**Source Code of Longest Increasing Sequence**

**public** **class** Longestincreasingsubsequence {

/\* lis() returns the length of the longest

increasing subsequence in arr[] of size n \*/

**static** **int** lis(**int** arr[], **int** n)

{

**int** lis[] = **new** **int**[n];

**int** i, j, max = 0;

/\* Initialize LIS values for all indexes \*/

**for** (i = 0; i < n; i++)

lis[i] = 1;

/\* Compute optimized LIS values in

bottom up manner \*/

**for** (i = 1; i < n; i++)

**for** (j = 0; j < i; j++)

**if** (arr[i] > arr[j] && lis[i] < lis[j] + 1)

lis[i] = lis[j] + 1;

/\* Pick maximum of all LIS values \*/

**for** (i = 0; i < n; i++)

**if** (max < lis[i])

max = lis[i];

**return** max;

}

**public** **static** **void** main(String args[])

{

**int** arr[] = { 10, 22, 9, 33, 21, 50, 41, 60 };

**int** n = arr.length;

System.***out***.println("Length of lis is " + *lis*(arr, n)

+ "\n");

}

}