**Name: Akhil Montrose**

**Course: Algorithm Analysis and Design**

**Document Type: Assignment 1**

**Student ID: 81788**

**Bubble Sort Algorithm**

bubbleSort (arr a)

arrLength (length of array a)

for i < 1 to i-1 (start and end of array)

let b < 0

for b < 0 to arrLength -1

if a[b] > a[b+1]

then swap a[b] with a[b+1]

i < 1 to i-1 > c1

b < 0 > 1

b < 0 to arrLength -1 > C2n

if a[b] > a[b+1] > n

swap a[b] with a[b+1] > n

So the number of primitive operations are:

C1 + C2n + 1 + n + n = 1 + C1 + C2n + 2n

**Selection Sort Algorithm**

arrLength < length of the array

SelectionSort (Array, arrLength)

For i = 0 to arrLength-2

Minimum = i

For b = i+1 to arrayLength-1

If Array[b] < Array[minimum]

Minimum = j

Swap (Array[b] < Array[minimum]

For i = 0 to arrLength-2 > c1

Minimum = i >1

b = i+1 > 1

b = i+1 to arrayLength-1 > C2n

Array[b] < Array[minimum] > n

Minimum = j > 1

(Array[b] < Array[minimum] > n

So the number of primitive operations are:

C1 + 1 + 1 + C2n + n + 1 + n = 3 + c1 + C2n + 2n