

10.5

Divide and Conquer

Divide and Conquer Paradigm

Divide and Conquer is a common and useful type of recursion

Approach

- 1 Break problem instance into smaller instances - divide step
- 2 **Recursively** solve problem on smaller instances
- 3 Combine solutions to smaller instances to obtain a solution to the original instance - conquer step

Question: Why is this not plain recursion?

- 1 In divide and conquer, each smaller instance is typically at least a constant factor smaller than the original instance which leads to efficient running times.
- 2 There are many examples of this particular type of recursion that it deserves its own treatment.

Divide and Conquer Paradigm

Divide and Conquer is a common and useful type of recursion

Approach

- 1 Break problem instance into smaller instances - divide step
- 2 **Recursively** solve problem on smaller instances
- 3 Combine solutions to smaller instances to obtain a solution to the original instance - conquer step

Question: Why is this not plain recursion?

- 1 In divide and conquer, each smaller instance is typically at least a constant factor smaller than the original instance which leads to efficient running times.
- 2 There are many examples of this particular type of recursion that it deserves its own treatment.

Divide and Conquer Paradigm

Divide and Conquer is a common and useful type of recursion

Approach

- 1 Break problem instance into smaller instances - divide step
- 2 **Recursively** solve problem on smaller instances
- 3 Combine solutions to smaller instances to obtain a solution to the original instance - conquer step

Question: Why is this not plain recursion?

- 1 In divide and conquer, each smaller instance is typically at least a constant factor smaller than the original instance which leads to efficient running times.
- 2 There are many examples of this particular type of recursion that it deserves its own treatment.

THE END

...

(for now)