CSC2410 - Assignment 3 solutions "sketch"

(1) Let $\alpha(G)$ be max size of an independent set, $\alpha'(G)$ be max size of a matching, $\beta(G)$ be min size of a vertex cover, $\omega(G)$ be max size of a clique, $\chi(G)$ be min size of a colouring, and $\theta(G)$ be a min size of a clique covering.

It is clear that $\chi(\overline{G}) \geq \omega(\overline{G})$. So we just need to prove that $\chi(\overline{G}) \leq \omega(\overline{G})$. First, for any graph, given a matching of size k, it is easy to construct a clique cover of size $n-k$ (take each edge in the matching as a clique of size 2, and the remaining vertices as cliques of size 1). So, we have $\alpha'(G) \leq n - \theta(G)$. Since G is a bipartite graph, by König theorem ($\alpha'(G) = \beta(G)$), we have $\beta(G) \leq n - \theta(G)$ which implies $\theta(G) \leq n - \beta(G)$. Since $n - \beta(G) = \alpha(G)$, we have $\theta(G) \leq \alpha(G)$. Taking the complement, we thus have $\chi(\overline{G}) \leq \omega(\overline{G})$, we are done.

(2) [Algorithm]:
(1) While there is a vertex v with degree at least $n^{\frac{k-2}{k-1}}$, colour **exactly** $n^{\frac{k-2}{k-1}}$ vertices (say this set S) in $N(v)$ using this algorithm recursively with a new set of colours, and then remove S from the graph.
(2) Colour the remaining graph using greedy algorithm with a new set of colours.

The resulting colouring is a proper colouring since we use a new set of colours in each iteration. We just need to analyse how many colours we have used. We shall prove that we need at most $k \times n^{\frac{k-2}{k-1}}$ colours. Let l be the number of iterations performed in step 1. Note that since each S is of size $n^{\frac{k-2}{k-1}}$, we have $l \leq n^{\frac{1}{k-1}}$. Also, each S can be coloured by at most $k-1$ colours (by induction, S can be coloured by $(k-1)|S|^{\frac{k-3}{k-2}}$ colours).

Total colours used in step 1 is at most:

$$\sum_{i=1}^{l}(k-1)(n^{\frac{k-2}{k-1}})^{\frac{k-3}{k-2}} \leq n^{\frac{1}{k-1}}(k-1)n^{\frac{k-3}{k-1}} = (k-1)n^{\frac{k-2}{k-1}}$$

Combining with $n^{\frac{k-2}{k-1}}$ colours used in the greedy algorithm. Total colours used in the algorithm is $kn^{\frac{k-2}{k-1}}$. Running time of the algorithm is linear (similar analysis, details omitted).

(3) Since adding edges to a graph would not decrease the chromatic number of a graph, we can assume, without loss of generality, that the given outerplanar graph G is 2-connected and every face is triangular (we keep adding edges until the above conditions are satisfied).

We first show that G has a vertex of degree 2. Consider the dual G^* of G, every cycle of G^* must pass through the vertex o corresponding to the outerface (otherwise there is an internal vertex). So $G^* - o$ is a forest. In fact, since G is 2-connected, $G^* - o$ is a tree. Arbitrarily take a leaf f of G^*, f must be a triangular face with two sides adjacent to the outerface (otherwise there is an internal vertex). So there is a degree 2 vertex v in f. Now, by induction, $G - v$ is 3-colourable and thus G is 3-colourable (since v is of degree 2).

Given any simple polygon, we can always triangulate it without introducing new vertex (whenever there are two adjacent lines with an angle less than 180, we add a line and the number of vertex decrease by 1). The resulting graph is an outerplanar graph and thus can be coloured by using at most 3 colours. There is a colour with at most $\lfloor n/3 \rfloor$ vertices, and we place the guards in those vertices with this colour. Every triangle is seen by at least one guard and thus every point is visible by some guard.

(4) If G is non-planar, by Kuratowski theorem, G has either K_5 or $K_{3,3}$ as a subdivision and thus G has either K_5 or $K_{3,3}$ as a minor (by contracting internal vertex one by one).

Since taking minor preserves planarity and K_5 and $K_{3,3}$ are not planar, if G is planar, G does not have K_5 or $K_{3,3}$ as a minor.

(5) Since G has no isolated vertex, $L(G)$ connected implies G connected. Also, $L(G)$ regular implies the degree sum of every pair of adjacent vertices the same. Consider an arbitrary edge $uv \in E(G)$, if $deg_G(u) = deg_G(v)$, then it forces every degree to be the same and thus G is regular. Otherwise, consider a BFS starting at u, then vertices of even distance must have degree equals to $deg_G(u)$ and vertices of odd distance must have degree equals to $deg_G(v)$. Therefore, no edge between vertices of odd distance and vertices of even distance (otherwise degree sum is not the same). So, we obtain a 2 colouring and G is a bipartite graph where vertices of the same partite set have the same degree.

(6) (a) Let C_1, C_2, C_3 be three of those nontrivial components, and y_1, y_2, y_3 be the neighbours of x in C_1, C_2, C_3, respectively. Consider a Hamiltonian cycle $\mathcal{C}$ starting at x and assume the Hamiltonian cycle visit C_1, C_2, C_3 in this order (consider the first time the cycle use vertex in each C_i). First, $xy_1 \notin \mathcal{C}$, otherwise $\mathcal{C}$ cannot leave C_1 (must go through x again) or cannot re-enter C_1. So $xy_1 \notin \mathcal{C}$, but then y_1y_2 must be in $\mathcal{C}$ (the only way from C_1 to C_2 without going through x). But then when $\mathcal{C}$ leaves C_2 it must pass through x again (or cannot re-enter C_2), and so the cycle is closed. But it never pass through C_3, so $\mathcal{C}$ is not a Hamiltonian cycle.

(b) The base case $n = 3, 4$ are easy to check. Let assume $n \geq 5$. Consider an arbitrary edge $e = xy$ in a tree T, and let T_x and T_y be the two components of $T - e$. Let T_x be a component with size at least 3 and $x'x \in E(T_x)$. By induction hypothesis, T_x has a Hamiltonian cycle C_x passes through $x'x$. If $T_y = \{y\}$, then $\mathcal{C} = C_x - x'x + xy + x'y$ is a Hamiltonian cycle of T^3 that passes through xy. If $T_y = \{y, y'\}$, then $\mathcal{C} = C_x - x'x + xy + yy' + x'y'$ is a Hamiltonian cycle of T^3 (since $d_T(x', y') = 3$) that passes through xy. Otherwise, let $yy' \in E(T_y)$. By induction hypothesis, T_y has a Hamiltonian cycle C_y passes through $y'y$. Then $\mathcal{C} = C_x + C_y - x'x - yy' + xy + x'y'$ is a Hamiltonian cycle of T^3 that passes through xy. Since xy is an arbitrary edge, we are done.

(7) (a) [Algorithm]: For each vertex v, place v in each side with probability $1/2$.

Let $Pr(e)$ be the probability that an edge is in the cut, and let $E(X)$ be the expected number of edges in the cut. Then, $Pr(e) = 1/2$ (probability that the two endpoints in different partite set). Then $E(X) = m \times Pr(e) = m/2$. Since the max-cut can have at most m edges, we obtain a 2-approximation algorithm with high probability (by repeating the above procedure, say, n times).

(b) [Algorithm]: Construct a random permutation and take the first half on one side and the second half on the other side.

Then $Pr(e) = (n/2)/(n-1)$ ($n/2$ positions on the other side, total number of remaining positions is $n-1$ since we already fix one). So $E(X) = (nm/2)/(n-1)$, and therefore a $(n-1)/(n/2)$-approximation.