AlgoExpert   Quad Layout   C++   14px   Sublime   Monokai   00:00:

Our Solution(s)   Run Code

Solution 1

```cpp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

#include <vector>
using namespace std;

vector<vector<int>> getKnapsackItems(vector<vector<int>> knapsackValu
                                     vector<vector<int>> items, int w

// O(nc) time | O(nc) space
vector<vector<int>> knapsackProblem(vector<vector<int>> items, int ca
  vector<vector<int>> knapsackValues(items.size() + 1,
                                     vector<int>(capacity + 1, 0));
  for (int i = 1; i < items.size() + 1; i++) {
    int currentWeight = items[i - 1][1];
    int currentValue = items[i - 1][0];
    for (int c = 0; c < capacity + 1; c++) {
      if (currentWeight > c) {
        knapsackValues[i][c] = knapsackValues[i - 1][c];
      } else {
        knapsackValues[i][c] =
            max(knapsackValues[i - 1][c],
                knapsackValues[i - 1][c - currentWeight] + currentValu
      }
    }
  }
  return getKnapsackItems(knapsackValues, items,
                          knapsackValues[items.size()][capacity]);
}

vector<vector<int>> getKnapsackItems(vector<vector<int>> knapsackValue
                                     vector<vector<int>> items, int we
  vector<vector<int>> solution = {{}, {}};
  int i = knapsackValues.size() - 1;
```

Your Solutions   Run Code

Solution 1   Solution 2   Solution 3

```cpp
#include <vector>
using namespace std;

vector<vector<int>> knapsackProblem(vector<vector<int>> items, int cap
  // Write your code here.
  // Replace return below.
  return {
      {10},   // total value
      {1, 2}, // item indices
  };
}
```

Our Tests

[illegible]

Custom Output   Submit Code

```
[illegible]
```

Run or submit code when you're ready.