

# Lab02-Divide and Conquer

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2021.

\* If there is any problem, please contact TA Haolin Zhou.

\* Name: \_\_\_\_\_ Student ID: \_\_\_\_\_ Email: \_\_\_\_\_

1. *Recurrence examples.* Give asymptotic upper and lower bounds for  $T(n)$  in each of the following recurrences. Assume that  $T(n)$  is constant for sufficiently small  $n$ . Make your bounds as tight as possible.

(a)  $T(n) = 4T(n/3) + n \log n$

(b)  $T(n) = 4T(n/2) + n^2\sqrt{n}$

(c)  $T(n) = T(n-1) + n$

(d)  $T(n) = 2T(\lfloor \sqrt{n} \rfloor) + \log n$

2. *Divide-and-conquer.* Given an integer array  $A[1..n]$  and two integers  $lower \leq upper$ , design an algorithm using **divide-and-conquer** method to count the number of ranges  $(i, j)$  ( $1 \leq i \leq j \leq n$ ) satisfying

$$lower \leq \sum_{k=i}^j A[k] \leq upper.$$

## Example:

Given  $A = [1, -1, 2]$ ,  $lower = 1$ ,  $upper = 2$ , return 4.

The resulting four ranges are  $(1, 1)$ ,  $(3, 3)$ ,  $(2, 3)$  and  $(1, 3)$ .

- (a) Complete the implementation in the provided C/C++ source code ([The source code \*Code-Range.cpp\* is attached on the course webpage](#)).
- (b) Write a recurrence for the running time of the algorithm and solve it by recurrence tree ([You can modify the figure sources \*Fig-RecurrenceTree.vsdx\* or \*Fig-RecurrenceTree.pptx\* to illustrate your derivation](#)).
- (c) Can we use the Master Theorem to solve the recurrence above? Please explain your answer.
3. *Transposition Sorting Network.* A comparison network is a **transposition network** if each comparator connects adjacent lines, as in the network in Fig. 1.

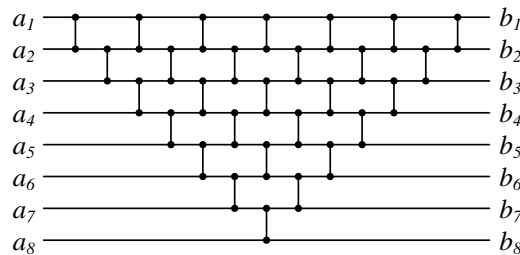


Figure 1: A Transposition Network Example

- (a) Prove that a transposition network with  $n$  inputs is a sorting network if and only if it sorts the sequence  $\langle n, n-1, \dots, 1 \rangle$ . ([Hint: Use an induction argument analogous to the \*Domain Conversion Lemma\*](#).)
- (b) (**Optional Sub-question with Bonus**) Given any  $n \in \mathbb{N}$ , write a program using Tkinter in Python to draw a figure similar to Fig. 1 with  $n$  input wires.