McGill University COMP360 Winter 2011 **Instructor: Phuong Nguyen**

Assignment 8

Due March 16 before the lecture

The work you submit must be your own. You may discuss problems with each others; however, you should prepare written solutions alone. Copying assignments is a serious academic offense, and will be dealt with accordingly.

Question 1 (10pt) You are given a color picture consisting of an $m \times n$ array A of pixels, where each pixel specifies a triple of red, green, and blue (RGB) intensities. Suppose that we wish to compress this picture slightly. Specifically, we wish to remove one pixel from each of the m rows, so that the whole picture become one pixel narrower. To avoid disturbing visual effects, however, we require that the pixels removed in two adjacent rows be in the same or adjacent columns; the pixels removed from a "seam" from the top row to the bottom row where successive pixels in the seam are adjacent vertically or diagonally.

(a) [**2pt**] Show that the number of such possible seams grows at least exponentially in m (the number of rows), assuming that $n > 1$.

(b) [**8pt**] Suppose now that along with each pixel $A[i, j]$, we have calculated a real valued disruption measure $d[i, j]$, indicating how disruptive it would be to remove pixel $A[i, j]$. Intuitively, the lower a pixel's disruption measure, the more similar the pixel is to its neighbors. Suppose further that we define the disruption measure of a seam to be the sum of the disruption measures of its pixels.

Give a dynamic programming algorithm to find a seam with the lowest disruption measure. The space used by your algorithm must be $\mathcal{O}(mn)$. (For space requirement, assuming that the total disruption measure of any seam requires a constant amount of space.) The running time of your algorithm must be a polynomial in mn; calculate it.

Question 2 (10pt) Consider a directed graph G where each edge e is labeled with a label called $\ell(e)$ that comes from a set L of labels. We define the label of a directed path to be the concatenation of the labels of the edges on that path.

(a) [**5pt**] Describe a polytime algorithm that, given an edge-labeled directed graph G with a distinguished vertex s and a sequence of labels $(\ell_1, \ell_2, \ldots, \ell_k)$, returns a path in G that begins with s and has the sequence $(\ell_1, \ell_2, \ldots, \ell_k)$ as its label, if any such path exists. Otherwise the algorithm must return NO-SUCH-PATH. Analyze the running time of your algorithm.

(b) [**5pt**] Now suppose that every edge e in the graph is associated with a positive weight $w(e)$. The weight of a path is defined to be the sum of the weights of its edges. Extend your algorithm in (a) so that it returns a path of maximum weight. Your algorithm must still run in time polynomial in the size of the input.

Here we assume that each label ℓ in L, each weight $w(e)$, as well as the total weight of any path of length n ($n = |V|$ is the total number of vertices in G) occupy a constant amount of space, and that adding/comparing two weights can be done in constant time.