HW 3

Due on **Wednesday, September 13, 2023 at 9pm**

CS 473: Algorithms, Fall 2023 Version: **1.1**

Submissions instructions: As in previous homework.

7 (100 PTS.) Cutting the grid.

You are given an $n \times m$ grid (i.e., rectangle). Each grid cell has a label which is its intended owner (i.e., a label is an integer number in $[\![nm]\!] = \{1, \ldots, nm\}$). (Say the labels are specified by table $L[n, m]$). Our purpose here is to cut the grid into pieces (i.e., rectangles) such that all the grid cells in a single piece have the same label. A grid (i.e., rectangle) can be cut only along one of its horizontal or vertical grid lines, which breaks it into two rectangles. The ***cost*** of cutting a rectangle along such a cut is the total area of the rectangle (i.e., the total number of grid cells in it). Describe an algorithm, as efficient as possible, that computes the minimum cost way of cutting the input grid into pieces, such that all the cells in each piece have the same label.

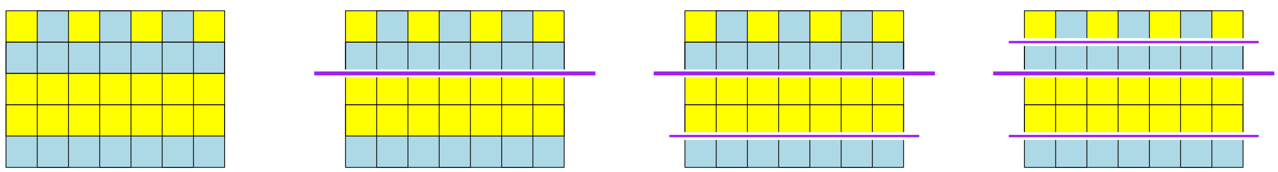

Figure 1: An initial input with three consecutive cuts illustrated. The prices of the cuts from left to right are 35, 21 and 14. (More cuts are required to get a valid solution.)

8 (100 PTS.) Synchronized motion planning II.

You are given a graph G with n vertices and m edges. You have two robots R_i, for $i = 1, 2$, where R_i must move along its given path π_i from its start vertex s_i to its end t_i. The two robots can move only forward along their respective paths (no going back to a vertex that was already used by the robot).

Specifically, a valid move for a robot is either to stay in its current location, or move to the next vertex on its path. In each round, each robot makes exactly one legal move. Once the two moves were chosen for both robots in a round, they both move simultaneously and instantaneously to their new locations. A sequence, specifying in each round a valid move for each robot, is a *schedule*.

Given a schedule of motion, its ***width*** is the maximum distance of the two robots from each other, in any time during the execution of this schedule. Describe an algorithm, as fast as possible, that computes the minimum width schedule for the two robots. What is the running time of your algorithm is function of n, m, N_1, N_2, where $N_i = |\pi_1|$ is the length (in number of edges) of π_i.

9 (100 PTS.) Cover by rectangles.

Let $p_1 = (x_1, y_1), \ldots, p_n = (x_n, y_n)$ be a sequence P of points in the plane, such that $x_i < x_{i+1}$, for all i. Let $R(i, j)$ denote the smallest axis-parallel rectangle having p_i and p_j as corners. A rectangle $R(i, j)$ is ***valid*** if it covers all the points in $p_i, \ldots, p_j$. Given a parameter k, describe an algorithm to compute the minimum area cover of all the points of P by k valid rectangles. What is the running time of your algorithm?

(Note: The k rectangles of the solution have to be interior disjoint.)