Paul Burgwardt

#0304700

CS344 – Professor Tamon

Homework 0

Analysis

The following was done to calculate the cost of the recursive algorithm.

Cost(N)	N
0	1
1	2
2T(N/2) + 2	> 2

With an input number of values in the array greater than two, the cost can be described as an equation:

$$C(N) = 2C\left(\frac{N}{2}\right) + 2$$

$$C(N) = 2\left(2C\left(\frac{N}{4}\right) + 2\right) + 2$$

$$C(N) = 4C\left(\frac{N}{4}\right) + 6$$

...

$$C(N) = 2^{k-1}C(2) + \sum_{i=1}^{k-1} 2^k$$

$$C(N) = 2^{k-1} + 2^k - 2$$

$$C(N) = \frac{3}{2}N - 2$$

The cost of the recursive algorithm is 3/2N – 2 for N that is a power of two.

True. Comparing the cost and formula values, they are exactly the same when N is a power of two.

The recursive algorithm performs worse than 3/2N – 2 on N that are not powers of two.

True. Comparing the cost and formula values, the cost values are higher (thus more costly) than the formula values.

The following was done to calculate the cost of the non-recursive algorithm.

Cost(N)	N
0	1
1	2
2T(N) - 2	> 2

With an input number of values in the array greater than two, the cost is described as the number of values minus one, since we are just iterating over the array, and comparing each pair of values:

$$C(N) = 2N - 2$$

Your non-recursive algorithm is as good as the recursive algorithm on N that are powers of two.

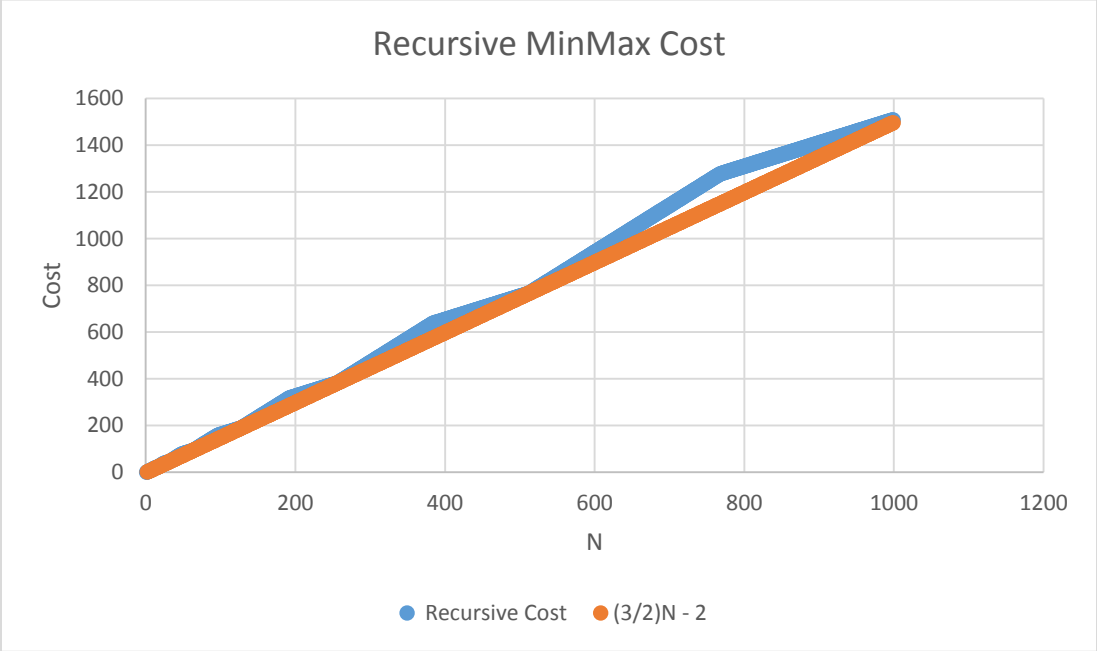
> False. The recursive algorithm outperforms the iterative algorithm, as shown by the plot.

Your non-recursive algorithm outperforms the recursive algorithm on N that are not powers of two.

> False. Comparing the cost values of the recursive and iterative algorithms, the recursive algorithm values are less than those of the iterative algorithm. Thus, the recursive algorithm outperforms the iterative algorithm.

Plot

Recursive



Iterative