Ma/CS 6a: Problem Set 3*

Due noon, Thursday, October 23

1. We run the BFS algorithm on a connected graph $G = (V, E)$ and a vertex $s \in V$, and then set $\alpha(G, s) = \sum_{v \in V} d[v]$ (where the $d[v]$ are the distance values that the algorithm determines).

(a) What is the minimum possible value that $\alpha(G, s)$ can have, and which graphs yield this value? What is the maximum possible value that $\alpha(G, s)$ can have, and which graphs yield this value? (both values should be functions of $|V|$). Explain your answers.

(b) Prove or disprove: Given a graph $G = (V, E)$, the value of $\alpha(G, v)$ is maximized for a vertex $v \in V$ that is of a minimum degree in G.

2. In class we saw that the graph edges that are not in a BFS tree are divided into two types: edges between vertices of the same level in the BFS tree and edges between consecutive levels. Then, to obtain the shortest paths graph, we added the edges of the latter type to the BFS tree.

(a) Consider the case of a directed graph. Characterize the types of edges that are not in the BFS tree, and explain which types should appear in the shortest paths graph.

(b) Let $G = (V, E)$ be a directed graph, and let $s, t \in V$. The *shortest paths graph from s to t* is the subgraph $G' = (V', E')$ that contains the vertices and edges of G that are part of some shortest path from s to t. Given G, s, t, explain how to efficiently find a shortest paths graph from s to t (hint: at some point the algorithm should reverse the directions of the edges of the graph).

3. Consider an undirected graph $G = (V, E)$, two vertices $v, u \in V$, and a subset $V' \subset V \setminus \{v, u\}$ such that every vertex of V' is connected to both v and u, and every vertex of $V \setminus (V' \cup \{v, u\})$ is connected neither to v nor to u (there is no restriction on whether v is connected to u or not). Let G' be the graph that is obtained by removing v, u from G (and the edges that are adjacent to them).

Prove or disprove:

(a) If G has an Eulerian cycle then G' has an Eulerian cycle.

(b) If G' has an Eulerian cycle then G has an Eulerian cycle.

4. Adam's pet alligator Biscuit likes to play dominoes. Unfortunately, Biscuit does not understand that the tiles can be rotated by 90° and 270° (he is OK with 180° rotations), and thus only places the tiles in one long row.[1] Describe an efficient algorithm that receives a pile of various domino tiles and checks whether Biscuit can place them all in one row.

5. (NO COLLABORATION) An undirected graph is said to be *almost bipartite* if it

*The awesome students who helped correcting this assignment: Ingrid Fiedler, Esha Wang, and Arpit Panda.

[1] As usual, each tile consists of two numbers between one and six. Two domino tiles may touch only if the two corresponding numbers are identical.

can be turned into a bipartite graph by removing at most one edge from it. Biscuit wrote the following algorithm for checking whether a connected undirected graph $G = (V, E)$ is almost bipartite:

- Arbitrarily choose a vertex $s \in V$, and then run the BFS algorithm from s.
- Go over every edge $(u, v) \in E$ and check whether $d[u] \neq d[v]$.
- If at most one edge failed this check, return "True". Otherwise return "False".

Either prove that Biscuit's algorithm is correct, or find a counterexample.