LIS Source Code:

**package** Datastructures;

**public** **class** Demo

{

**static** **int** incre\_subseq(**int** my\_arr[], **int** arr\_len)

{

**int** seq\_arr[] = **new** **int**[arr\_len];

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

**for** (i = 0; i < arr\_len; i++)

seq\_arr[i] = 1;

**for** (i = 1; i < arr\_len; i++)

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

**if** (my\_arr[i] > my\_arr[j] && seq\_arr[i] < seq\_arr[j] + 1)

seq\_arr[i] = seq\_arr[j] + 1;

**for** (i = 0; i < arr\_len; i++)

**if** (max < seq\_arr[i])

max = seq\_arr[i];

**return** max;

}

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

{

**int** my\_arr[] = { 1, 2, 2, 4, 4, 6, 7, 8};

**int** arr\_len = my\_arr.length;

System.***out***.println("The length of the longest increasing subsequence is " + *incre\_subseq*(my\_arr, arr\_len));

}

}

/\*

OUTPUT:

The length of the longest increasing subsequence is 6

\*/