CS130 #

DATA STRUCTURES

Work Sheet 6

The convention is to denote number of vertices in a graph by n and number of edges by m.

- Show an efficient procedure to count the number of connected components of a graph. Analyze for time and space.
- Given an directed acyclic graph, suggest an efficent way to find shortest path between a pair of nodes. Edges have been given positive lengths(i.e. not necessarily 1). It can obviously be done using Dijkstra. Can you suggest an $O(m)$ time algorithm?
- What is the running time of heapsort on an array of A of length n that is already sorted in increasing order? What about decreasing order?
- A simple minded Dijkstra takes $O(n^2)$ time. Can you suggest use of some data structure that would give a running time of $O(m \log n)$
- Suppose there are three types of elements in an array. Red, yellow and blue. You have to sort the elements such that all reds come first, followed by yellow andthen blues. Give a linear time procedure *in place. You can not use more than* $O(1)$ *additional space.*