SEMINAR

Optimisation Problems

Week 09 – Seminar 5

Problem 1: *Can I have some change, please!*
Design an algorithm finding the minimum number of coins n that add up to a given amount of money M. For simplicity, we assume we have an infinite number of coins of each denomination (£2, £1, 50p, 20p, 10p, 5p, 2p, 1p). You could use brute force to solve the problem, but there is a better alternative that you probably used yourself in your everyday shopping.

Does this algorithm work if you do not have an infinite number of coins? For example, using the coins you have in a till?

Problem 2: *Everybody knows the moon is made of cheese...*
I am going back to France for the Christmas season, and I am aiming to bring lots of cheese back to York. The problem is that I can only carry W grams of cheese in my suitcase. I have bought N pieces of cheeses, each summarised buy a pair (w_i, p_i) where w_i is the weight of that piece of cheese in grams, and p_i is the price I paid for that cheese. I want to pack my suitcase with cheese (have a thought for my neighbours on the Eurostar), so the total weight of cheese is less or equal to W, and I maximise the value of the content of my suitcase. Furthermore, I am allowed to cut a piece of cheese into smaller portions to maximise my solution.

Design an algorithm that returns the maximum value I can get from the content of my suitcase.

For example, assuming I bought three loaves of cheese, a Beaufort (3000grams, £150), a Comte (2000grams, £80) and a Tomme de Savoie (3000grams, £90), and a maximum weight allowance for the suitcase equal to 6000grams. As I am allowed to cut a portion of a loaf, the maximum value of my suitcase is $1 \times 150 + 1 \times 80 + {}^{1}\!/_{3} \times 90 \; = \; 260$. I can achieve the maximum value by taking the entire Beaufort, the entire Comte and a third of the Tomme de Savoie.

Problem 3: *Sacrebleu! No wine with my cheese?*
Being back in York with my cheese, I thought it would be nice to have some wine with it. I have asked my family to send me some bottles of wine in a parcel. My family bought several bottles, each bottle v_i containing a volume of wine q_i in centilitres and having a price per litre p_i (in euros). This time we cannot put a portion of a bottle in the parcel, either we put the whole bottle in, or we don't.

The quantity q_i may differ from one bottle to another. For example, a bottle of size “Chopine” contains 25cl while a “Melchior” contains 1800cl (yes you read right 18 litres) and a Magnum contains 150cl. For simplicity, we assume that one litre of wine weighs one kilogram and we ignore the weight of the glass bottle. Again, I would like to optimize my parcel to achieve the highest possible value in Euros given a maximum allowance W in grams.

Could we reuse the algorithm from problems 1 or 2? If not, why (give a counter example)?

If none of them are suitable, could you design another algorithm that will provide the correct answer?