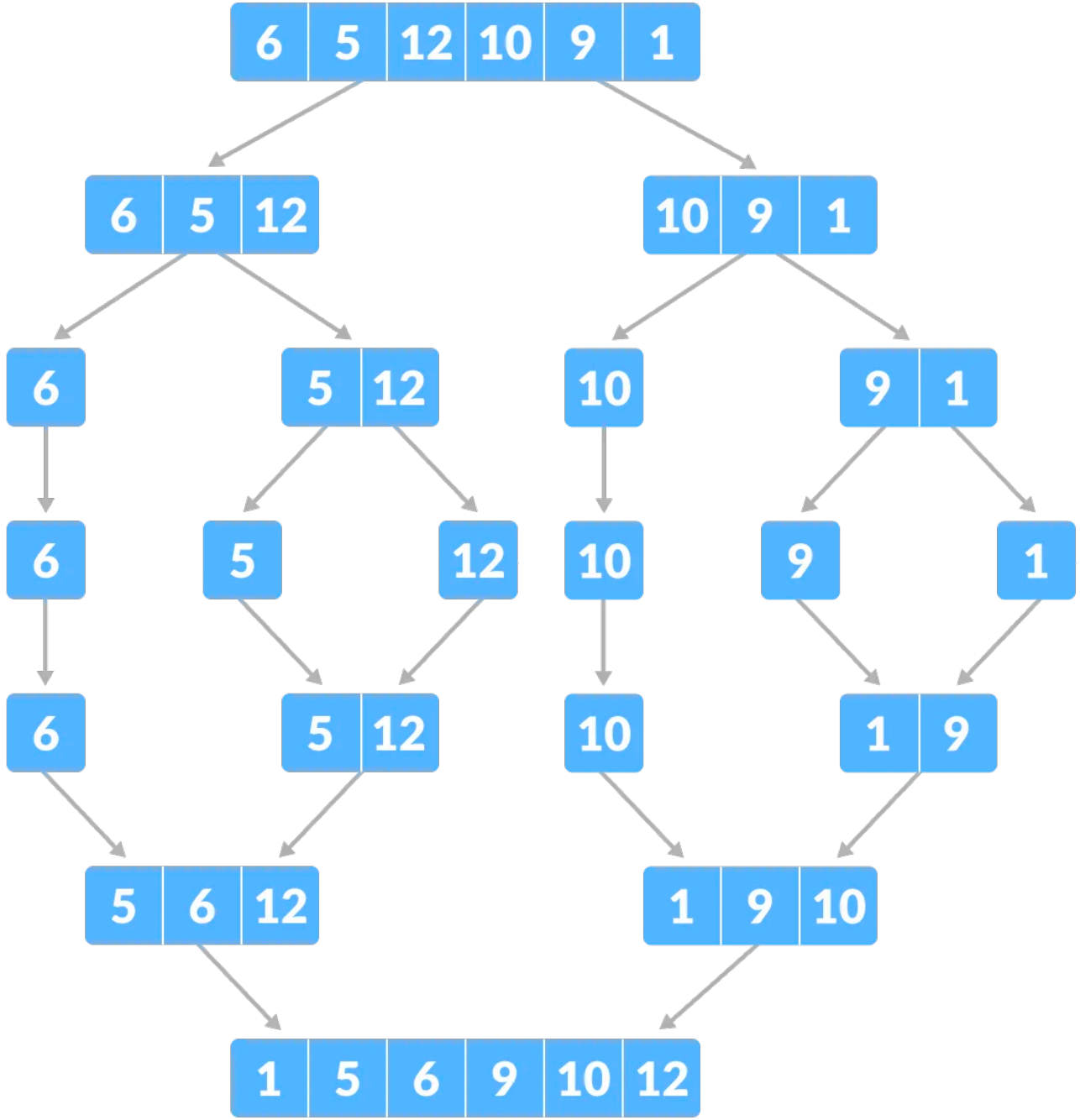
$O(N^2)$

Sorting algorithms and their time complexities



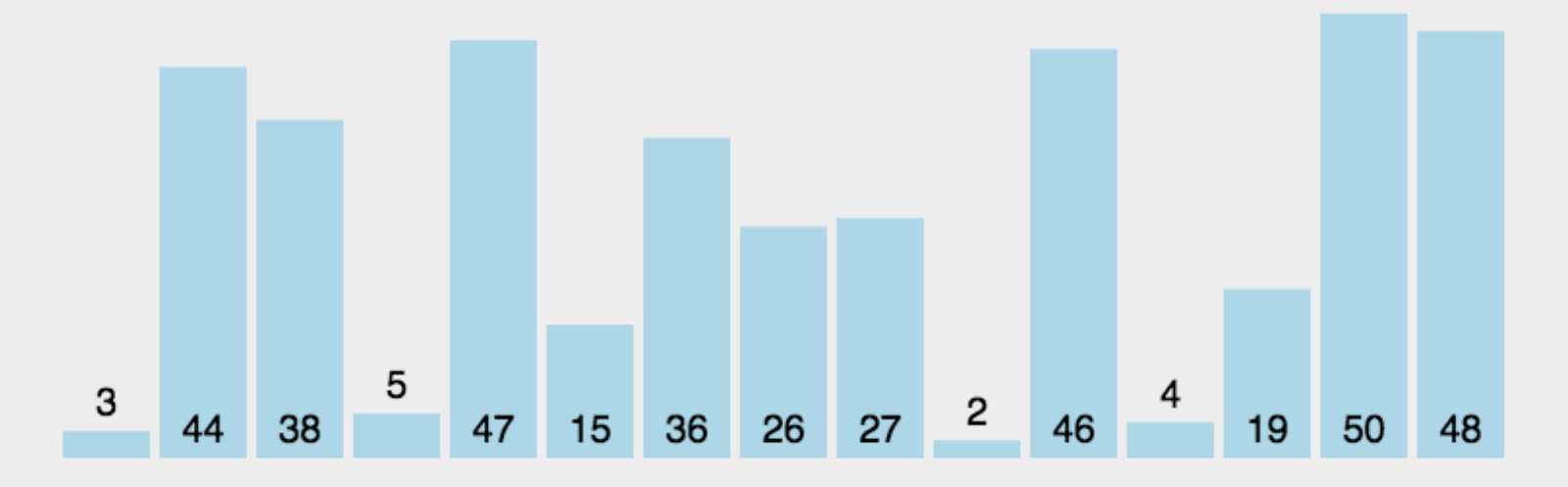
Merge Sort

O(NlogN)


6 5 12 10 9 1
6 5 12
10 9 1
6
5 12
10
9 1
6
5
12
10
9
1
6
5 12
10
1 9
5 6 12
1 9 10
1 5 6 9 10 12

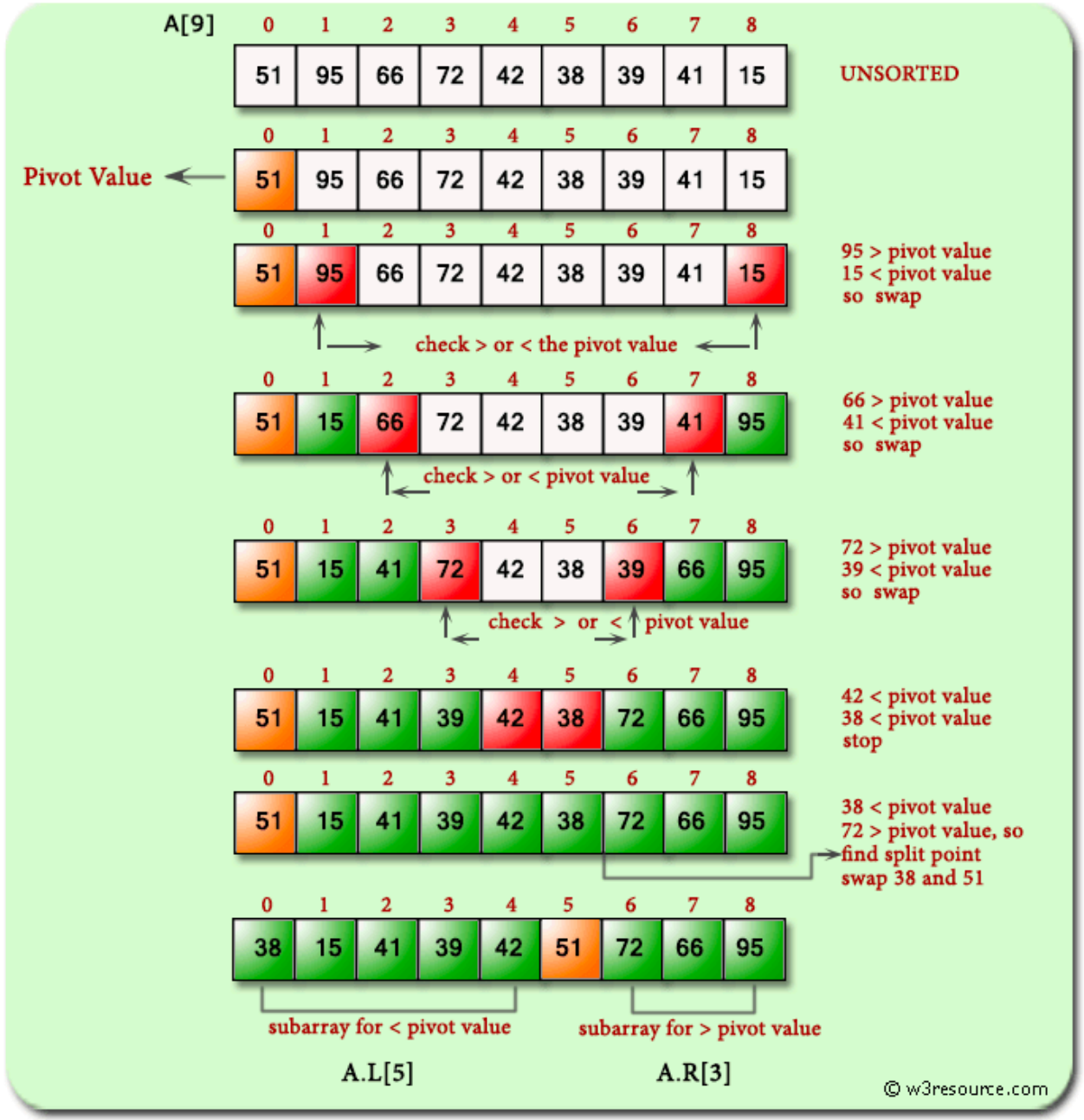
Sorting algorithms and their time complexities

Quick Sort



O(NlogN)

Quick Sort


A[9]
0 1 2 3 4 5 6 7 8
51 95 66 72 42 38 39 41 15
UNSORTED
Pivot Value
95 > pivot value
15 < pivot value
so swap
check > or < the pivot value
66 > pivot value
41 < pivot value
so swap
check > or < pivot value
72 > pivot value
39 < pivot value
so swap
check > or < pivot value
42 < pivot value
38 < pivot value
stop
38 < pivot value
72 > pivot value, so
find split point
swap 38 and 51
subarray for < pivot value
subarray for > pivot value
A.L[5]
A.R[3]
© w3resource.com