**Growable array**

As we know for a growable array ,the values are copied from the last array each time a new element is inserted so we can write the number of operations as

1 + 2 + 3 +4 + 5 + 6 + 7..+ (N-1) + N

where n is the size of array,as we are dealing with with 2MB file the sum of this series is calculated as

N/2 x(N+1) which is roughly estimated to N square So

2x10^6/2 x (2 x 10^6 +1) = 2.001x 10^12 operations

So time complexity can be written as O(n^2)

**Vector**

For a vector, the capacity of the new created array is doubled which leaves extra space to insert new elements and saves the process of copying for every iteration,which can be calculated as

1 + 2 + 3 + 1 + 5 + 1 + 1 + 1 + 9 + 1 + 1 + 1 + 1 + 1 + 1 + 1 + 17 ....

if we take out the 1 for N number of times,we get

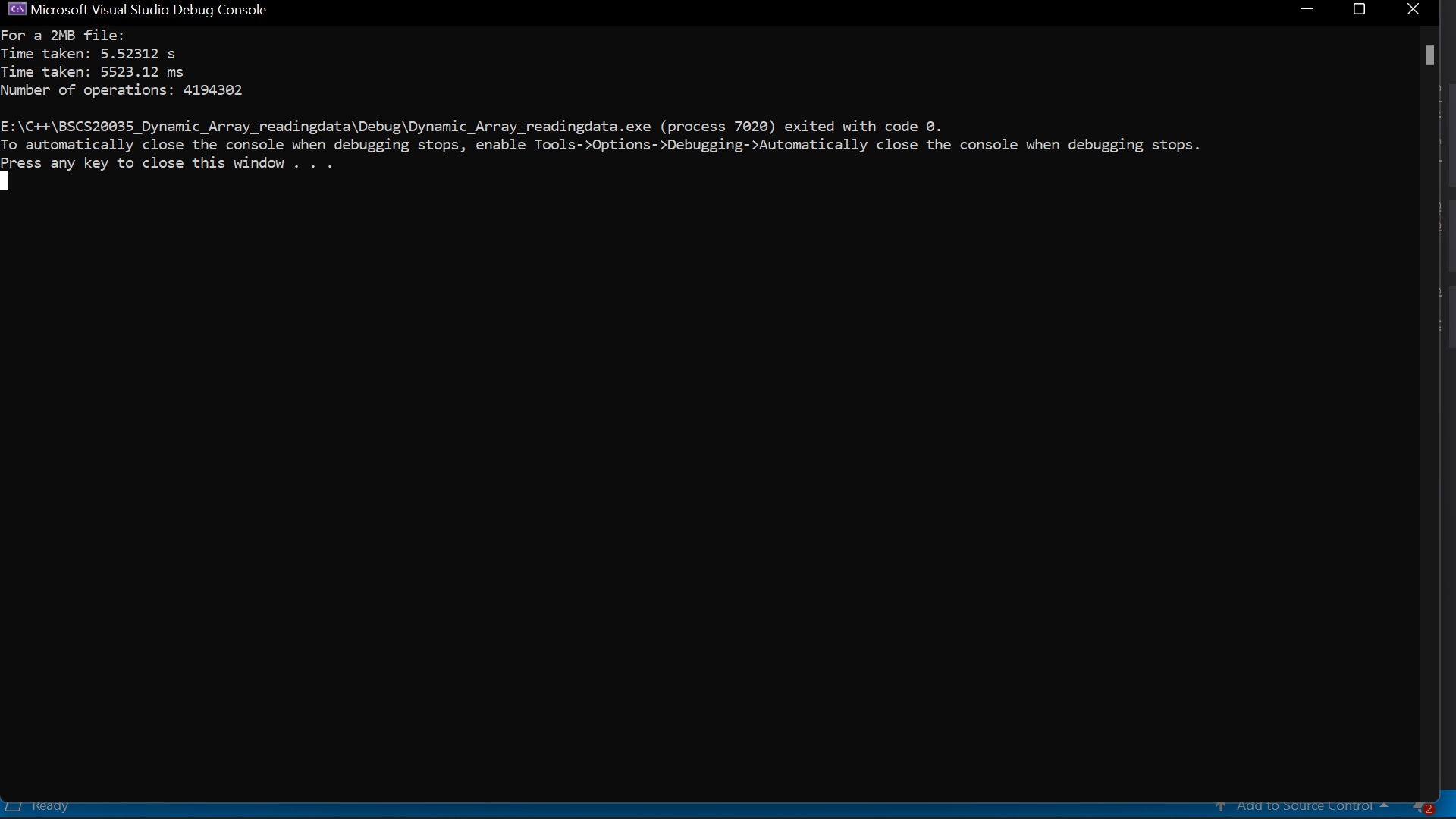
1 + 1 + 1 + 1 + 1 + 1.... = N

and we are left with

0 + 2 + 4 + 8 + 16 + .... N/4 + N/2 + N <= 2N

so the total cost is 2N + N <= 3N

which can be written as O(N)



From the above picture we can see that the number of operations is less than 3N which is equal to

3 (2 x10^6) = 6000000 which proves the time complexity is less than 3N

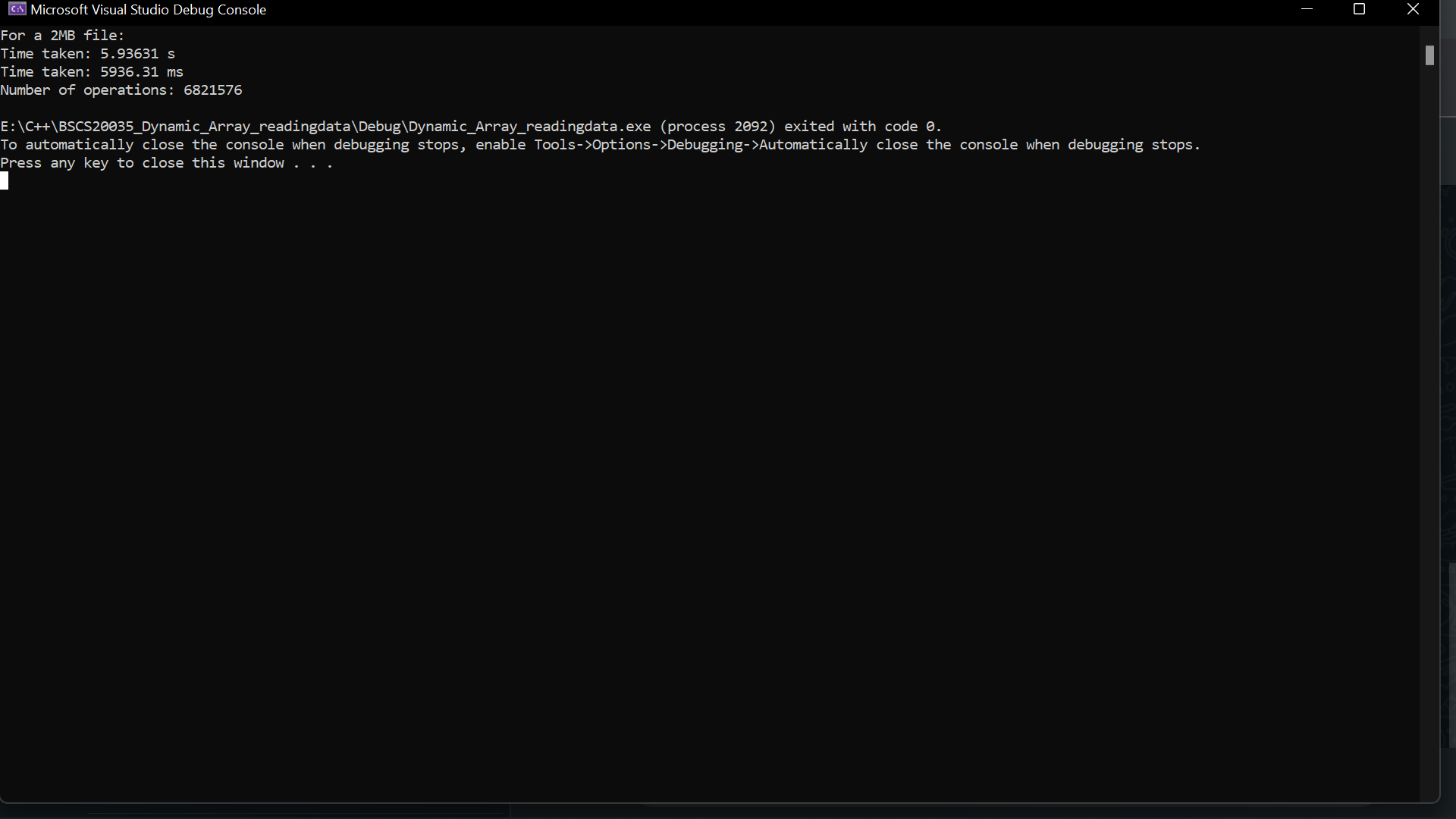
**ArrayList**

For an array list, the size of array is increased by 1.5 which decreases the memory usage but increases the time it takes to load the data,the Complexity is calculated using the geometric sum formula

N(K)+1/(K-1) where N is and K is the factor by which it increases.This formula is roughly estimated to be O(n).

BY putting the values we get

(2x10^6)(1.5) +1 / 1.5 -1 which is roughly equal to 6000000



We can see that the number of operations is roughly equal to the calculated time complexity . Since the capacity factor is less than vector, the number of operations must be greater than the vector for the same file which can be seen from the above pictures.