**import** java.util.Arrays;

**public** **class** Demo {

**public** **static** **int**[] largestSubsquence(**int**[] arr) {

**int** n = arr.length;

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

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

**int** maxLen = 0;

**int** endIndex = -1;

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

lisLengths[i] = 1;

previousIndices[i] = -1;

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

**if** (arr[i] > arr[j] && lisLengths[j] + 1 > lisLengths[i]) {

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

previousIndices[i] = j;

}

}

**if** (lisLengths[i] > maxLen) {

maxLen = lisLengths[i];

endIndex = i;

}

}

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

**int** curInd = maxLen - 1;

**while**(endIndex != -1) {

lis[curInd] = arr[endIndex];

curInd--;

endIndex = previousIndices[endIndex];

}

**return** lis;

}

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

**int**[] arr = {10,22,9,33,21,50,80,60};

**int**[] res = *largestSubsquence*(arr);

System.***out***.println("Original Array");

System.***out***.println(Arrays.*toString*(arr));

System.***out***.println("Array with longest increasing subsequence");

System.***out***.println(Arrays.*toString*(res));

}

}