**Assignment 5**

CS 362

Baker, Alex

# Problem 1

Using the contraction algorithm for min cut can be found using Krager’s algorithm. This algorithm uses contractions to combine vertices until only two are left which provide the cut of the graph. Choosing the vertices to combine during the contraction are chosen at random. If the graph is given as an adjacency list, the the first vertex is chosen from the list of all vertices, and the vertex to contract with is chosen at random from its list of adjacent vertices. This grantees that each contraction will be between two vertices that share an edge, therefore they do not need to be repacked.

# Problem 2

# Problem 3

In order for there to be a product of 1, there would have to be a number x, and its reciprocal 1/x. I hash function can be used to iterate through the list a place all of the values into a hash table. For every element, its reciprocal is also calculated and tested to see if that value is in the hash table. If the reciprocal is already in the hash table, then there exists a pair of number whose product is 1. This can be done in O(n) time.

# Problem 4

# Problem 5

Because this only needs to find one local minimum, this can be done in a binary search manner. Start by picking the minimum card and inspecting the two cards on either side of it. One of these cards must be less than or equal to the middle card. If the middle card is a local minimum, then return the middle card, if it is not, the recursively search half of the remaining cards on the side with the smaller card. Because this uses a BST, it can be done in O(log n) time.

# Problem 6

The shortest path can be found by using linear programming have the condition to minimize the sum of all of the edges in a set, subject to the constraint that all edges in the set must be a path in the graph. This LP program will find the shortest path with the given constraints.

# Problem 7

# Problem 8

In order to find a minimum vertex cover of a bipartite graph, a maximum matching could optionally be found. This comes from Konig’s theory which states that the number of edges in a maximum matching is the same as the minimum vertex cover. This can be done in O(n) time. A maximum matching is the maximum set of edges which do not share an endpoint. The vertex cover is a subset of the vertices from the maximum matching problem which do not share an endpoint. The Hopcroft-Karp algorithm can be used to find a maximum matching on a bi partite graph which runs in

# Problem 9

A LP can be reduced to this problem by using slack variables. These slack variables represent the surplus of addition or subtraction and can reduce inequalities to equalities.