

Imperial College London – Department of Computing

MSc in Computing Science

580: Algorithms  
Tutorial: Dynamic Programming

1. A thief can carry  $k$  kilograms of loot in his *knapsack*. He robs a shop containing  $N$  items. Item  $i$  is worth  $b_i$  bitcoin and weighs  $k_i$  kilos. The thief wants to decide which items to take to maximise the total value he steals.
  - (a) How would you decompose this problem into subproblems? Does the problem have optimal substructure and overlapping subproblems?
  - (b) Write an algorithm that, given an array  $B$  such that  $B[i]$  is the value of item  $i$  and an array  $K$  such that  $K[i]$  is the weight of item  $i$ , and a maximum weight  $k$ , solves the problem in  $\mathcal{O}(kN)$  time.
  - (c) Since he is greedy, the thief attempts to use the following strategy: the next item chosen should always be the one with the greatest value per kilogram, from those remaining. Show that this strategy is not guaranteed to give the optimal solution.

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs