Assignment 5 Problem 1

Rather than solving the problem in graph G, we need to work on an extended graph G' of G.

To construct G', we do the following thing:

- For each vertex v in G, there are k corresponding vertices in G' namely $v_0, v_2, \dots, v_{k-1}$
- For each edge uv in G, there are k corresponding edges in G' namely $u_0v_1, u_1v_2, u_2v_3, \dots, u_{k-2}v_{k-1}, u_{k-1}v_0$
  (i.e. $u_iv_{i+1}\ \forall\ i \in [0, k-2], and\ u_{k-1}v_0$)

By doing this, we have another graph G' with kn vertices and km edges.

Note that constructing G' costs O(km+kn) time.

Then just doing an BFS on G' with source vertex s_0, then the level number of t_0 in the BFS tree is the shortest length of walk from s to t whose length is divisible by k in G.

To prove the correctness, here is the reason why constructing G' by the way that stated previously. For each vertex v_i, this means that the walk reaches v with walk length $l \equiv i\ (mod\ k)$. And each edge u_iv_{i+1} (for $i \in [0, k-2]$) means that at the end of a walk length l, we walk 1 length further, then the new walk is of length $l + 1 \equiv i + 1\ (mod\ k)$. And for edge $u_{k-1}v_0$, this means that at the end of a walk length l, where $l \equiv k - 1$, we walk 1 length further, then the new walk is of length $l + 1 \equiv k \equiv 0\ (mod\ k)$. Then by constructing the G', we have initially add some modulo information from G into G'.

The reason to let s_0 as the source vertex in BFS is because we want to use s as a source vertex in G, the length of the shortest walk from G to G is obviously 0. Because we want to find the shortest walk from s to t whose length is divisible by k, so as long as we reach t_0 in G', we have a walk that is divisible by k by the reason stated previously (the way we construct edges force the correctness). And since the level number of BFS tree for each vertex is minimized. Therefore, the level number of t_0 in the BFS tree is the shortest walk which is divisible by k.

The running time for BFS on a graph with kn vertices and km edges is O(kn+km).

So, the total running time for this algorithm is $O(kn + km)$.

Since we assume that $m \in \Omega(n)$, the runtime for the algorithm is $O(km)$