COT5405/CIS4930: ANALYSIS OF ALGORITHMS

Exam III

Date: April 18, 2017, Tuesday
Time: 8:20pm – 10:10pm (110 minutes)
Professor: Alper Üngör (Office CSE 534)

This is a closed book exam. No collaborations are allowed. Your solutions should be concise, but complete, and handwritten clearly. Use only the space provided in this booklet, including the even numbered pages. Feel free to refer to algorithms, definitions and concepts discussed in class rather than describing them in detail unless of course you are asked for it specifically.

(: GOOD LUCK and HAVE A GREAT SUMMER :)

First Name:

Last Name:

UF ID:

	Credit	Max
Problem 1		20
Problem 2		30
Problem 3		30
Problem 4		30
TOTAL		110

1. [20 points] PICK ONE *No justification is needed for this problem.*

   (a) Which one of the following is incorrect for Two Dimensional Range Trees?

      i. Counting range queries take $O(\log^2(n))$ time.

      ii. Reporting range queries take $O(\log^2(n) + k)$ time.

      iii. Construction time is $\Theta(n \log n)$.

      iv. Space complexity is $\Theta(n^2 \log n)$.

   (b) CLIQUE-DEGREE-4 problem: Given a graph $G = (V, E)$ where all vertices have degree at most 4, and an integer k, determine whether V has a subset S of size at least k that forms a clique in G. Which of the following statement(s) is/are correct?
   (a) CLIQUE-DEGREE-4 is NP-Hard; (b) CLIQUE-DEGREE-4 is in P; (c) Approximating CLIQUE-DEGREE-4 is NP-Hard; (d) There exists a polynomial time reduction from CLIQUE-DEGREE-4 to VERTEXCOVER problem.

      i. (a) and (d)

      ii. (b) and (d)

      iii. (a) and (c)

      iv. (b) and (c)

      v. All of them

   (c) MAJORITY-CHECK problem: Given a set S of n real numbers, determine whether more than half the numbers in S are exactly the same. Assuming two numbers can be compared in constant time, which of the following statement(s) is/are correct?
   (a) MAJORITY-CHECK is in NP; (b) There exists no comparison-based algorithm solving MAJORITYCHECK in $\Theta(n)$ time; (c) MAJORITY-CHECK is proven to be NP-hard via a reduction from SUBSETSUM problem.

      i. only (a)

      ii. (a) and (b)

      iii. (a) and (c)

      iv. All of them

      v. None of them

   (d) MAX-SAT: Given a CNF (Conjunctive Normal Form) Boolean formula and a positive integer k, does there exist a truth assignment that satisfies at least k clauses.
   Which one of the following is correct about this problem?

      i. This problem can be solved in polynomial time since k is a small number.

      ii. This problem is NP-hard since it is a special case of the SAT problem.

      iii. This problem is NP-hard since it is a generalization of the SAT problem.

      iv. This problem can be shown to be NP-hard by giving a reduction from MAX-SAT to CLIQUE problem.

      v. None of the above

2. [30 points] GEOMETRIC ALGORITHMS - CO-CIRCULARITY CHECK
Following procedures each with planar input points and constant running time are provided: $\underline{\mathrm{CCW}(p, q, r)}$ decides if r is LEFTOF, ON, or RIGHTOF the oriented line going through p, q. $\underline{\mathrm{InCircle}(p, q, r, s)}$ decides whether s is INSIDE, ON, or OUTSIDE the unique circle going through p, q, r.

(a) **Draw** five points a, b, c, d, e such that the radii of the circles going through triples of points (a, b, c), (a, b, d), and (a, b, e) are all the same, a, b, c, d are co-circular, and yet a, b, c, e are not co-circular, i.e., $\mathrm{circumradius}(a, b, c) = \mathrm{circumradius}(a, b, d) = \mathrm{circumradius}(a, b, e)$, InCircle $(a, b, c, d) =$ ON, and InCircle $(a, b, c, e) \neq$ ON.

(b) Given n points in the plane (no three of which are co-linear) we want to determine whether any four of them are co-circular. Naive algorithm would take $O(n^4)$ time. **Design** and **analyze** an algorithm that solves this problem in $O(n^3 \log n)$ time.
(*Hints: Sorting takes* $O(n \log n)$ *time. Recall the co-linearity checking algorithm.*)

3. [30 points] NP-COMPLETENESS - STRONGLY INDEPENDENT SETS

Consider a simple graph $G = (V, E)$. A subset $S_1 \subseteq V$ is called an *independent set*, if no two vertices in S_1 have an edge (path of length one) between them. A subset $S_2 \subseteq V$ is called a *strongly independent set*, if no two vertices in S_2 have a path of length one or two between them, i.e., $\forall\ u, v \in S_2$, we have $(u, v) \notin E$, and $\nexists\ w \in V$ such that $(u, w) \in E$ and $(w, v) \in E$.

INDEPENDENTSET PROBLEM: Given a simple graph G and an integer k, determine if G contains an independent set of size at least k.

STRONGLYINDEPENDENTSET PROBLEM: Given a simple graph G and an integer k, determine if G contains a strongly independent set of size at least k.

(a) **Draw** a connected graph that has an independent set of size 4. **Draw** another connected graph that has a strongly independent set of size 4.

(b) **Prove** that STRONGLYINDEPENDENTSET PROBLEM is NP-Complete. (*Hints:* INDEPENDENTSET PROBLEM *is NP-hard. For your reduction, consider additional vertices positioned strategically and forming a clique among themselves.*)

4. [30 points] APPROXIMATION - PACK YOUR BOOKS, YOU GOT A JOB AT GOOGLE
Imagine that you just graduated UF and got a job at Google. For your move to California, you need to pack all your books to boxes each of which has capacity a real number between 1 and 2. For simplicity, assume that each book has weight a real number between 0 and 1. You want to minimize the number of boxes that you use. More formally:

PACKBOOKS PROBLEM: Given n books with weights $w_1, \ldots, w_n$ where $\forall$ i, $0 < w_i < 1$, m boxes with capacities $c_1, \ldots, c_m$ where $\forall$ j, $1 < c_j < 2$, and an integer k, is it possible to pack all books in at most k boxes such that total weight in each box is within its capacity.

(a) **Prove** that PACKBOOKS PROBLEM is NP-hard. (*Hints: Consider a special case of this problem that you are familiar with. No reduction necessary.*)

(b) **Design** and **analyze** an approximation algorithm for the minimization version of the PACKBOOKS PROBLEM with an approximation ratio at most 4.