NATIONAL UNIVERSITY OF SINGAPORE
SCHOOL OF COMPUTING

Midterm Assessment for CS3230

March 3, 2022 Time Allowed: 80 minutes (2:15–3:35pm)

INSTRUCTIONS:

- This paper consists of **FOUR** questions for a total of 60 points. Please read instructions to each question carefully.
- This is an **OPEN BOOK/NOTES** examination.
- Do **NOT** use the internet for help while taking the examination. Any evidence otherwise will result in **severe penalties**.
- Do **NOT** seek the help of others during the examination. Any evidence otherwise will result in **severe penalties**.
- You have **15** minutes (**3:35pm** to **3:50pm**) to upload the scan of your hand-written solutions to LumiNUS at `Midterm/Submissions`.
- You need to generate a **separate PDF file** for each of the four questions on the assessment. The four files must be named as `Axxxxxxxy_Q1.pdf`, `Axxxxxxxy_Q2.pdf`, `Axxxxxxxy_Q3.pdf` and `Axxxxxxxy_Q4.pdf` where `Axxxxxxxy` is your student number.
- Submit a file even if you do not attempt a particular question. It is **your own responsibility** to ensure that the PDFs correctly contain your solutions.
- Recall that lg denotes the logarithm with base 2.

QUESTIONS:

1. (12 points) For each question in this part, write "True", "False", or "Don't know". Each correct answer is worth 2 points; each answer "Don't know" guarantees 1 point. No need to give justifications.

   (a) $n! = O(2022^n)$

   (b) $n^2 = o(4^{\lg n})$

   (c) $n^{0.5} = \omega(n^{0.1})$

   (d) $1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n} = \Omega(\lg n)$

   (e) $n^{n-1} = \Theta(n^n)$

   (f) There exists a constant $c > 0$ such that the running time of the following program for computing the n-th Fibonacci number is $O(n^c)$.

```
FIBONACCI(n)
if n = 0 or n = 1
    return n
else
    return FIBONACCI(n − 1) + FIBONACCI(n − 2)
```

2. (18 points) For each question in this part, provide the requested answer in the simplest form possible. No need to give justifications. (However, for (e) and (f), you *may* write down a brief justification; partial credit may be awarded.)

   For questions (a), (b), (c), and (d), solve the following recurrences by providing tight asymptotic bounds using the Θ-notation. You can ignore the fact that $n/2$, $2n/3$, and $\sqrt{n}$ may not be integers.

   (a) (2 points) $T(n) = T(n-1) + 3230n$

   (b) (2 points) $T(n) = 7T(n/2) + n^3$

   (c) (3 points) $T(n) = T(2n/3) + \lg n + 1$

   (d) (3 points) $T(n) = T(\sqrt{n}) + \sqrt{n}$

   For question (e), suppose that you throw n balls uniformly and independently at random into n bins.

   (e) (4 points) What does the expected fraction of bins with exactly three balls converge to as $n \to \infty$?

   For question (f), suppose that you receive a stream of numbers. Upon receiving the first number x_1, you set a random variable X to be x_1. For $k \geq 2$, upon receiving the k-th number x_k, you set $X = x_k$ with probability $\frac{1}{k^2}$, and keep X unchanged with the remaining probability $1 - \frac{1}{k^2}$.

   (f) (4 points) Assume that the numbers $x_1, x_2, \ldots, x_{2022}$ are distinct. After you have performed this procedure for x_{2022}, what is the probability that $X = x_1$?

3. (10 points) Prove that for all positive integers $n \geq 2$ and $m \geq 2$, there exists a family $\mathcal{H}$ of hash functions mapping $N = \{1, 2, \ldots, n\}$ to $M = \{1, 2, \ldots, m\}$ such that for any two distinct $x, y \in N$,

$$\Pr_{h \in \mathcal{H}}[h(x) = h(y)] < \frac{1}{m}.$$

Note that the inequality above is strict.

(**Hint:** Start by thinking about the family of *all* functions from N to M. Can you modify this family slightly?)

4. (20 points) Your friend Serena organized a tennis event at her house last weekend. The event involved 15 players $P_1, P_2, \ldots, P_{15}$, every two of whom played each other exactly once. (So the total number of matches was $\binom{15}{2} = 105$.) Each match had exactly one winner (i.e., there was no tie/draw).

   You did not attend the event, but would like to find out some information by asking Serena a number of questions. With each question, you can choose two players P_i, P_j and ask Serena "Did P_i beat P_j?" Your next question can depend on Serena's answers to your previous questions, but you cannot use randomness (e.g., by flipping a coin to decide your next question).

   (a) (10 points) Determine, with proof, a small number k such that by asking k questions, you can always find out whether there exists a player who beat all other 14 players. You should try to make k as small as possible, but do not need to show that your k is optimal.

   (**Hint:** Use ideas from the algorithm for finding the second largest element in an array from Lecture 1.)

   (b) (10 points) Determine, with proof, the **smallest** number k such that by asking k questions, you can always find out the set of players who won the highest number of matches among all players. (If there is more than one such player, you must find all of them.)