

Solution 1

```
1  # Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3  # O(n^2) time | O(n) space
4  def maxSumIncreasingSubsequence(array):
5      sequences = [None for x in array]
6      sums = [num for num in array]
7      maxSumIdx = 0
8      for i in range(len(array)):
9          currentNum = array[i]
10         for j in range(0, i):
11             otherNum = array[j]
12             if otherNum < currentNum and sums[j] + currentNum >= sums[i]:
13                 sums[i] = sums[j] + currentNum
14                 sequences[i] = j
15             if sums[i] >= sums[maxSumIdx]:
16                 maxSumIdx = i
17     return [sums[maxSumIdx], buildSequence(array, sequences, maxSumIdx)]
18
19
20 def buildSequence(array, sequences, currentIdx):
21     sequence = []
22     while currentIdx is not None:
23         sequence.append(array[currentIdx])
24         currentIdx = sequences[currentIdx]
25     return list(reversed(sequence))
26
```