Exam 1: Untimed Part

Name: _____

February 25, 2025
CSCI 4104: Spring 2025

Section: _____

Instructions: This is a the untimed part of the exam. Your job is to explain why the errors on the exam are incorrect. **DO NOT JUST WRITE THE CORRECT ANSWER. THAT DOES NOT EXPLAIN WHY SOMETHING IS INCORRECT.**

Problem 1. Which of the following is true about a tuple in Python?

(a) They are a set

(b) Tuples are ordered

(c) They have the same syntax as a list

(d) Their size can be changed after creating them

Problem 2. How do you define a function in Python?

(a) def functionFoo:

(b) def functionFoo():

(c) def functionFoo()

(d) def: functionFoo()

Problem 3. What is the output of the following code?

```
x = 3
x = "string"
print(x)
```

(a) string

(b) 3

(c) Error in Code

(d) Nothing is printed because Python doesn't know what type x is

Problem 4. In simple terms, Big O is a notation used to define the asymptotic _____ growth of an algorithm.

(a) exact

(b) upper bound

(c) both the upper and lower bound

(d) lower bound

Problem 5. When preforming an Aggregate Analysis, the computation being done at the end of the analysis is

(a) The number of operations divided by the total cost

(b) The total cost multiplied by the number of operations

(c) The number of operations multiplied by the total cost

(d) Total cost divided by the number of operations

1 Short Answer (24 points)

Problem 6. Give 2 differences between declaring a variable in Python vs declaring a variable in Java.

You have to use single quotes in Python for a string. In Java, you must use double quotes. In Java, you can change the variable type after the variable is declared. In Python, once a variable is declared it must stay the same type.

Problem 7. How does Big O and Little o relate to each other?

Little o implies Big O.

Problem 8. Give a big-O estimate for $f(n) = (n+3)(n+6) + n\log(3n)$, where n is a positive integer. **You don't need to give a formal proof, but you must show work.**

$$\begin{aligned} f(n) &= (n+3)(n+6) + n\log(3n) \\ &= n^2 + 9n + 18 + 3n\log(n) \\ &= O(n\log(n)) \end{aligned}$$

Problem 9. What is the difference between an aggregate analysis and an average case analysis?

Aggregate analysis is more accurate than average case analysis. Average case analysis is based off of probability and best case analysis. Aggregate analysis does not use probability. It does use best case analysis.

2 Analyses (51 Points)

Problem 10. (10 Points) Using the limit definition, determine whether $f(n) = 3n + \log(2n)$ is O, o, Ω, ω, or Θ of $g(n) = 5n^2$.

$$\lim_{n\to\infty} \frac{3n + \log(2n)}{5n^2}$$
$$= \lim_{n\to\infty} \frac{3}{10n}$$
$$= \infty$$

Therefore, $f(n) = \omega(g(n))$.

Problem 11. (10 Points) Using the limit definition, determine whether $f(n) = n^2 + 2n$ is O, o, Ω, ω, or Θ of $g(n) = n^3 + 5n$.

$$\lim_{n\to\infty} \frac{n^3 + 5n}{n^2 + 2n}$$
$$= \lim_{n\to\infty} \frac{3n^2 + 5}{2n + 2}$$
$$= \lim_{n\to\infty} \frac{6n}{2}$$
$$= 3$$

Therefore, $f(n) = \Theta(g(n))$.

Problem 12. (16 points) Prove $f(n) = n^3 - 16n$ is $\Theta(n^3)$ by showing $f(n)$ is $O(n^3)$ and $\Omega(n^3)$.

First we will show $f(n)$ is $O(n^3)$. Consider,

$$\frac{f(n)}{g(n)} = \frac{n^3 - 16n}{n^3} = 1.$$

Therefore $C = 1$. Since a C was found, $f(n)$ is $O(n^3)$.
Now, we will show $f(n)$ is $\Omega(n^3)$. Let $k = 1$ and thus $n > 1$. So,

$$\frac{g(n)}{f(n)} = \frac{n^3}{n^3 - 16n} = 1.$$

Therefore, $C = 1$. Since a C was found, $f(n)$ is $\Omega(n^3)$.
Since we showed $f(n)$ is $O(n^3)$ and $\Omega(n^3)$, then $f(n) = n^3 - 16n$ is $\Theta(n^3)$.♠

Problem 13. (15 points) Consider a sequence of n operations on a queue. Enqueue (inserts an element at the end of the queue) takes O(1) time. Dequeue (removes and returns an element that is at the front end of the queue) takes O(1) time. Consider that after every 3 enqueues, the queue must double in size. This takes O(n) time. Calculate the amortized time for this sequence.

Enqueue takes O(1) time and Dequeue takes O(1) time. We do 3 Enqueues and then a Dequeue. So, this is a total of $3n+n = 4n$ operations. This takes O(n) time. The second part is that the queue is doubling in size after the 3 Enqueues. This also takes O(n) time. Thus, we have a runtime of

$$\frac{O(n)}{O(n)} = O(1).$$