COMP 360 - Fall 2015 - Sample Final Exam

There are in total 105 points, but your grade will be considered out of 100.

1. (10 points) Prove that the following problem belongs to P: Given a graph G, we want to know whether G has an independent set of size 100.

   Solution: Let n be the number of vertices of G. For every possible way of choosing 100 vertices, we check whether they form an independent set. Note that there are $\binom{n}{100} \leq n^{100}$ possible ways of choosing 100 vertices out of n vertices, and it takes on $O(1)$ to see whether they form an independent set. Thus the running time of this algorithm is $O(n^{100})$.

2. (10 Points) Prove that the following problem belongs to PSPACE: Given a graph G and an integer k, we want to know whether the number of independent sets in G is equal to k.

   Solution: Let n be the number of vertices of G. One can generate all the possible 2^n subsets of the vertices, one by one (reusing the memory). Each such set takes n bits of space. We will also have a variable I that is equal to the number of sets that form an independent set. Every time that a new subset is generated, we check whether it is an independent set, and then update the variable I accordingly. Note that I takes only $\log 2^n = n$ bits of memory. Hence in total the required space is going to be $O(n)$.

3. (10 Points) Show that the following problem is NP-complete:

   - Input: An undirected graph G and an edge e.
   - Question: Does G have a Hamiltonian cycle that passes through the edge e.

   Solution: We reduce the Hamiltonian cycle problem to this problem. For an input G for Hamiltonian cycle problem, we run the following oracle algorithm:

   - For every edge e in G.
   -         Run the oracle on $\langle G, e \rangle$ and if the output is YES, output YES and terminate.
   - EndFor
   - If not terminated yet, output NO.

4. (10 Points) Consider the following optimization problem:

   - Input: A 4CNF ϕ where the variables appearing in each clause are distinct.
   - Question: Find a truth assignment that maximizes the number of clauses that receive at least one true and one false term.

   Prove that a random truth assignment (i.e. every variable is independently set to either T or F with equal probability) provides a $\frac{7}{8}$-factor approximation algorithm for this problem. In other words, the expected value of the output is at least $\frac{7}{8}$ of the optimal solution.

Solution: Let m be the number of clauses. To every clause we assign an indicator random variable. That is $X_i = 1$ if the i-th clause receives at least one true and one false term. Then the number of clauses that receive at least one true and one false term is equal to $X = \sum_{i=1}^{m} X_i$. Then by linearity of expectation, we have

$$\mathbb{E}[X] = \mathbb{E}[X_1] + \ldots + \mathbb{E}[X_m].$$

Note further that $\Pr[X_i = 1] = \frac{14}{16} = \frac{7}{8}$ as out of the 16 possible assignments to the four variables in the clause, only 2 of them fail to create at least one true and one false term in the clause. Hence $\mathbb{E}[X_i] = \Pr[X_i = 0] \times 0 + \Pr[X_i = 1] \times 1 = \frac{7}{8}$. Consequently

$$\mathbb{E}[X] = \frac{7}{8}m \geq \frac{7}{8}\mathrm{OPT},$$

as optimal solution is obviously cannot be larger than m.

5. (10 points) Prove that the following algorithm is a 2-factor approximation algorithm for the minimum vertex cover problem:

   - While there is still an edge e left in G:
   - Delete all the two endpoints of e from G
   - EndWhile
   - Output the set of the deleted vertices

   Solution: Note that at every step the above algorithm finds a new edge e and deletes both its endpoints. Let M be the set of all such edges found by the algorithm. Note that the edges in M are completely disjoint (they do not share any vertices) as every time that the algorithm finds such an edge it deletes both its endpoints. The optimal solution must pick at least one vertex from each one of these edges, and thus the size of the optimal solution is at least $|M|$ (i.e. the number of the edges in M). On the other hand the algorithm deleted at most $2|M|$ vertices. Hence the output of the algorithm is at most $2 \times \mathrm{OPT}$.

6. (10 points) Prove that the following algorithm is a $\frac{1}{2}$-factor approximation algorithm for the MAX-SAT problem: Given a CNF ϕ on n variables $x_1, \ldots, x_n$:

   - For $i = 1, \ldots, n$ do
   - IF x_i appears in more clauses than $\overline{x_i}$ THEN
   - Set $x_i = T$
   - Else
   - Set $x_i = F$
   - Remove all True clauses from ϕ and remove x_i and $\overline{x_i}$ from all the other clauses
   - EndFor

   Solution: Let t_i be the number of clauses that become TRUE when we set the value of x_i, and similarly let f_i be the number of clauses that become FALSE (This means that at this point all the terms in the clause are already set to FALSE). Obviously the way the algorithm works guarantees $t_i \geq f_i$.

   On the other hand note that after the algorithm terminates, $\sum t_i$ is the total number of TRUE clauses in the formula and $\sum f_i$ is the total number of false clauses in the formula, and from the previous discussion we know that $\sum t_i \geq \sum f_i$. Thus the algorithm satisfies at least half of the clauses which is at least $\frac{1}{2}\mathrm{OPT}$.

7. (15 Points) Let G be a 4-regular graph on n vertices (4-regular means that every vertex is adjacent to 4 edges). We want to color the *edges* of G with two colors Red and Blue such that the number of vertices that are adjacent to exactly two Red and two Blue edges is maximized. If we color the edges at random, then what is the expected number of vertices that satisfy the above condition?

   Solution: For every vertex i, define X_i to be the following random variable. We have $X_i = 1$ if the ith vertex is adjacent to exactly two Red and two Blue edges, and $X_i = 0$ otherwise. Then the number of the vertices that satisfy the above condition is given by $X = X_1 + \ldots + X_n$, and

   $$\mathbb{E}[X] = \mathbb{E}[X_1] + \ldots + \mathbb{E}[X_n].$$

   Now note that

   $$\mathbb{E}[X_i] = \frac{\binom{4}{2}}{16} = \frac{3}{8},$$

   as there are $\binom{4}{2}$ choices for choosing the two blue edges incident to the i-th vertex. Thus

   $$\mathbb{E}[X] = \frac{3}{8}n$$

8. Consider a graph $G = (V, E)$. The chromatic number of G is the minimum number of colors required to color the vertices of G properly. Let $\mathcal{I}$ be the set of all independent sets in G (Note that every element in $\mathcal{I}$ is a set).

   (a) (10 Points) Prove that the solution to the following linear program provides a lower-bound for the chromatic number of G.

   $$\begin{array}{lll} \max & \sum_{I\in\mathcal{I}} x_I & \\ \text{s.t.} & \sum_{I:v\in I} x_I \leq 1 & \forall v \in V \\ & x_I \geq 0 & \forall I \in \mathcal{I} \end{array}$$

   Solution: Consider an optimal coloring of the vertices of G with k-colors where k is the chromatic number of G. Note that every color-class forms an independent set as vertices of the same color cannot be adjacent. Let $I_1, \ldots, I_k$ be the independent sets corresponding to the color-classes. Consider the following solution to the linear program: set $x_{I_1} = \ldots = x_{I_k} = 1$ and $x_I = 0$ for every other independent set I. This is a feasible solution as every vertex v belongs to exactly one of $I_1, \ldots, I_k$, and hence $\sum_{I:v\in I} x_I = 1$. Moreover the objective value of the linear program for this feasible solution is k. Hence the optimal value is at most k.

   (b) (10 Points) Write the dual of the above linear program.

   Solution:

   $$\begin{array}{lll} \min & \sum_{v\in V} y_v & \\ \text{s.t.} & \sum_{v\in I} y_v \geq 1 & \forall I \in \mathcal{I} \\ & y_v \geq 0 & \forall v \in V \end{array}$$

   (c) (10 Points) Prove that every clique in G provides a solution to the dual linear program.

   Solution: Let S be the set of the vertices of a clique in G. Set $y_v = 1$ if $v \in S$ and $y_v = 0$ if $v \notin S$. Since S forms a clique, no independent set contains more than one vertex from S, and thus $\sum_{v\in I} y_v \geq 1$ for every $I \in \mathcal{I}$. On the other hand the objective value of the linear program for this feasible solution is equal to $|S|$.