CS386 Assignment 5

Due: Wednesday 10/21 10PM ET

1. Do 1 of the following (all of the following from Nisan et al.'s text). For each scenario below, I am asking you to model design a VCG mechanism and determine

   - The set $\mathcal{A}$ of *alternatives*: the possible outcomes
   - The valuation functions of all players
   - The prices to assign each player using Clarke-Pivot Rules

   (a) The government is considering a public project with cost C. Each citizen i has a valuation of the project v_i. (When modeling this, let the government be a 'player' with valuation of the project at $-C$). Under Clarke-Pivot rules, notice that if the mechanism decides to do the project, then $\sum_{\text{citizens}i} p_i < C$, meaning that the government will still have to subsidize the project via other means.

   (b) In a reverse auction, the bidder wants to *procure* an item from the bidder with lowest cost (note that the bidders here have negative valuations). For example, I might have some work I want done on my home and the bidders offer quotes. Clarke-Pivot rules will be a little weird here: explain why it's 'weird' and then take inspiration from the second-price auction to come up with a more 'reasonable' pricing.

   (c) In a bilateral trade problem a seller holds an item they value at s and a potential buyer values it at b. You'll notice something weird about the prices set by the Clarke-Pivot rules in the no-trade result - a more natural h_i function for each player is just the other player's value of the item. Recalculate the prices using this h_i: this will be more natural, but you'll still probably notice something odd about the trade result.

   (d) Consider a communication network, modeled as a directed graph $G = (V, E)$, where each link $e \in E$ is owned by a different player, and has a cost $c_e \geq 0$ if his link is used for carrying some message. We are asked to procure a communication path between two specified vertices $s, t \in V$. Under the Clarke-Pivot rules, exactly who is making payments?

2. **Optional** - finishing up from class In class, we started investigating combinatorial auctions for sub-additive (aka complement-free) valuation functions: $v(A \cup B) \leq v(A) + v(B)$. We first pointed out that if we could optimally determine the allocation ($\textsc{Opt} = (A_1, ..., A_n)$ where $A_i \cap A_j = \emptyset$ for all $i \neq j$ and $\cup_i A_i = M$) that maximizes social welfare ($\sum_i v_i(A_i)$) in polynomial time, then we would have that $\mathbf{P} = \mathbf{NP}$. So we defined the following algorithm:

   ```
   For each player i:
           Ask player i for v_i(M)
           For each item m ∈ M:
   ```

```
                Ask player i for v_i({m})

Make a complete weighted bipartite graph G = ([n] ∪ M, E)
        where w(i,m) = v_i({m}).
        **/ the left side is [n] and the right side is M **/

Compute the maximal weighted matching P
Let w(P) denote its weight

Let i* maximize v_i(M)
If w(P) > v_{i*}(M):
        Allocate according to P
        **/ give each item to the person they're matched to **/
```

Note that the only possibly options of this algorithm are $A = \{\text{all items go to } i \mid 1 \leq i \leq n\} \cup \{\text{each player gets just 1 item}\}$. In fact, the whole point of everything after `Make a complete weighted...` is simply to ensure that we choose the option in $a^* \in A$ that maximizes $\sum_i v_i(a)$. Since we are doing that, we can simply use Clarke-Pivot-Rule payments to ensure that the resulting mechanism is a VCG mechanism, meaning it's incentive compatible! Later this semester we'll actually determine the algorithm that finds that maximal weighted matching P (it's called the Hungarian algorithm if you would like to look ahead): its runtime is $O(|V|^3) = O((m+n)^3)$, which is polynomial in n and m. So now the only thing left to do is show that the maximum social value allocation in A is at least $\frac{1}{\sqrt{m}}$ the social value of OPT. To do so, I'm going to use the following notation:

- Let $\text{OPT} = (S_1^*, ..., S_n^*)$ be the optimal allocation.
- Let $\{T_1, ..., T_k\} = \{S_i^* \mid |S_i^*| < \sqrt{m}\}$
- Let $\{Q_1, ..., Q_\ell\} = \{S_i^* \mid |S_i^*| \geq \sqrt{m}\}$
- Because we've just reordered the bundles (T_1 is not necessarily S_1^*) the notation is going to get super abused going foward. Let $v(S)$ denote the value of S according to whoever is getting that set. For example, if $T_i = S_j^*$, then $v(T_i) = v_j(S_j^*)$. Obviously this is gross notation, but ah well.
- Lastly, define $v(\text{OPT}) = \sum_i v_i(S_i^*) = \sum_i^k v(T_i) + \sum_i^\ell v(Q_i)$.

The overall goal is to show that $w(P)$ or $w_{i^*}(M)$ is at least $\Omega(\frac{\text{OPT}}{\sqrt{m}})$.

(a) First, explain why $v(\text{OPT}) = \sum_i^k v(T_i) + \sum_i^\ell v(Q_i)$.

(b) We'll break it into two cases:
if $\sum_i^k v(T_i) \leq \sum_i^\ell v(Q_i)$:
   i. Explain why this means $\sum_i v(Q_i) \geq \frac{1}{2} v(\text{OPT})$.
   ii. Explain why $\ell \leq \sqrt{m}$ (proof by contradiction).
   iii. Let i^* maximize $v_i(Q_i)$.
      A. Show $v_{i^*} \geq v_{i^*}(Q_{i^*})$
      B. Show $v_{i^*} \geq \frac{1}{\sqrt{m}} \frac{v(\text{OPT})}{2}$

if $\sum_i^k v(T_i) > \sum_i^\ell v(Q_i)$:

i. Explain why this means $\sum_i v(T_i) \geq \frac{1}{2} v(\text{OPT})$.

ii. Let $g_i = \text{argmax}_{g \in T_i} v(\{g\})$. Use sub-additivity to show that $v(\{g_i\}) \geq \frac{v(T_i)}{|T_i|}$.

iii. Consider the partial matching P' on G given by $i \to g_i$.

A. Explain why $w(P) \geq w(P') = \sum_i v(\{g_i\})$.

B. Show that $w(P) \geq \frac{1}{\sqrt{m}} \frac{v(\text{OPT})}{2}$.

(c) Put it together - why are we guaranteed an $O(\sqrt{m})$ approximation?

3. **Optional** In class I mentioned that there is a 2-approximation algorithm for *submodular* valuation functions that is not incentive compatible. A submodular valuation function v satisfies $v(S \cup \{x\}) - v(S) \geq V(T \cup \{x\}) - v(T)$ for any $S \subseteq T$. This is kinda like saying that I get less value for x based on how much more I already have (S vs T). (3.(c) asks you to show that if v is submodular then v is subadditive, but it's not necessary for this problem). As a handy notation, let $v(x \mid S) = v(S \cup \{x\}) - v(S)$. Let M be the set of items to allocate and initialize $S_1 = S_2 = ... = S_n = \emptyset$. We'll then run the following recursive algorithm:

```
def Divvy(M):
    if M is empty:
        return

    Pick an item x ∈ M
    Ask each player j to report v_j(x | S_j)
    Add x to the bundle S_j that maximizes v_j(x | S_j)
    Divvy(M \ {x})
```

(a) Observe that this algorithm only requires polynomial number of operations.

(b) Why is this algorithm without any prices obviously not incentive compatible?

Let Q be some instance of this problem and let Q' be the instance on the remaining $m-1$ items after the first item x_1 is assigned: so the player j that got the first item has v_j replaced with v'_j where $v'_j(S) = v(S \mid \{x_1\} = v(S \cup \{x_1\}) - v(\{x_1\})$. All other valuations remain the same. Let $A(Q)$ denote the value of the allocation produced by the above algorithm and let $\text{OPT}(Q)$ denote the value of the optimal allocation. Let $p_1 = v_j(x_1)$.

(c) Show that $A(Q) = A(Q') + p_1$.

Let $S = (S_1, ..., S_n)$ be the optimal allocation for Q and let $x_1 \in S_k$. Let S' be the same allocation for Q' except where $S'_k = S_k \setminus \{x_1\}$. So all players except k have exactly the same allocation and all players except j have the same valuation function. For now, assume $j \neq k$.

(d) Use the submodularity of valuations to show that player k loses at most p_1 value between S and S'.

(e) Show that player j also loses at most p_1 value between S and S'.

(f) Conclude that $\text{OPT}(Q') \geq \text{OPT}(Q) - 2p_1$.

(g) Explain why even if $j = k$ we still get that $\text{OPT}(Q') \geq \text{OPT}(Q) - 2p_1$.

(h) Put it together: use induction to show that $\text{OPT}(Q) \leq 2A(Q)$. (This is very straightforward - just induct on the number of remaining items in M)

4. Some other entirely optional problems to think about:

   (a) Show that Clarke-Pivot rule gives that player i pays zero if and only if the chosen outcome maximizes the total reported valuation of all other players.

   (b) Give an example of an social choice function (the f part of mechanisms) for single item auctions that doesn't maximize reported social welfare, but show that this function can still be implemented as an incentive compatible mechanism with suitable payments.

   (c) Let v be a submodular valuation function. Show that $v(A \cup B) \leq v(A) + v(B)$. (Let $B = \{b_1, ..., b_n\}$, $B_i = \{b_1, ..., b_i\}$. Show that $v(A \cup B_i) - v(A \cup B_{i-1}) \leq v(B_i) - b(B_{i-1})$.)