Due 23:59 Nov 22 (Tuesday). There are 100 points in this assignment.
Submit your answers (**must be typed**) in pdf file to CourSys `https://coursys.sfu.ca/2022fa-cmpt-705-x1/`. Submissions received after 23:59 of Nov 22 will get penalty of reducing points: 20 and 50 points deductions for submissions received at $[00:00, 00:10]$ and $(00:10, 00:30]$ of Nov 22, respectively; no points will be given to submissions after $00:30$ of Nov 22.

1. (Chapter 8 Problem 4 of the text book) 20 points

   A system consists of n processors and m resources; each processor requests a subset of resources. An assignment of resources to processors is to allocate each resource to at most one processor. A processor is active in an assignment if all resources it requests are allocated to it, otherwise blocked. The resource reservation problem is that given n processors, m resources, a set of requested resources for each processor, and an integer $k > 0$, is it possible to allocate resources to processors so that at least k processors are active. For each of the following resource reservation problems, give a polynomial time algorithm or prove it is NP-complete.

   (a) The general resource reservation problem as above.

   (b) The special case of the problem with $k = 2$.

   (c) The special case that there are two types of resouces (e.g., hard disks and GPUs) and each processor requests at most one resource of each type.

   (d) The special case that each resource is requested by at most two processors.

   Answer

   ---

2. (Chapter 8 Problem 6 of the text book) 15 points

   A CNF F on Boolean variables $x_1, .., x_n$ is monotone if each literal in each clause of F is the form of x_i (the negation $\bar{x}_i$ does not appear in any clause for any i). Given a monotone CNF F, the monotone satisfiability with few true variable problem asks that whether there is a satisfiable assignment for F in which at most k of $x_1, .., x_n$ are set to 1? Prove this problem is NP-complete.

   Answer

   ---

3. (Chapter 8 Problem 21 of the text book) 15 points

   The fully compatible configuration (FCC) problem is defined as follows: An instance of FCC consists of disjoint sets of $A_1, .., A_k$, each A_i is a set of options, and a set P of incompatible pairs (x, y), where $x \in A_i$ and $y \in A_j$ with some $1 \leq i \neq j \leq k$. The FCC problem is to decide whether there is fully configuration which is a selection of one element from each A_i $(1 \leq i \leq k)$ so that no pair of selected elements is in P. Prove the problem is NP-complete.

   Answer

   ---

4. (Chapter 11 Problem 1 of the text book) 15 points

   There are n containers $1, .., n$ of weights $w_1, .., w_n$ and some trucks, each truck can hold at most K units of weight (w_i and K are positive integers, $w_i \leq K$). Multiple containers can be put on a truck subject to the weight restriction K. The problem is to use a minimum number of trucks to carry all containers. A greedy algorithm for this problem as follows: start with an empty truck and put containers $1, .., j$ on the truck to have this truck loaded (i.e., $\sum_{1 \leq i \leq j} w_i \leq K$ and $\sum_{1 \leq i \leq j+1} w_i > K$); then put containers $j+1, j+2, ...$ on a new empty truck to have this truck loaded; continue this process until all containers are carried.

   (a) Give an example of a set of containers and a value K to show that the greedy algorithm above does not give an optimal solution.

   (b) Prove the greedy algorithm is a 2-approximation algorithm.

Answer

---

5. (Chapter 11 Problem 3 of the text book) 15 points

Given a set of positive integers $A = \{a_1, .., a_n\}$, positive integer B and a subset $S \subseteq A$, $t(S) = \sum_{a_i \in S} a_i$ is called the total sum of S. A subset S is called a feasible set if $t(S) \leq B$. The maximum feasible set problem is find a feasible set S with $t(S)$ maximized.

(a) A simple greedy algorithm works as follows:

Initially $S = \emptyset$; $t(S) = 0$;

for $i = 1, 2, .., n$ **do**

  if $t(S) + a_i \leq B$ **then** $S = S \cup \{a_i\}$ and $t(S) = t(S) + a_i$;

endfor

Give an instance that the algorithm returns a feasible set S with $t(S) < (1/2)t^*$, where t^* is the maximum total sum of a feasible set.

(b) Give an $O(n \log n)$ time (1/2)-approximate algorithm for the problem.

Answer

---

6. (Chapter 11 Problem 5 of the text book) 20 points

   (a) Consider a load balancing problem instance of 10 machines and n jobs $S = \{1, .., n\}$ with $1 \leq t_i \leq 50$ for every i and $\sum_{i=1}^{n} t_i \geq 3000$. Prove that the greedy algorithm discussed in class finds a solution of makespan T with $T \leq (1.2)(\sum_{1 \leq i \leq n} t_i)/10$ for this instance.

   (b) Implement the greedy algorithm to find a solution for the load balancing problem instance in (a) and compare the solution with the lower bound $(\sum_{1 \leq i \leq n} t_i)/10$ on the makespan of the instance (each t_i can be generated randomly). Submit the results of the greedy algorithm and the lower bound for some instances of about $30 \sim 50$ jobs.

Answer
