AlgoExpert Quad Layout JavaScript 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1

```
1\, // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
 3 // O(nc) time | O(nc) space
 4 function knapsackProblem(items, capacity) {
      const knapsackValues = [];
      for (let i = 0; i < items.length + 1; i++) \{
        const row = new Array(capacity + 1).fill(0);
 8
        knapsackValues.push(row);
 9
10
      for (let i = 1; i < items.length + 1; i++) {</pre>
        const currentWeight = items[i - 1][1];
11
        const currentValue = items[i - 1][0];
12
        for (let c = 0; c < capacity + 1; c++) {</pre>
13
          if (currentWeight > c) {
14
15
            knapsackValues[i][c] = knapsackValues[i - 1][c];
16
          } else {
17
            knapsackValues[i][c] = Math.max(
              knapsackValues[i - 1][c],
knapsackValues[i - 1][c - currentWeight] + currentValue,
18
19
20
21
22
23
24
      \textbf{return} \hspace{0.2cm} \texttt{[knapsackValues[items.length][capacity], getKnapsackItems(knapsackValues, items)];} \\
25
26
27 function getKnapsackItems(knapsackValues, items) {
28
      const sequence = [];
29
      let i = knapsackValues.length - 1;
30
      let c = knapsackValues[0].length - 1;
      while (i > 0) {
       if (knapsackValues[i][c] === knapsackValues[i - 1][c]) {
32
33
         i -= 1;
34
          sequence.unshift(i - 1);
35
36
          c -= items[i - 1][1];
37
          i -= 1;
38
        if (c === 0) break;
39
40
41
      return sequence;
42 }
43
44 exports.knapsackProblem = knapsackProblem;
```