University of Waterloo
CS240 Winter 2025
Assignment 1

Due Date: Tuesday, January 21 at 5:00pm

Please read `https://student.cs.uwaterloo.ca/~cs240/w25/assignments.phtml#guidelines` for guidelines on submission. **Each question must be submitted individually to Crowdmark.** Submit early and often.

Grace period: submissions made before 11:59PM on January 21 will be accepted without penalty. Your last submission will be graded. Please note that submissions made after 11:59PM **will not be graded** and may only be reviewed for feedback.

Reminder: all logarithms are in base 2 unless stated otherwise.

Question 1 [1+1+1+1=4 marks]

Find the mistake in each of the following proofs from the definition of the order notation.

a) Let $f(n) = 50n \log n + 4n$ and $g(n) = n \log n$. Show that $f(n)$ is $O(g(n))$.

Proof: For all $n \geq 1$, we have that $50n \log n + 4n \leq 50n \log n + 4n \log n = 54n \log n$. So take $n_0 = 1$ and $c = 54$.

b) Let $f(n) = 50n^2 + 4n$ and $g(n) = n^3$. Show that $f(n)$ is $O(g(n))$.

Proof: For all $n \geq 0$, we have that $50n^2 + 4n \leq 50n^3 + 4n^3 = 54n^3$. So take $n_0 = 0$ and $c = 54$.

c) Let $f(n) = 2n^2 + 4n$ and $g(n) = n^2$. Show that $f(n)$ is $\Theta(g(n))$.

Proof: For all $n \geq 10$, we have that $2n^2 + 4n \leq 2n^2 + 4n^2 = 6n^2$. So take $n_0 = 10$ and $c = 6$.

d) Let $f(n) = 2n^2 - 4n$ and $g(n) = n^2$. Show that $f(n)$ is $\Omega(g(n))$.

Proof: For all $n \geq 1$ we have that $2n^2 - 4n \geq 2n^2$. So take $n_0 = 1$ and $c = 2$.

Question 2 [3+3+3+3=12 marks]

Provide a complete proof of the following statements from first principles (i.e., using the original definitions of order notation).

a) $2^{2^n} \log n + 2^{2^{100}} \cdot 2^n \in O(2^{2^n} \log n)$

b) $0.5n^4 - 10n^{3.99} - 15n \in \Omega(n^4)$

c) $n^5 + 2^{100}n^2 + 2^{100} \in o(n^5 \log n)$

d) $n^2 + n \in \omega(n^{1.9999})$

Question 3 [3+3=6 marks]

Prove or disprove each of the following statements. All functions map from $\mathbb{N} \to \mathbb{R}^+$.

a) $\frac{f(n)g(n)+\log g(n)}{f(n)+2g(n)}$ is $\Theta(\min\{f(n), g(n)\})$.

b) $(\log n)^{\log n} \in O(n^2)$.

Question 4 [3 marks]

Arrange the following functions by the order of their growth rates:

$$4^n, 2^{\cos n}, 3^n, 2^{n \log n}, (\log n)^{100}, n^2, \frac{n^2 + n^{10}}{\log n}, (n^3 + \log n)^{(n^2 - n^4 - 10)}$$

This question will be marked as an 'all or nothing' question. No justification is required.

Question 5 [5 marks]

Define a function $f : \mathbb{N} \to \mathbb{R}^+$ such that f satisfies these three conditions:

(1) $f(n) \in O(n^4)$

(2) $f(n) \notin \Theta(n^4)$

(3) $f(n) \notin o(n^4)$

Justify your answer.

Question 6 [4+4=8 marks]

Analyze the following pieces of pseudocode and give a tight (Θ) bound on the running time as a function of n. Show your work. In all cases, n is assumed to be a positive integer.

a)
```
x = 0
for i = 1 to n do
    for j = i to n do
        if i == j then
            k = n
            while k > 0 do
                x = x + 1
                k = k/3
```

b)
```
sum = 0
for i = 1 to n
     sum = sum + i
     j = i
     while j > 0
        sum = sum + j
        j = j - 1
        k = j
        while k <  j - i
            sum = sum + 1
            k = k + 1
```

Question 7 [4 marks]

Analyze the best case time efficiency of the following algorithm. A is an array of size n storing integers in the range from 1 to n. Furthermore, if integer i occurs in array A, then it occurs at most $\sqrt{n}$ times. You can assume $\sqrt{n}$ is an integer.

```
Algorithm Lazy(A,n)

i = 0
sum = 0
while i < n
   sum = sum + A[i]
   sum2 = 0
   while sum2 < sum
      sum2 = sum2 + 1
   i = i + 1
```

Question 8 [3 marks]

In the sum below, replace '*' with the correct expression to derive the lower bound of $\left(\frac{n}{3}\right)^2 c$ for the sum. You can assume $n/3$ is an integer. Explain your work.

$$\sum_{i=1}^{n}\sum_{j=1}^{i} c \geq \sum_{i=*}^{*}\sum_{j=*}^{*} c = \left(\frac{n}{3}\right)^2 c$$