# Final In-Class Exam

**( 4:05 PM – 5:20 PM : 75 Minutes )**

- This exam will account for either 15% or 30% of your overall grade depending on your relative performance in the midterm and the final. The higher of the two scores (midterm and final) will be worth 30% of your grade, and the lower one 15%.
- There are three (3) questions, worth 75 points in total. Please answer all of them in the spaces provided.
- There are 14 pages including four (4) blank pages and one (1) page of appendices. Please use the blank pages if you need additional space for your answers.
- The exam is *open slides* and *open notes*.

**GOOD LUCK!**

| Question | Pages | Score | Maximum |
|---|---|---|---|
| 1. Parallel DFT | 2–5 | | 30 |
| 2. Trapping the Median | 7–9 | | 30 |
| 3. Files on Compact Discs | 12 | | 15 |
| Total | | | 75 |

NAME: ______________________________

**Question 1.** **[ 30 Points ] Parallel DFT.** Given the coefficient vector $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ of a polynomial $P(x) = a_0 + a_1 x + a_2 x^2 + \ldots a_{n-1} x^{n-1}$, the Par-Rec-DFT function shown below (in Figure 1) computes another vector $\langle y_0, y_1, \ldots, y_{n-1} \rangle$, where $y_i = P((\omega_n)^i)$ and $\omega_n$ is the primitive $n$-th root of unity. The output vector $\langle y_0, y_1, \ldots, y_{n-1} \rangle$ is called the *Discrete Fourier Transform* (DFT) of the input vector $\langle a_0, a_1, \ldots, a_{n-1} \rangle$. We assume for simplicity that $n$ is a power of 2.

Par-Rec-DFT( $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ )

(Input is the coefficient vector $\langle a_0, a_1, \ldots, a_{n-1} \rangle$ of a polynomial $P(x) = a_0 + a_1 x + a_2 x^2 + \ldots a_{n-1} x^{n-1}$. The output is another vector $\langle y_0, y_1, \ldots, y_{n-1} \rangle$, where $y_i = P((\omega_n)^i)$ and $\omega_n$ is the primitive $n$-th root of unity. We assume for simplicity that $n$ is a power of 2.)

***if*** $n = 1$ ***then return*** $\langle a_0 \rangle$
***else***
    $\langle y_0^{even}, y_1^{even}, \ldots, y_{\frac{n}{2}-1}^{even} \rangle \leftarrow$ ***spawn*** Par-Rec-DFT( $\langle a_0, a_2, \ldots, a_{n-2} \rangle$ )  {*even numbered coefficients*}
    $\langle\ y_0^{odd},\ y_1^{odd}, \ldots,\ y_{\frac{n}{2}-1}^{odd} \rangle \leftarrow$ Par-Rec-DFT( $\langle a_1, a_3, \ldots, a_{n-1} \rangle$ )  {*odd numbered coefficients*}
    ***sync***
    $w_0 \leftarrow 1$
    ***parallel for*** $j \leftarrow 1$ ***to*** $\frac{n}{2} - 1$ ***do***
        $w_j \leftarrow n$-th primitive root of unity  $\left\{ i.e.,\ w_j \leftarrow e^{\frac{2\pi i}{n}},\ where\ i = \sqrt{-1} \right\}$
    $\langle s_0, s_1, \ldots, s_{\frac{n}{2}-1} \rangle \leftarrow$ Prefix-Sum( $\langle w_0, w_1, \ldots, w_{\frac{n}{2}-1} \rangle$, $\times$ )  {*prefix sum using the product operator*}
    ***parallel for*** $i \leftarrow 0$ ***to*** $\frac{n}{2} - 1$ ***do***  $\left\{ compute\ y\ from\ y^{even}\ and\ y^{odd} \right\}$
        $y_j \leftarrow y_j^{even} + s_j y_j^{odd}$
        $y_{\frac{n}{2}+j} \leftarrow y_j^{even} - s_j y_j^{odd}$
    ***return*** $\langle y_0, y_1, \ldots, y_{n-1} \rangle$

Figure 1: A parallel recursive divide-and-conquer algorithm for computing the Discrete Fourier Transform (DFT) of a 1D array (vector).

1($a$) [ **10 Points** ] Write down a recurrence relation describing the work done (i.e., $T_1$) by Par-Rec-DFT, and solve it.

1(*b*) [ **10 Points** ] Write down a recurrence relation describing the span (i.e., $T_\infty$) of Par-Rec-DFT, and solve it. Please assume that the span of a ***parallel for*** loop with $n$ iterations is $\mathcal{O}(\log n + k)$, where $k$ is the maximum span of a single iteration.

1($c$) [ **10 Points** ] Find the parallel running time (i.e., $T_p$) and parallelism of Par-Rec-DFT.

Use this page if you need additional space for your answers.

**QUESTION 2.** [ **30 Points** ] **Trapping the Median.** Given an array $A[1:n]$ of $n$ distinct numbers as input, the function TRAP-MEDIAN shown below (in Figure 2) returns another array $A'[1:n']$ containing $n'$ distinct numbers from $A$ such that w.h.p. in $n$, $n' = \mathcal{O}\left(n^{\frac{3}{4}}\right)$ and $A'$ still includes the median of $A$. We assume for simplicity that $n$ is an odd positive integer.

TRAP-MEDIAN( $A$, $n$ )
(Input is an array $A[1:n]$ of $n$ distinct numbers, where $n$ is an odd positive integer. Output is an array $A'[1:n']$ containing $n'$ distinct numbers from $A$ such that w.h.p. in $n$, $n' = \mathcal{O}\left(n^{\frac{3}{4}}\right)$ and $A'$ contains the median of $A$.)

1. choose each entry of $A$ with probability $n^{-\frac{1}{4}}$ independent of others, and collect them in an array $B$
2. $m \leftarrow |B|$
3. ***if*** $\left\lfloor \frac{m}{2} - \sqrt{n} \right\rfloor > 0$ ***and*** $\left\lceil \frac{m}{2} + \sqrt{n} \right\rceil \leq m$ ***then***
4.     sort $B$ using an optimal sorting algorithm
5.     $x \leftarrow B\left[\left\lfloor \frac{m}{2} - \sqrt{n} \right\rfloor\right]$, $y \leftarrow B\left[\left\lceil \frac{m}{2} + \sqrt{n} \right\rceil\right]$
6.     $r_x \leftarrow$number of items in $A$ with value $\leq x$
7.     $r_y \leftarrow$number of items in $A$ with value $\leq y$
8.     ***if*** $r_x < \frac{n+1}{2} < r_y$ ***then***     {*if $x$ is smaller than the median of $A$, and $y$ is larger than the median*}
9.         $n' \leftarrow$number of items in $A$ with value between $x$ and $y$     {*count each $z$ in $A$ with $x < z < y$*}
10.         allocate an array $A'[1:n']$
11.         scan $A$ again, and copy each number $z \in (x, y)$ from $A$ to $A'$
12.         ***return*** $A'$
13.     ***else return*** NIL
14. ***else return*** NIL

Figure 2: Trap the median of $n$ numbers in a set of size asymptotically smaller than $n$.

2$(a)$ [ **12 Points** ] Prove that $n^{\frac{3}{4}} - n^{\frac{7}{16}} < m < n^{\frac{3}{4}} + n^{\frac{7}{16}}$ holds w.h.p. in $n$ (in Step 2).

**Hint:** *Compute the expected value of $m$, and then use Chernoff bounds (or even Chebyshev's inequality) to show that both $Pr\left(\ m \leq n^{\frac{3}{4}} - n^{\frac{7}{16}}\ \right)$ and $Pr\left(\ m \geq n^{\frac{3}{4}} + n^{\frac{7}{16}}\ \right)$ are low.*

$2(b)$ [ **12 Points** ] Show that $r_x < \frac{n+1}{2} < r_y$ holds w.h.p. in $n$ (in Step 8). You may assume that $m = \Theta\left(n^{\frac{3}{4}}\right)$ holds w.h.p. in $n$ (from part $2(a)$).

**Hint:** *Associate a 0-1 random variable to each element in $B$. Each variable takes the value 1 (when the corresponding element is smaller than the median of $A$) with probability $p = \frac{1}{2}$, and 0 (when the corresponding element is larger than the median of $A$) with probability $q = 1-p = \frac{1}{2}$. Let $X$ be the sum of these $m$ mutually independent random variables which represents the the number of elements in $B$ that are smaller than the median of the elements in $A$. Now if you can prove that $X > \frac{m}{2} - \sqrt{n}$ holds w.h.p. in $n$ (perhaps using a Chernoff bound or even Chebyshev's inequality) then the element $x$ chosen from location $\frac{m}{2} - \sqrt{n}$ of $B$ (in Step 5) must be smaller than the median of $A$ (i.e., $r_x < \frac{n+1}{2}$) w.h.p. in $n$. Similarly for $r_y$.*

2$(c)$ [ **6 Points** ] Show that the running time of TRAP-MEDIAN is $\mathcal{O}(n)$ w.h.p. in $n$. You may use the results you proved in parts 2$(a)$ and 2$(b)$, if needed.

Use this page if you need additional space for your answers.

Use this page if you need additional space for your answers.

**Question 3.** [ **15 Points** ] **Files on Compact Discs.** I have $m > 0$ files and a set $S$ of $n > 1$ compact discs (CDs). I have copied each file to exactly two of the CDs in $S$. Different files may be copied to different CD pairs. Now given that for each file I know the two CDs I copied them to, I want to find a subset $S' \subseteq S$ such that each file is contained in at least one CD of $S'$, and $|S'|$ is as small as possible.

3($a$) [ **15 Points** ] Give a polynomial-time 2-approximation algorithm for solving this problem. In other words, the size of the subset returned by your algorithm must not be more than 2 times larger than the size of the subset returned by an optimal algorithm.

**Hint:** *Map this problem to one of the problems you saw in the class lectures on approximation algorithms, and use the* 2*-approximation algorithm for that problem to solve the problem in this task.*

Use this page if you need additional space for your answers.

# Appendix I: Useful Tail Bounds

**Markov's Inequality.** Let $X$ be a random variable that assumes only nonnegative values. Then for all $\delta > 0$, $Pr\left[X \geq \delta\right] \leq \frac{E[X]}{\delta}$.

**Chebyshev's Inequality.** Let $X$ be a random variable with a finite mean $E[X]$ and a finite variance $Var[X]$. Then for any $\delta > 0$, $Pr\left[|X - E[X]| \geq \delta\right] \leq \frac{Var[X]}{\delta^2}$.

**Chernoff Bounds.** Let $X_1, \ldots, X_n$ be independent Poisson trials, that is, each $X_i$ is a 0-1 random variable with $Pr[X_i = 1] = p_i$ for some $p_i$. Let $X = \displaystyle\sum_{i=1}^{n} X_i$ and $\mu = E[X]$. Following bounds hold:

Lower Tail:

- for $0 < \delta < 1$, $Pr\left[X \leq (1-\delta)\mu\right] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu}$
- for $0 < \delta < 1$, $Pr\left[X \leq (1-\delta)\mu\right] \leq e^{-\frac{\mu\delta^2}{2}}$
- for $0 < \gamma < \mu$, $Pr\left[X \leq \mu - \gamma\right] \leq e^{-\frac{\gamma^2}{2\mu}}$

Upper Tail:

- for any $\delta > 0$, $Pr\left[X \geq (1+\delta)\mu\right] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{\mu}$
- for $0 < \delta < 1$, $Pr\left[X \geq (1+\delta)\mu\right] \leq e^{-\frac{\mu\delta^2}{3}}$
- for $0 < \gamma < \mu$, $Pr\left[X \geq \mu + \gamma\right] \leq e^{-\frac{\gamma^2}{3\mu}}$

# Appendix II: The Master Theorem

Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be a function, and let $T(n)$ be defined on the nonnegative integers by the recurrence

$$T(n) = \begin{cases} \Theta(1), & \text{if } n \leq 1, \\ aT\left(\frac{n}{b}\right) + f(n), & \text{otherwise,} \end{cases}$$

where, $\frac{n}{b}$ is interpreted to mean either $\left\lfloor \frac{n}{b} \right\rfloor$ or $\left\lceil \frac{n}{b} \right\rceil$. Then $T(n)$ has the following bounds:

**Case 1:** If $f(n) = \mathcal{O}\left(n^{\log_b a - \epsilon}\right)$ for some constant $\epsilon > 0$, then $T(n) = \Theta\left(n^{\log_b a}\right)$.

**Case 2:** If $f(n) = \Theta\left(n^{\log_b a} \log^k n\right)$ for some constant $k \geq 0$, then $T(n) = \Theta\left(n^{\log_b a} \log^{k+1} n\right)$.

**Case 3:** If $f(n) = \Omega\left(n^{\log_b a + \epsilon}\right)$ for some constant $\epsilon > 0$, and $af\left(\frac{n}{b}\right) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large $n$, then $T(n) = \Theta\left(f(n)\right)$.