

Let $G = (V, E, w)$ be a directed and weighted graph with edge weights $w(e)$, which can be negative or positive or zero, for each edge $e \in E$. You may assume that G has no *negatively weighted* cycles. Additionally, each vertex is coloured either *red* or *blue*. Recall that a *walk* is a sequence of edges joining a sequence of vertices; there are no restrictions on the edges or vertices (i.e. they may repeat or not repeat).

Given an integer $k \geq 2$, describe an $O(kn(km + n))$ algorithm that returns a walk with the smallest weight, satisfying the following conditions:

- The walk starts and ends on a vertex coloured red, and
- The number of vertices coloured blue in the walk is divisible by k .

Note. For a directed and weighted graph $G = (V, E, w)$ where the edge weights can be negative, Bellman-Ford will compute the shortest distance from a vertex u to every vertex in time $O(|V| \cdot |E|)$. We will see how it works later but for now, you may use it as a black box.

Hint. Try constructing your edges in layers modulo k . You should have $km + 2n$ edges and $kn + 2$ vertices in your new graph.

Solution. Consider taking a copy of k graphs, layered $1, \dots, k$, keeping the original marked by introducing an auxiliary start and end red vertex that connects to all other red points in this original layer. This means that our new graph will have kn vertices for the k copies of the original and 2 for the auxiliary starting and ending points, yielding a total of $kn + 2$ vertices. We must also have k copies of the original edges, km , and, at worst $2n$. These new edges that connect from the auxiliary start and to the end point must have weight 0. (assuming all other vertices are red), yielding a total of $km + 2n$ edges. The connections from the auxiliary start and to the end point All connections to blue vertices must be connected to the corresponding blue vertex in the layer above rather than on the same layer.

The original layer must contain both the connections to the start and end layer, as the layer number represents the value of the amount of blue vertices being passed through in modulo k . Now, we must find our shortest path from the auxiliary starting point to the end point, which can be achieved by using the Bellman-Ford algorithm. To correspond this to the original graph, we simply imagine the projection from "above" the layers and follow that projected path. Or, more formally, the vertices that correspond to the same vertex in each layer is flattened back into the original graph. The ideal start and end-points work by identifying which red vertex the algorithm chooses at the start and the end after using the auxiliary endpoints.

The time complexity is given by $O(|V| \cdot |E|) = O((kn + 2)(km + 2n)) = O(kn(km + n))$.