**Largest increasing subsequence source code:**

**package** longestincresingsubsequence;

**public** **class** LISubsequence {

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

//new array lis is created to store count for each sequence

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

**int** k, k2, max = 0;

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

lis[k] = 1;

//comparing each element of given array with the rest elements of an array

**for** (k = 0; k < n; k++) {

**for** (k2 = 0; k2 < k; k2++)

**if** (arr[k] > arr[k2] && lis[k] < lis[k2] + 1)//whenever next bigger element found count is increased

lis[k] = lis[k2] + 1;

}

// array of count of each increasing subsequence possible

**for** (k = 0; k < n; k++) {

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

max = lis[k];

}

**return** max;

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**package** longestincresingsubsequence;

**public** **class** TestLIS {

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

//int arr[] = { 1,2,3,4 };

**int** arr[] = { 10, 17, 12, 15, 20,9,11,14,19,23,55,66 };

**int** n = arr.length;

System.***out***.println("largest increasing subsequence length is=" + longestincresingsubsequence.LISubsequence.*lis*(arr, n));

}

}