Sai gandham @01499559

Introduction to Algorithms Project Report

1)Shellsort is an in-place comparison sort. The idea is to arrange the list of elements so that, starting anywhere considering every hth element gives a sorted list.
For the given array A={9,8,7,6,5,4,3,2,1}. Value of the array after each iteration of for loop

9	8	7	6	5	4	3	2	1
2	8	7	6	5	4	3	**9**	1
2	**1**	7	6	5	4	3	9	**8**
2	1	7	6	5	4	3	9	8
2	1	7	6	5	4	3	9	8
2	1	**4**	6	5	**7**	3	9	8
2	1	4	**3**	5	7	**6**	9	8
2	1	4	3	5	7	6	9	8
2	1	4	3	5	7	6	9	8
1	2	4	3	5	7	6	9	8
1	2	4	3	5	7	6	9	8
1	2	**3**	**4**	5	7	6	9	8
1	2	3	4	5	7	6	9	8
1	2	3	4	5	7	6	9	8
1	2	3	4	5	**6**	**7**	9	8
1	2	3	4	5	6	7	9	8
1	2	3	4	5	6	7	**8**	**9**

Sai Gandham @01499559

2) loop invariants:

a) first while loop produces h values. All h values are less than n i.e., Array size. All the hvalues produced are odd numbers. The i th iteration "h" value is equals to floor ((i+1 th iteration " h")/2) value.

b) Second while loop invariants: h is always greater than 0, h ordered sorted sets are formed at the end of each loop.

c) For loop: for every iteration A[j] is picked out of array. h separated sub array is produced.

d) Starting in position j-h, elements are successively moved h positions to the right until proper position for A[j] is found.

3)proof by contradiction .
Given condition B[i]<=C[i] for all 1<=i<=m before sorting ----------condition #1
Asked to prove B[i]<=C[i] condition holds true after sorting for all 1<=i<=m.
Lets assume that B[i]>C[i] for all 1<=i<=m after sorting
Then it means that B[i] value >= i elements of C for all 1<=i<=m---------2
But as per the 1^{st} condition it gets that any i elements of C >= any i elements of B
Here we gets that B[i]> i elements of B, which is not true for the sorted order
Therefore B[i]<=C[i] holds true after sorting

4)proof by contradiction
Given condition B[i]<=C[i+j] for 1<=i<=m-j before sorting ----------condition #1
Asked to prove B[i]<=C[i+j] for 1<=i<=m-j holds true after sorting
Lets assume that B[i]>C[i+j] for all 1<=i<=m-j after sorting,
From the 1^{st} condition we gets that any i C elements are greater than or equals to any i B elements
From that we gets that B[i]> i+j C elements> i B elements , it is a fallacy as B[i] cant be greater than i B elements after sorting.
There fore B[i]<=C[i+j] holds true after sorting

5)Given K>h>1. To show A remains K-ordered after A is h-sorted. Have to prove A[i]<=A[i+k] remains same after sorting h gap elements of Array A
we take a difference of h on both sides to show that A remains K ordered after h sorted.
B[]=….A[i-h],A[i],A[i+h]…..and C[]=…..A[i+k-h],A[i+k],A[i+k+h]…..
Have to prove The above sequences are independently sorted the condition still holds
i.e., B[i]<=C[i+k] for values 1….i-h<=i<=i+h…m--------condition 1
By contradiction : lets assume that B[i]>C[i+k] after sorting
It means that B[i] is greater than i+k elements of sorted array C.
As per 1^{st} condition we gets that any i C elements are greater than or equal to any i B elements.
From that we gets that B[i]>i+k C elements > I B elements , it is a not possible as B[i] can not be greater than I B elements after sorting.

Sai gandham @01499559

Therefore B[i]<=C[i+k] holds true

EMPIRICAL ANALYSIS

For Empirical analysis I used 4 different gap sequences

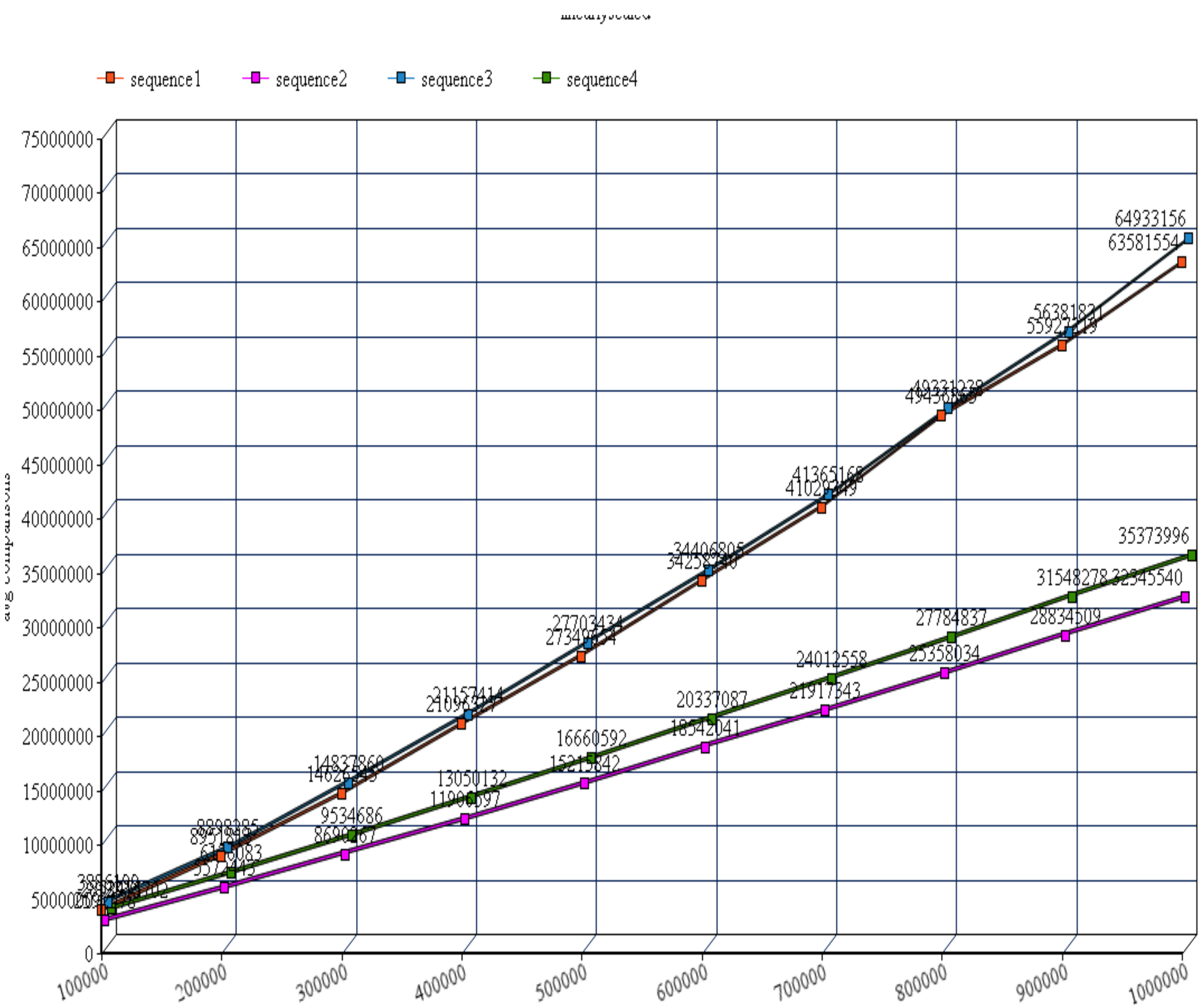
Sequence1: h=2*h+1

Sequence2: $\left\lceil \frac{9^k - 4^k}{5 \cdot 4^{k-1}} \right\rceil$

Sequence3: $(3^k - 1)/2$

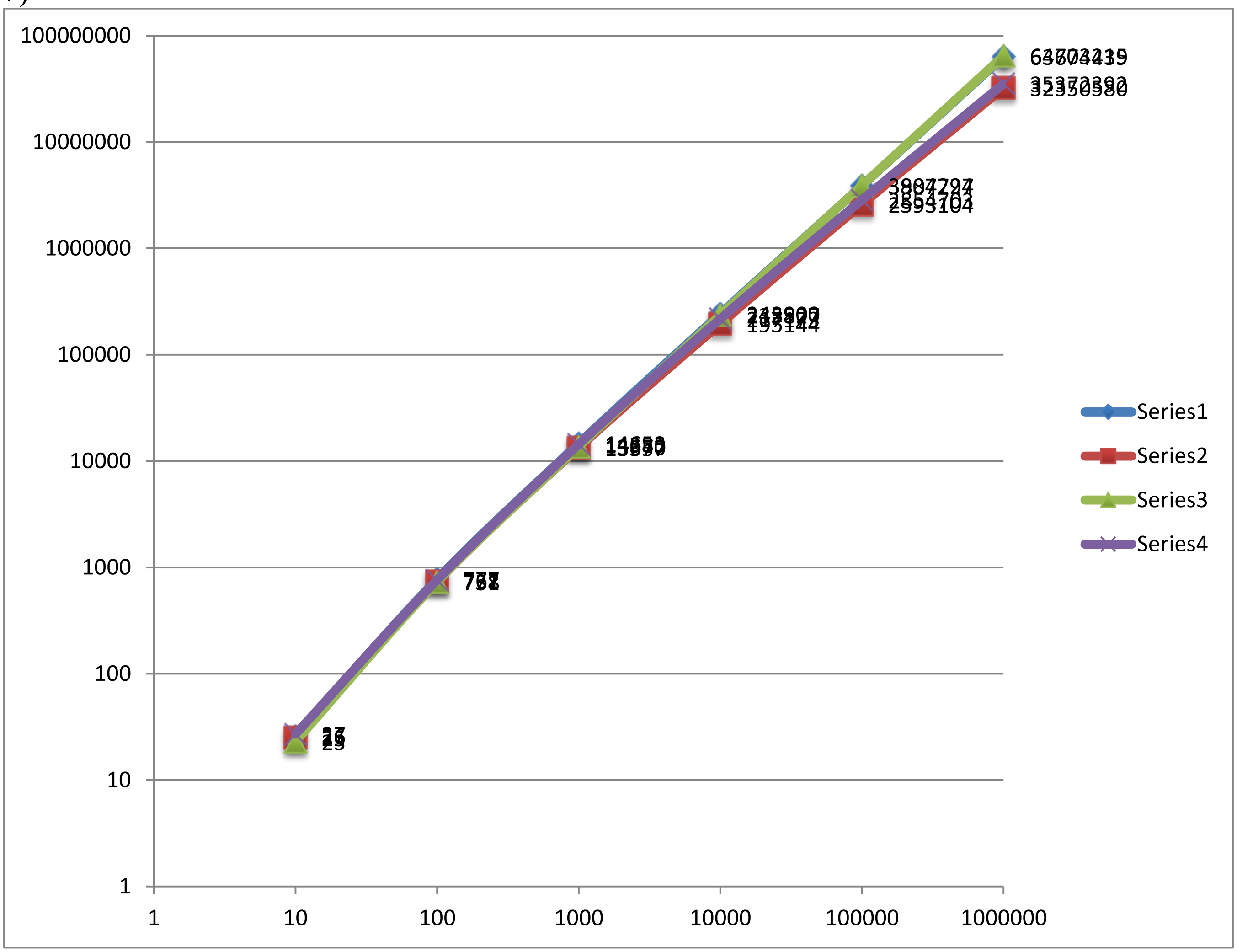
My Sequence 4: e^i.

6)

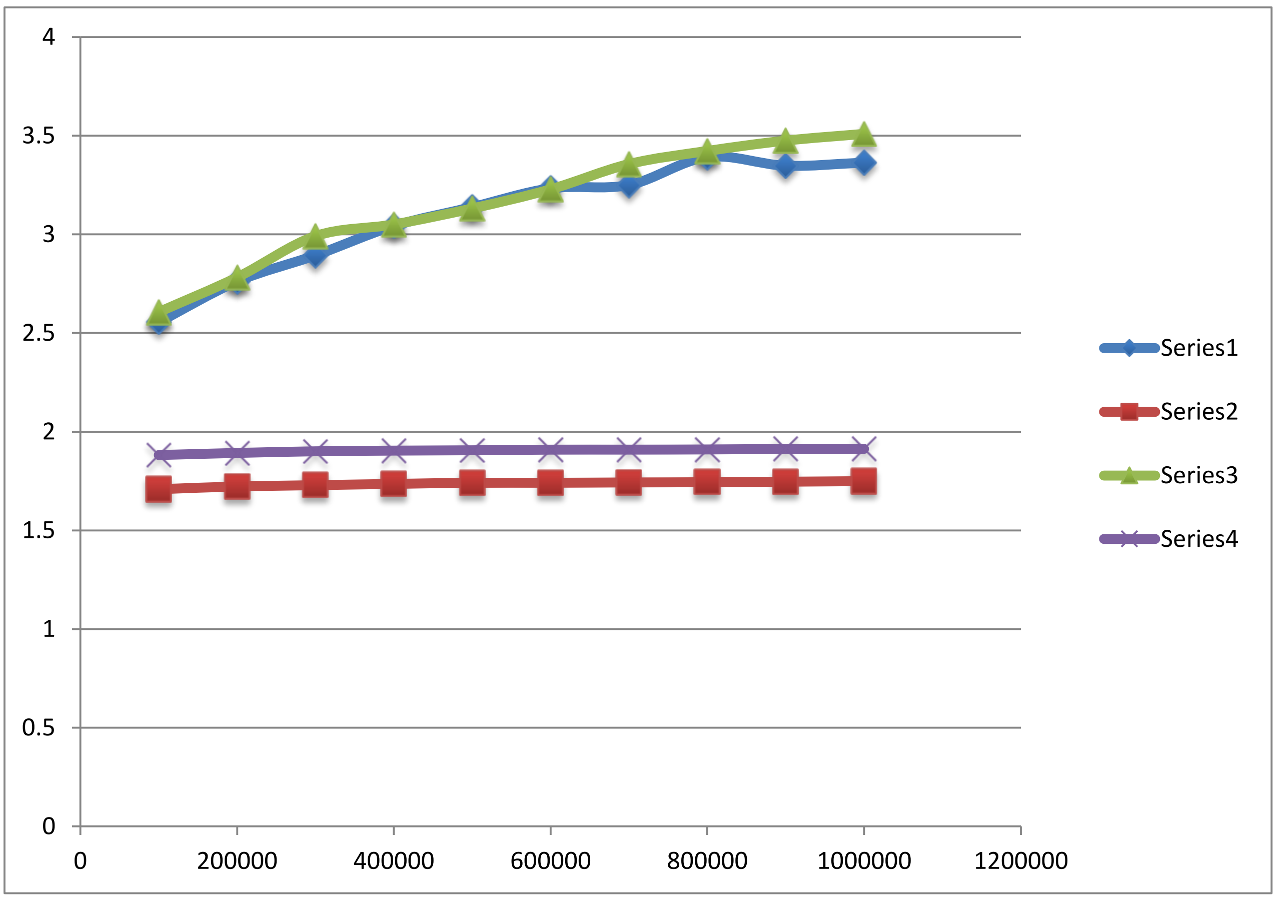

sequence1
sequence2
sequence3
sequence4
75000000
70000000
65000000
60000000
55000000
50000000
45000000
40000000
35000000
30000000
25000000
20000000
15000000
10000000
0
100000
200000
300000
400000
500000
600000
700000
800000
900000
1000000
64933156
63581554
41365168
34406805
27703434
21157414
14837860
35373996
31548278
32345540
27784837
28834609
24012558
25358034
20337087
21917343
16660592
18542041
15215842
13050132
9534686

Sai gandham @01499559

7)



8)


4
3.5
3
2.5
2
1.5
1
0.5
0
0
200000
400000
600000
800000
1000000
1200000
Series1
Series2
Series3
Series4

Sai gandham @01499559

9)