**Design and Analysis of Algorithm (V semester) Assignment -1**

1. Write an algorithm to generate the primes till a given limit,
   1. Using sieve of Eratosthenes
   2. Using sieve of Sundaram

Compare both the algorithms by counting the number of times the basic operation is executed to find out which algorithm is better.

1. Consider the algorithm for the sorting problem that sorts an array using shell sort, the method starts by sorting elements far apart from each other and progressively reducing the gap between them, to get the sorted array as a result:

**Algorithm ShellSort(A[0..n-1])**

//Sorts an array using shell sort

//Input: Array A[0..n − 1]

//Output: Array A[0..n − 1] of A’s elements sorted in ascending order

d 🡨 length

do

d 🡨 (d + 1)/2

for i🡨0 to (length - d) do

if A[i + d] < A[i]

tmp🡨 A[i+d]

A[i + d] 🡨 A[i]

A[i] 🡨 tmp

while d > 1

* 1. Apply this algorithm to sorting the list 60, 35, 81, 98.
  2. Is this algorithm stable?
  3. Is it in place?

1. Let A be the adjacency matrix of an undirected graph. With an example for each, explain what property of the matrix indicates that
   1. The graph is complete
   2. The graph has a loop, i.e., an edge connecting a vertex to itself
   3. The graph has an isolated vertex, i.e., a vertex with no edges incident to it.

Answer the same questions for adjacency list representation.

1. Let (i) f(n)= 10n3+5

(ii) f(n)= 6\*2n+n2.

Express f(n) using Big Oh, Big Omega and Big Theta notations.

1. Consider the following recursive algorithm:

Algorithm Value(A[0..n-1])

// Input: An array A[0..n-1] of real numbers

//Output:

if n=1

return A[0]

else

temp🡨Value(A[0..n-2])

if temp≤A[n-1]

return temp

else

return A[n-1]

* 1. What does this algorithm compute?
  2. Set up a recurrence relation for the algorithm’s basic operation count and solve it.