CS 530 - Fall 2018
Homework 5

Due: None. These are practice problems, mostly about approximation. they will not be graded, but can serve as study problem for the final exam.

Reading :

Look through the first half of chapter 34 on NP. If you find topics that are new or unclear then do some reading in this chapter or elsewhere on that topic.
Read Chapter 35 sections 1,2,3 and 5.

Problems:

1. The weighted set cover problem (WSC) is similar to the unweighted case. However here the elements x of the set X we are covering all have positive rational number weights w and the cost of a S set S is the sum of the weights of its elements. The WSC problem asks us to find a set cover C of smallest possible weight. The wight of C is the sum of the weights of the sets we put into C.

In class we considered a greedy approximation algorithm for the unweighted set cover. We can extend this greedy algorithm to the weighted case in a simple way. Namely we order the sets which might go into our cover from smallest weight to largest weight. We then consider these sets in that order and put them into our cover if doing so covers some element which has not already been covered. We stop when all the elements of X are covered. ¡p¿

(i). Give an example of a WSC problem where the above algorithm results in an answer which is more than twice the optimal answer. Show that your example works as described.

¡p¿ (ii). Is there some fixed constant A where the above algorithm always gives an A-approximation to WSC. Why or why not ? (Clearly A would have to be greater than 2.)

2. State the decision version of the weighted set cover problem and show that this problem is in NP.

3. The next fit (NF) bin packing algorithm is one which runs extremely efficiently but is not very close to optimal. The algorithm is simple.

We are given an unordered list of objects all of size ≤ 1 to pack. We want to pack the objects into bins of size 1 as usual, using as few total bins as possible.

We start with one open empty bin and never have more than one open bin at any time. We pack the objects in the order given until all objects are packed.

To pack the next object O, we first see if it fits in the open bin.

If so, put it in.

If not, close the open bin, open an empty bin and put O in the newly open bin.

i. Give an example of a bin packing instance where the NF algorithm does worse than 1.3 times the optimal solution.
(That is, give an instance I for bin packing where NF(I)> 1.3 OPT(I).)

ii. Show that the NF algorithm above gives a 2-approximation for bin packing.

4. One of the following three problems is known to be in NP.
Say which one is in NP and prove it.

i. Number of perfect matchings problem: The input is a bipartite graph G with the same number of vertices on each side of the graph. The output is the number of perfect matchings in G.

ii. The graph sub-isomorphism problem: Given two graphs G and H, does G have a subgraph which is isomorphic to H

iii. The clique problem: Given a graph G, what is the largest clique (that is complete subgraph) contained in G ?

5. The graph vertex coloring problem (GVC) is: Given a graph, find the fewest number of colors that can be used to color its vertices so that no two connected vertices have the same color.

i. Give an efficient algorithm to decide if a graph can be colored with 2 colors.

ii. Reducing the GVC problem to decision GVC: State the decision version of the graph vertex coloring problem and show that if this decision problem has has an efficient (polynomial time) solution than so does the GVC problem.