*# Course: CS261 - Data Structures*

*# Student Name: Khrystian Clark*

*# Assignment: Assignment 2, pt5*

*# Description: Show implementation and understanding of Amortized Analysis*

*# Last revised: 02/01/2021*

1. How many cost units are spent in the entire process of performing  40 consecutive append operations on an empty array which starts out at capacity 3, assuming that the array will *double in capacity each time* a new item is added to an already full dynamic array? As N (i.e., the number of appends) grows large, under this strategy for resizing, what is the amortized big-O complexity for an append?
   * 40 = number of appends being performed
   * 3, 6, 12, 24 = costs when the capacity doubles
   * Cost units = 40+3+6+12+24 = 85
   * Cost spent is 85 units
   * The big-O complexity is O(1), because it has consistent input size
2. How many cost units are spent in the entire process of performing 40 consecutive append operations on an empty array which starts out at capacity 3, assuming that the array will *grow by a constant 2 spaces each time* a new item is added to an already full dynamic array? As N (i.e., the number of appends) grows large, under this strategy for resizing, what is the amortized big-O complexity for an append?
   * 40 = number of appends being performed
   * 3, 5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39= costs when the capacity increases
   * Cost spent is 439 units
   * The big-O complexity is O(n), because increase loop is based solely on input size.