## Question 1

The intersection function $INT\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is $INT(x,y) = (x_1 \text{ AND } y_1) \text{ OR } \cdots \text{ OR } (x_n \text{ AND } y_n)$. Show that

(a) $INT$ requires a (deterministic) read-once branching program of width $2^n$ (in the order $x_1, \dots, x_n, y_1, \dots, y_n$).

**Solution:** If the width is less than $2^n$ there exist two distinct strings $x$ and $x'$ that reach the same state after being read. Without loss of generality, these strings differ in position $i$ where $x_i = 1$ and $x'_i = 0$. If $y$ is the string that has a one in position $i$ and zero everywhere else then $INT(x,y) = 1$ but $INT(x', y) = 0$. However the branching program reaches the same final state on these two inputs so it cannot compute $INT$ correctly on both.

(b) $r_0(INT) \le 2^n$, where $r_0(f)$ is the size $|X| \cdot |Y|$ of the largest rectangle $X \times Y$ for which $f(x,y) = 0$ for all $x \in X$ and $y \in Y$. (**Hint:** Reduce to $IP$)

**Solution:** If $INT(x,y) = 0$ then $x_i \text{ AND } y_i = 0$ for all $i$ so $IP(x,y)$ is also zero. Therefore $r_0(INT) \le r_0(IP)$. In Lecture 3 we showed that if $(x,y)$ is chosen at random from the product set $X \times Y$ then $\mathrm{P}[IP(x,y) = 1] \ge 1/2 - 1/2\sqrt{2^n/|X||Y|}$. If $IP(x,y)$ is zero for all $(x,y) \in X \times Y$ it must be that $1/2 - 1/2\sqrt{2^n/|X||Y|} \le 0$, so $|X||Y| \le 2^n$.

(c) If $f(x,y)$ can be computed by a width-$w$ read-$k$-times branching program then $f$ can evaluate to zero on at most $r_0(f)w^{2k}$ inputs.

**Solution:** We also showed in Lecture 3 that for every such $f$ there exists a partition of its domain $\{0,1\}^n \times \{0,1\}^n$ into at most $w^{2k}$ product sets on which $f$ is constant. The number of inputs on which $f$ equals zero can be at most the number of such product sets on which $f$ evaluates to zero times the number of inputs in each product set, which is at most $2^{2k} \times r_0(f)$.

(d) Use parts (c) and (d) to show that $INT$ requires read-$k$-times branching program width at least $(3/2)^{n/2k}$.

**Solution:** $INT$ takes value zero if and only if $x_i y_i \in \{00, 01, 10\}$ for all $i$. By independence, $INT$ takes value zero with probability $(3/4)^n$. By parts (c) and (d), $(3/4)^n \le r_0(f)w^{2k} \le 2^n w^{2k}$, so $w \ge (3/2)^{n/2k}$.

## Question 2

Let $X$ be an $n$ by $n$ matrix and $f\colon \{0,1\}^{n^2} \to \{0,1\}$ be the function

$$f(X) = \begin{cases} 1, & \text{if } f \text{ has } \textit{exactly one} \text{ column consisting of zeros only,} \\ 0, & \text{otherwise.} \end{cases}$$

Determine the following quantities up to a constant factor (i.e., in $\Theta(\cdot)$ notation). Provide both upper and lower bound proofs.

(a) the deterministic query complexity $D(f)$

**Solution:** This is $n^2$ by an "adversary argument". Answer the queries of the decision tree by zeros, until a whole column is queried, in which case the last column query is answered by 1. If the decision tree has depth strictly less than $n^2$ the queried part of the input $X$ is consistent both with the possibilities $f(X) = 0$ and $f(X) = 1$, so the decision tree cannot compute $f$ on all inputs.

(b) the exact degree $\deg(f)$ when $f$ is viewed as a real-valued polynomial

**Solution:** This is also $n^2$, giving also an alternative proof of part (a). We will represent the polynomial as a function from $\{0,1\}^n \to \{0,1\}$ for convenience as this does not affect the degree. Then $f(X) = g(h(X^1),\ldots,h(X^n))$, where $X^1,\ldots,X^n$ are the columns of $X$, $h$ is the "zeros only" function, and $g$ is the "exactly one one" function. The unique polynomial representations of $h$ and $g$ are

$$h(x_1,\ldots,x_n) = (1-x_1)\cdots(1-x_n) \qquad g(y_1,\ldots,y_n) = \sum_{i=1}^{n} y_i \prod_{j\neq i}(1-y_j).$$

Both $h$ and $g$ contain the degree-$n$ monomials $x_1\ldots x_n$ and $(-1)^{n-1}ny_1\cdots y_n$, respectively, so their composition $f$ must contain the degree-$n^2$ monomial $\prod_{i,j=1}^{n} X_{ij}$.

(c) the sensitivity $\text{sens}(f)$

**Solution:** This is $2n$. If $X$ has exactly two all-zero columns, then changing any of the $2n$ entries in these columns flips the value of $f$ showing that the sensitivity is at least $2n$. We argue it is at most $2n$ by cases. Matrices with 3 or more all-zero columns are insensitive. If there are exactly two, only the $2n$ entries in those two can change the value of $f$ from 0 to 1. If there is exactly one all-zero column, then the entry can be changed from 1 to zero either by destroying this column or creating a new all-zero column. There are $n$ choices for the first possibility and at most $n-1$ for the second as the only way to create an all-zero column is to flip a 1-entry in it provided it is unique, for a total of at most $2n-1$. Finally, if there are no all-zero columns, there can be at most $n$ variables that can be flipped to create one.

(d) **(Optional)** the Monte Carlo randomized query complexity $R(f)$

**Solution:** $\Omega(n^2)$. Justifying this is tricky because the randomized algorithm can be adaptive. We will argue that any algorithm that makes $q$ queries has probability at most $2q/n^2$ at distinguishing between the the distributions $X_4$ of a uniformly random matrix with exactly one 1 per column, and $Y_4$ which is the same as $X_4$ except that a single random column is all zero. When $q < n^2/6$ the advantage is less than $1/3$ so the randomized algorithm must fail.

We start with the fact that the probability that a $q$-query algorithm distingushes a $n^2$-size database $X_1$ with a single random item marked `P` (the prize) from an all-zero database $Y_1$ is (at most) $q/n^2$. Unless the algorithm hits `P` in one of the $q$ queries, which happens with probability at most $q/n$ its views will be identical in $X_1$ and $Y_1$.

Now let $Y_2$ be a random table of size $n^2$ with exactly one 1 per column, and $X_2$ be like $Y_1$ but with an extra random item marked `P`. If the cell marked `P` already contains a 1 the item is marked `P1`. The $q$-query distinguishing advantage of $X_2$ and $Y_2$ can be at most the $q$-query advantage for $X_1$ and $Y_1$, that is $q/n^2$. This is because any distinguisher $D_2$ for the former yields a distinguisher $D_1$ with the same query complexity and advantage for the latter obtained by running $D_2$ on the input for $D_1$ with an additional random 1 in each column. Under this change of input $X_2$ maps to $X_1$ and $Y_2$ maps to $Y_1$.

Next, let $Y_3$ be like $Y_2$ and $X_3$ be like $X_2$ except that the special column containing `P` has the 1-item erased from it. We claim that the $q$-query advantage of distinguishing $X_3$ from $Y_3$ can be at most twice the advantage of distinguishing $X_2$ from $Y_2$, that is at most $2q/n^2$. For suppose $D_3$ distinguishes $X_3$ and $Y_3$ with advantage $\varepsilon$. When $D_3$ samples the first item marked 1 or `P` in the special column, the conditional probability that the item is `P` is half, in which case $D_3$ would distinguish the corresponding inputs in $X_2$ and $Y_2$.

Finally, let $Y_4$ be like $Y_3$ and obtain $X_4$ from $X_3$ by erasing the `P`. Then the $q$-query advantage of distingushing $X_4$ for $Y_4$ is at most that of distingushing $X_3$ from $Y_3$, that is $2q/n^2$. Given any distinguisher $D_4$ for the former, a distingusher from the latter can be obtained by pretending that the answer to the `P`-query is zero.

(e) **(Optional; possible project)** the quantum query complexity $Q(f)$

## Question 3

The *correlation* between two strings $a, b \in \{-1, 1\}^n$ is the number $\langle a, b\rangle/n = (a_1b_1 + \cdots + a_nb_n)/n$ in the range $[-1, 1]$. You will study the classical and quantum query complexities of estimating correlation. An *unbiased estimator* for correlation is an algorithm that accepts $(x_0, x_1)$ with probability $\frac{1}{2} + \frac{1}{2}\langle x_0, x_1\rangle/n$. The input $x = (x_0, x_1)$ is represented as the $2n$-bit string $x_{01} \cdots x_{0n}x_{11} \cdots x_{1n}$. Show that

(a) There exists a 2-query randomized unbiased estimator for correlation.

(b) The estimator queries $x_{0i}$ and $x_{1i}$ for a random $i$ and accepts if they are equal. Given the choice of $i$ acceptance is determined by the value $(1 + x_{0i}x_{1i})/2$. The probability of acceptance is therefore the average of those values, which equals $(1/n)\sum(1 + x_{0i}x_{1i})/2 = \frac{1}{2} + \frac{1}{2}\langle x_0, x_1\rangle/n$.

(c) Any 1-query randomized algorithm has the same acceptance probability on the input distributions

$$\{(X_0, X_1) : X_0 \text{ and } X_1 \text{ are the same random } n\text{-bit string}\} \quad \text{and}$$
$$\{(X_0, X_1) : X_0, X_1 \text{ are independent random } n\text{-bit strings}\}.$$

(**Hint:** Argue this for deterministic algorithms first.)

**Solution:** In both distributions every bit $X_{0i}$ or $X_{1i}$ is an unbiased random bit (1 and $-1$ with probability half each). Therefore the distribution of outputs of any algorithm that queries a single bit will be the same in both cases; it would be the same as if the answer to the algorithm's query was a random bit. This holds for deterministic as well as randomized algorithms. (It is a bit easier to think about randomized algorithms in which $i$ is determined ahead of time which is why I gave the hint.)

(d) There does not exist a 1-query randomized unbiased estimator for correlation.
(**Hint:** Can the algortihm answer correctly in expectation on both distributions in part (b)?)

**Solution:** By part (b) the average acceptance probability of any 1-query algorithm must be the same in both distributions. However the average correlation is zero in the first distribution and one in the second one. Therefore the algorithm cannot be estimating correlation without bias.

(e) The quantum algorithm

Measure the first qubit of $H_1\Phi^x|+\rangle$ and accept if it is zero

is a (1-query) unbiased estimator for correlation. Here, $|+\rangle$ is the state $(|01\rangle + \cdots + |0n\rangle + |11\rangle + \cdots + |1n\rangle)/\sqrt{2n}$ and $H_1$ is the Hadamard gate applied to the first qubit $|b\rangle$. In $\pm 1$ bit representation $\Phi^x$ is the phased-query gate $\Phi^x|bi\rangle = x_{bi}|bi\rangle$.

**Solution:** After the phased query the algorithm is in state $\Phi^x|+\rangle = (\sum x_{bi}|bi\rangle)/\sqrt{2n}$. After the Hadamard query the state becomes

$$H_1\Phi^x|+\rangle = \frac{1}{\sqrt{n}}\sum_i \frac{x_{0i} + x_{1i}}{2}|0i\rangle + \frac{x_{0i} - x_{1i}}{2}|1i\rangle.$$

For each $i$, this superposition contains exclusively state (+ or $-$)$|0i\rangle$ if $x_{0i} = x_{1i}$ and state (+ or $-$)$|0i\rangle$ if $x_{0i} \neq x_{1i}$. Therefore the probability of measuring zero in the first register is exactly the fraction of indices $i$ for which $x_{0i} = x_{1i}$, which equals $\frac{1}{2} + \frac{1}{2}\langle x_0, x_1\rangle/n$ by part (a).

(f) **(Optional)** There is a 1-query quantum unbiased estimator of $\frac{1}{n}\sum A_{ij}x_{0i}x_{1j}$ for every $n \times n$ orthogonal (real unitary) matrix $A$.

**Solution:** Let $A'$ be the linear transformation that applies $A$ to the second register if the first register is $|1\rangle$ and applies the identity to the second register if the first register is $|0\rangle$. Then $A'$ is unitary because it is invariant on the subspaces spanned by $|01\rangle, \ldots, |0n\rangle$ and $|11\rangle, \ldots, |1n\rangle$ and it is unitary on each ($A$ on the first, the identity on the second). The algorithm is "Measure the first qubit of $H_1A'\Phi^x|+\rangle$ and accept if it is zero". The state $A'\Phi^x|+\rangle$ has amplitude $x_{0i}$ in direction $|0i\rangle$ and amplitude $(Ax_1)_i$ in direction

$x_{1i}$. By a calculation as in (c), the amplitude of $|0i\rangle$ in $H_1 A' \Phi^x |+\rangle$, i.e., the value $\langle q0i | H \rangle_1 A' \Phi^x |+\rangle$, equals $(x_{0i} + (Ax_1)_i)/2\sqrt{n}$. Therefore the probability of measuring zero equals

$$\frac{1}{n}\sum_{i=1}^{n} \frac{(x_{0i} + (Ax_1)_i)^2}{4} = \sum \frac{x_{0i}^2}{4n} + \sum \frac{(Ax_1)_i^2}{4n} + \sum \frac{x_{0i}(Ax_1)_i}{2n}$$

The first term equals $1/4$ because $x0i^2 = 1$ for all $i$. The second term also equals $1/4$ because the orthogonal matrix $A$ is length-preserving so $\sum (Ax_1)_i^2 = \sum x_{1i}^2 = n$. The last term equals $\sum A_{ij} x_{0i} x_{1j}/2n$, so the acceptance probability is $1/2 + (\sum A_{ij} x_{0i} x_{1j}/2n$ as desired.