Solutions for Assignment 3, CS206

In response to student requests, I'm compiling solutions for assignment #3. These solutions are intended only to help you to figure out the right answers: in particular, they don't always offer the level of detail or formality expected of your *solutions.*

1a) This was a trick question. Alice values all subsets except the empty subset, so she will have to list all 2^n-1 of them. The bid for subsets of size 1 will be p_1, for subsets of size 2 it will be $p_1 + p_2$, and for subsets of size 3 or more it will be $p_1 + p_2 + p_3$.

b) Alice's valuation can't be represented, because it involves substitutabilities.

c) Alice's bid should take the following form, where the goods are g_1 through g_n, and where d_1, d_2 and d_3 are dummy goods: $(g_1 d_1, p_1)$ OR $(g_1 d_2, p_2)$ OR $(g_1 d_3, p_3)$ OR … OR $(g_n d_1, p_1)$ OR $(g_n d_2, p_2)$ OR $(g_n d_3, p_3)$. The effect here is that only one of the bids for p_1 can be chosen; likewise only one of the bids for p_2 and for p_3. The size of this representation is $3n$.

d) This solution is essentially the same as in part (c). The bid takes the form: $[(g_1, p_1)$ XOR … XOR $(g_n, p_1)]$ OR … OR $[(g_1, p_3)$ XOR … XOR $(g_n, p_3)]$. Again, the size is $3n$.

e) Here there is really nothing clever that you can do. In fact, most attempts to be clever on this question were less compact than the most obvious thing, and so lost points. The obvious approach is to forget about the OR's entirely, and represent the bids as a bunch of XOR'ed offers. Alice will make offers for all subsets of size 1, 2 or 3, with a total size of $\binom{n}{1} + \binom{n}{2} + \binom{n}{3}$.

2) The optimal allocation was A-5, B-3, C-5, D-1 for \$88. To calculate payments, you had to solve the optimization problem again dropping each of the bidders in turn. Dropping A, the best allocation was B-3, C-3, D-1 for \$86. The total declared utility for bidders other than A in the optimal allocation was \$80, since A's declared utility was \$8. Hence A's payment is \$80 – \$86 = -\$6: in other words, A must *pay* \$6. Likewise for B, the best allocation was C-1, D-1 for \$82, and B pays \$82 - \$60 = \$22 (note reversal of the sign from now on). Without C the best allocation is A-5, B-1, D-1, for \$87; C pays \$87 - \$76 = \$11. Without D the best allocation is A-1, B-3; D pays \$73 – \$48 = \$25.

3) The main problem people had with this question was to try to say what D *should* have done: of course, D doesn't actually know what bids the other bidders placed, so you can't claim that he should have bid \$26. (Furthermore, the other bidders wouldn't have submitted those bids under naïve payments anyway.) You were just intended to argue that D was *not* best off bidding his true valuation. Note that if he does bid this way and wins, he gains 0 since he pays his utility; if he loses he also gains 0. Hence his expected gain is 0. On the other hand, if he bids a bit less than his valuation he will have positive gain with non-zero probability. Hence D is not best off bidding his valuation.

4) Here's how to reduce the problem to maximal weighted matching. Create one node for every good. For bids on pairs of goods, create an edge between the goods with a weight corresponding to the price. This takes care of bids on pairs of goods because the maximal weighted matching will keep the edges with maximal weights, subject to the constraint that each good can only belong to one winning bid. However, we have no way of representing bids on a single good, because there aren't two nodes to connect in this case. Two approaches make sense: (a) create a self-loop for the good node in this case; (b) create a new node for this bid, and create an edge from the good to the new bid node. Note that creating a node for each *bidder* doesn't work, since this prevents a single bidder from winning with two single-good bids.

5) This question was graded based on whether you said things that were true, rather than on whether you said things that appear in my list. I gave detailed comments for everyone, so hopefully you can figure out why you lost points. I deducted between 1 and 3 points for each comment I made, depending on the severity of the misunderstanding you demonstrated. Anyway, I'll give an example of an answer that I would have given full credit:

MMV Advantages:

1) Economically efficient allocation: the most important packets always get through.
2) It is a dominant strategy for bidders to bid truthfully: this makes bidders' strategic job easy, drawing on a well-understood auction mechanism (VCG)
3) When there is no congestion, bidders are not charged to use the network. This is an advantage because many people believe that the internet should be free, and this mechanism allows free traffic up to the available capacity of the network.

MMV Disadvantages:

1) Incredibly difficult to implement: routers would have to conduct auctions, maintain billing information. Really, there is no hope that this could ever happen.
2) Prices charged to users could fluctuate wildly. Past cost would not be indicative of future cost. Users like to know what they will have to pay.
3) Even disregarding (2), bidders may not *know* their own valuations exactly, in which case they will have a hard time bidding. A related point is that this mechanism is counterintuitive for many people.

PMP Advantages:

1) Very simple to implement: few changes are required in routers.
2) Very simple to for users to understand. Prices are fixed in advance, and expected load in each channel may be inferred from past performance.
3) PMP is compatible with the idea that free traffic should always be accommodated on the internet. Unlike MMV, which forces free traffic out when congestion occurs, the designer of a PMP network could guarantee a certain percentage of bandwidth to free traffic.

PMP Disadvantages:

1) PMP is economically inefficient. If the most expensive channel is full, there is nothing else that an agent can do, even if he is willing to pay more than the cost of the channel.

2) When demand for a PMP network exceeds its bandwidth, we get the same sorts of congestion problems that the internet experiences today: everybody gets degraded service. This is bad for users that are paying, because despite the payment they do not receive a guaranteed QoS.
3) Although PMP may balance the load in the long term, it deals very badly with short-term effects. For example, if usage spikes in the free channel, a lot of users may move up by one channel to try to improve their service quality, congesting the higher channel and paradoxically uncongesting the free channel.

Your opinion: You could say just about anything here, as long as it wasn't *wrong* (e.g., if you had written that you preferred PMP because it guaranteed paying users a certain QoS, you would have lost 3 points).

Finally, these are the notes that I made to myself to explain my grading on your assignments. They may be helpful to you, so here they are:

Problem 1:
A: part c: used only one dummy good
B: part e: an over-complicated scheme that was not as compact as the one given above. For example, you tried to use real goods to emulate dummy goods.
C: parts d, e: used dummy goods
D: part a: failed to list all subsets having non-zero valuations
E: part e: solution not as compact as the one given above.

Problem 2:
A: right allocation, wrong payments

Problem 3:
A: claim that D *will* bid $26 (or $25 or $27), implying that he has knowledge that he does not. Or otherwise not addressing the fact that you are being asked to consider a disequilibrium argument.
B: your solution doesn't resemble an equilibrium analysis.
C: you claim that D should bid less than his valuation, but you do not adequately explain why.

Problem 4:
A: no single-good bids
B: picked the best bid for each good, but neglected to notice that this allows two conflicting bids to be selected
C: violated the requirement that in a solution no two edges may share an end-node

Problem 5:
As described above, these were graded on a case-by-case basis.