Final Exam, Advanced Algorithms 2019-2020

- You are only allowed to have a handwritten A4 page written on both sides.
- Communication, calculators, cell phones, computers, etc... are not allowed.
- Your explanations should be clear enough and in sufficient detail that a fellow student can understand them. In particular, do not only give pseudo-code without explanations. A good guideline is that a description of an algorithm should be such that a fellow student can easily implement the algorithm following the description.
- **You are allowed to refer to material covered in the lecture notes** including theorems without reproving them.
- **Problems are not necessarily ordered by difficulty.**
- **Do not touch until the start of the exam.**

Good luck!

Name: ______________________ **N° Sciper:** ______________

Problem 1 / 24 points	Problem 2 / 14 points	Problem 3 / 12 points	Problem 4 / 17 points	Problem 5 / 19 points	Problem 6 / 14 points

Total / 100

1 *(24 pts)* **Linear programming.**

1a *(12 pts)* **Duality.** Consider the following linear program:

$$\begin{aligned} \textbf{Minimize} \quad & x_1 + 8x_2 + 20x_3 \\ \textbf{Subject to} \quad & x_1 + 7x_2 + 2x_3 \geq 9 \\ & 3x_1 + x_2 + 4x_3 \geq 1 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

Write down its dual and the complementarity slackness conditions.

1b *(12 pts)* **Simplex method.** Suppose we use the Simplex method to solve the following linear program:

$$\begin{aligned}
\textbf{Maximize}\quad & 6x_1 - 4x_2 - 9x_3 \\
\textbf{Subject to}\quad & x_1 - 2x_3 + s_1 = 5 \\
& 2x_1 + x_2 + 4x_3 + s_2 = 35 \\
& 4x_3 - 3x_2 + s_3 = 12 \\
& x_1,\ x_2,\ x_3,\ s_1,\ s_2,\ s_3 \geq 0
\end{aligned}$$

At the current step, we have the following Simplex tableau:

$$\begin{aligned}
x_1 &= 5 + 2x_3 - s_1 \\
s_2 &= 25 - x_2 - 8x_3 + 2s_1 \\
s_3 &= 12 + 3x_2 - 4x_3
\end{aligned}$$

$$z = 30 - 4x_2 + 3x_3 - 6s_1$$

Write the tableau obtained by executing one iteration (pivot) of the Simplex method starting from the above tableau.

2 (14 pts) **Set Balancing.** You are given a collection of sets $S_1, S_2, \ldots, S_m$ in a universe U of size n with $|S_i| \le t$ for all $i = 1, \ldots, m$. A function $\chi : U \to \{-1, +1\}$ is called a q-balanced coloring if for every $i = 1, \ldots, m$ one has

$$\left| \sum_{e \in S_i} \chi(e) \right| \le q.$$

Give an efficient randomized algorithm for finding a q-balanced coloring with $q = O(\sqrt{t \log m})$. Your algorithm must succeed with probability at least $9/10$.

In this problem you should (a) design an efficient algorithm and (b) prove its correctness.

3 *(12 pts)* **Alice, Bob and Charlie.** Suppose that Alice and Bob have two documents d_A and d_B respectively, and Charlie wants to learn about the difference between them. We represent each document by its word frequency vector as follows. We assume that words in d_A and d_B come from some dictionary of size n, and let $x \in \mathbb{R}^n$ be a vector such that for every word $i \in [n]$[1] the entry x_i equals the number of times the i-th word in the dictionary occurs in d_A. Similarly, let $y \in \mathbb{R}^n$ be a vector such that for every word $i \in [n]$ the entry y_i denotes the number of times the i-th word in the dictionary occurs in d_B. We assume that the number of words in each document is bounded by a polynomial in n.

In the subproblems below you will design efficient communication protocols for Alice and Bob that let Charlie learn about the difference of d_A and d_B. All protocols must succeed with probability at least 9/10. For all subproblems below you may assume that Alice, Bob and Charlie have a source of shared random bits.

3a (5 pts) Suppose that for an integer parameter k there exist at most k words that are present in d_A but not in d_B, and at most k words that are present in d_B but not in d_A. Show that Alice and Bob can each send a message of $O(k \log^2 n)$ bits to Charlie, from which Charlie can recover the words that are present in d_A but not in d_B.

3b (7 pts) Suppose that there exists $i^* \in [n]$ such that for all $i \in [n] \setminus \{i^*\}$ one has $|x_i - y_i| \leq 2$, and for i^* one has $|x_{i^*} - y_{i^*}| \geq n^{1/2}$. Show that Alice and Bob can each send a $O(\log^2 n)$-bit message to Charlie, from which Charlie can recover the identity of the special word i^*.

(Hint: recall one of the streaming algorithms covered in class, and note that it is a linear sketch that can be applied to arbitrary vectors of length n in with integer entries bounded by a polynomial in n.)

(Recall that you are allowed to refer to material covered in the lecture notes. In this problem you must explain how your protocol works and why it is correct with the required probability.)

Note: you do not need to do detailed calculations in this problem.

[1] We let $[n] := \{1, 2, \ldots, n\}$.

(This page is intentionally left blank.)

4 *(17 pts)* **Spectral graph theory.** Let $G = (V, E)$ be a d-regular undirected graph, and let $M = \frac{1}{d}A$ denote its normalized adjacency matrix. Let $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{n-1} \geq \lambda_n$ denote the eigenvalues of M.

4a *(12 pts)* Prove that if $\lambda_{n-1} = -1$, then G is disconnected.

4b *(5 pts; do not start this problem until you are done with all others)* Prove that if $\lambda_n > -1/2$, then the graph G is not tripartite (recall that a graph is tripartite if its vertex set can be partitioned into three disjoint sets such that all of its edges connect vertices in different components).

(This page is intentionally left blank.)

5 *(19 pts)* **Online ad allocation.** Alice and Bob started companies selling hand sanitizer, and are now advertising their products online to potential customers $c_1, c_2, \ldots, c_n$, where $c_i \in \mathcal{C}$ for all $i = 1, \ldots, n$. When a customer c_i arrives, they can be shown advertisement for either Alice's or Bob's hand sanitizer – we say that the customer is *allocated* to either Alice or Bob in that case. If S_1 and S_2 are the sets of customers allocated to Alice and Bob respectively at the end of the sequence, Alice will pay $v_1(S_1)$ Francs to the online advertisement engine and Bob will pay $v_2(S_2)$. Here $v_1 : 2^{\mathcal{C}} \to \mathbb{R}_+$ and $v_2 : 2^{\mathcal{C}} \to \mathbb{R}_+$ are non-negative monotone submodular functions. The goal in the online ad allocation problem is to design an allocation rule that maximizes $v_1(S_1) + v_2(S_2)$. In this problem you will analyze the competitive ratio of the greedy algorithm, stated below:

Algorithm 1 Greedy algorithm for online ad allocation

$S_1 \leftarrow \emptyset, S_2 \leftarrow \emptyset$
for $i = 1, \ldots, n$ **do**
  if $v_1(c_i|S_1) \geq v_2(c_i|S_2)$ **then**
    $S_1 \leftarrow S_1 \cup \{c_i\}$ ▷ Allocate i-th customer c_i to Alice
  else
    $S_2 \leftarrow S_2 \cup \{c_i\}$ ▷ Allocate i-th customer c_i to Bob
  end if
end for

You will prove that greedy achieves a competitive ratio of $1/2$ by induction on n, the number of customers. We now describe the inductive step. Suppose that the first customer is allocated to Alice (the other case is analogous). Define for $S \subseteq \mathcal{C}$ the functions $v_1'(S) = v_1(S|\{c_1\})$ and $v_2'(S) = v_2(S)$, and let $p = v_1(\{c_1\})$. Let $\mathcal{I} = (v_1, v_2; c_1, \ldots, c_n)$ denote the input instance of the ad allocation problem, and let $\mathcal{I}' = (v_1', v_2'; c_2, \ldots, c_n)$ denote the instance $\mathcal{I}$ with the first customer removed and the functions v_1, v_2 replaced with v_1', v_2'. Let ALG denote the value achieved by greedy on $\mathcal{I}$, and let OPT denote the optimal offline solution on $\mathcal{I}$. Similarly, let ALG' denote the value achieved by greedy on $\mathcal{I}'$, and let OPT' denote the optimal offline solution on $\mathcal{I}'$.

5a *(5 pts)* Prove that $v_j', j = 1, 2$, are non-negative monotone submodular functions.

5b *(10 pts)* Prove that $OPT \leq OPT' + 2p$.

5c *(4 pts)* Show how to complete the proof using **(b)**.

6 *(14 pts)* **Learning from experts.** Recall that the hedge algorithm for learning from experts achieves the following guarantees in the setting of N experts with payoffs $\mathbf{m}^{(t)} \in [-1, +1]^N$ for $t = 1, 2, \ldots, T$. For every $\epsilon \in (0, 1)$, if $\mathbf{p}^{(t)}$ for $t = 1, \ldots, T$ is the distribution picked by Hedge, then for every expert $i = 1, 2, \ldots, N$, one has $\sum_{t=1}^{T} \mathbf{p}^{(t)} \cdot \mathbf{m}^{(t)} \leq \sum_{t=1}^{T} m_i^{(t)} + \frac{\ln N}{\epsilon} + \epsilon T$. Minimizing the additive error term on the right hand side, we set $\epsilon = \sqrt{\frac{\ln N}{T}}$ and obtain

$$\sum_{t=1}^{T} \mathbf{p}^{(t)} \cdot \mathbf{m}^{(t)} \leq \sum_{t=1}^{T} m_i^{(t)} + 2\sqrt{T \ln N}.$$

In this problem you will prove that the above guarantee is essentially the best possible, showing that there exists a constant $c > 0$ such that no algorithm[2] can satisfy

$$\sum_{t=1}^{T} \mathbf{p}^{(t)} \cdot \mathbf{m}^{(t)} \leq \sum_{t=1}^{T} m_i^{(t)} + c\sqrt{T \ln N}. \tag{1}$$

For every $t = 1, 2, \ldots, T$ let $\mathbf{m}^{(t)} \in \{-1, +1\}^N$ denote a vector of independent Bernoulli random variables, and for every $i = 1, 2, \ldots, N$ let $X_i = \sum_{t=1}^{T} m_i^{(t)}$. Assume that N is bounded by a polynomial in T.

6a *(7 pts)* Prove that $\mathbb{E}[\min_{i=1,2,\ldots,N} X_i] = -\Omega(\sqrt{T \log N})$. You may use the following

Theorem 1 (Anti-concentration inequality for binomial random variables) *Let* $Y = \sum_{t=1}^{T} Y_t$, *where* Y_t *are independent random variables such that* $\Pr[Y_t = 1] = \Pr[Y_t = 0] = 1/2$. *Then for* $r \in [0, T/8]$

$$\Pr[Y < T/2 - r] \geq \frac{1}{15} e^{-16r^2/T}.$$

[2] Recall that an algorithm here is a method for choosing $\mathbf{p}^{(t)}$ as a function of $\mathbf{m}^{(s)}, s = 1, \ldots, t-1$, for every $t = 1, \ldots, T$.

6b *(7 pts)* Use **(a)** to prove that no algorithm can satisfy (1) for some absolute constant $c > 0$. *(Hint: recall that* $\mathbf{p}^{(t)}$ *is chosen before observing* $\mathbf{m}^{(t)}$*)*

(This page is intentionally left blank.)