

**CHIANG MAI UNIVERSITY**

**Bachelor of Science (Digital Industry Integration)**

**College of Arts, Media and Technology**

**3rd Semester / Academic Year 2019**

 Data Structure and Algorithm



1. Write a program to find the smallest number in an input array using recursion.
   1. Analyze the best-case time complexity of your code.
   2. Describe the best-case scenario.
   3. Analyze the worst-case time complexity of your code.
   4. Describe the worst-case scenario.
2. Write a program to calculate a Fibonacci using recursion (based on the example from the lecture slide)
   1. Analyze the best-case and worst-case time complexities of your code.
   2. Design an alternative algorithm to calculate a Fibonacci with a time complexity of O(n).
   3. Explain how your new algorithm manages to reduce the time complexity.
3. The *lucky number* of an array is the value of key that occurs the most frequently in the array. For example, in array (2, 4, 3, 2, 5, 2) the *lucky number* is 2.
   1. Design a linear-time algorithm or write a program to compute the *lucky number* of an array consisting of n integers, each of which has a value between 1 and 4n. If there are multiple *lucky numbers*, return only one of them.
   2. Analyze your algorithm to show the worst-case time complexity is O(n).