

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 // O(nc) time | O(nc) space
4 function knapsackProblem(items, capacity) {
5   const knapsackValues = [];
6   for (let i = 0; i < items.length + 1; i++) {
7     const row = new Array(capacity + 1).fill(0);
8     knapsackValues.push(row);
9   }
10  for (let i = 1; i < items.length + 1; i++) {
11    const currentWeight = items[i - 1][1];
12    const currentValue = items[i - 1][0];
13    for (let c = 0; c < capacity + 1; c++) {
14      if (currentWeight > c) {
15        knapsackValues[i][c] = knapsackValues[i - 1][c];
16      } else {
17        knapsackValues[i][c] = Math.max(
18          knapsackValues[i - 1][c],
19          knapsackValues[i - 1][c - currentWeight] + currentValue,
20        );
21      }
22    }
23  }
24  return [knapsackValues[items.length][capacity], getKnapsackItems(kna
25 }
26
27 function getKnapsackItems(knapsackValues, items) {
28   const sequence = [];
29   let i = knapsackValues.length - 1;
30   let c = knapsackValues[0].length - 1;
31   while (i > 0) {
32     if (knapsackValues[i][c] === knapsackValues[i - 1][c]) {
33       i -= 1;
```

Solution 1 Solution 2 Solution 3

```
1 function knapsackProblem(items, capacity) {
2   // Write your code here.
3   // Replace return below.
4   return [
5     10, // total value
6     [1, 2], // item indices
7   ];
8 }
9
10 // Do not edit the line below.
11 exports.knapsackProblem = knapsackProblem;
12
```

Our Tests

Custom Output

Submit Code



Run or submit code when you're ready.