Your last recorded submission was on 2024-01-30, 21:19 IST **Due date: 2024-02-07, 23:59 IST.**

All questions carry equal weightage. You may submit as many times as you like within the deadline. Your final submission will be graded.

1) An image processing application begins with two n×n matrices A and B. The first phase of preprocessing the inputs takes $O(n^2)$ steps for each of A and B. The second step involves a convolution of A and B to yield a new matrix C in time $O(n^3)$. This is followed by an edge detection phase that takes times $O(n^2)$ for matrix C. What is the most accurate and concise description of the complexity of the overall algorithm? ***2 points***

- ○ $O(n^2)$
- ◉ $O(n^3)$
- ○ $O(n^2+n^3)$
- ○ $O(n^5)$

2) We are trying to determine the worst case time complexity of a library function that is provided to us, whose code we cannot read. We test the function by feeding large numbers of random inputs of different sizes. We find that for inputs of size 400 and 4,000, the function always returns well within one second, but for inputs of size 40,000 it sometimes takes a couple of seconds and for inputs of size 400,000 it sometimes takes a few minutes. What is a reasonable conclusion we can draw about the worst case time complexity of the library function? (You can assume, as usual, that a typical desktop PC performs 10^9 basic operations per second.) ***2 points***

- ○ $O(n \log n)$
- ◉ $O(n^2)$
- ○ $O(n^3)$
- ○ $O(n^3 \log n)$

3) Suppose f(n) is $252n^3+164n^2+507$ and g(n) is $n^4 + 5n + 12$. Let h(n) be a third, unknown function. Which of the following is **not** possible. ***2 points***

- ○ h(n) is O(f(n)) and h(n) is also O(g(n))
- ○ h(n) is O(f(n)) but h(n) is not O(g(n))
- ◉ h(n) is O(g(n)) but h(n) is not O(f(n))
- ○ h(n) is not O(f(n)) and h(n) is also not O(g(n))

3) Suppose f(n) is $252n^3+164n^2+507$ and g(n) is $n^4 + 5n + 12$. Let h(n) be a third, unknown function. Which of the following is **not** possible. ***2 points***

- ○ h(n) is O(f(n)) and h(n) is also O(g(n))
- ○ h(n) is O(f(n)) but h(n) is not O(g(n))
- ◉ h(n) is O(g(n)) but h(n) is not O(f(n))
- ○ h(n) is not O(f(n)) and h(n) is also not O(g(n))

4) How many times is the comparison `i >= n` performed in the following program? ***2 points***

```
int i = 300, n = 150;
main(){
  while (i >= n){
    i = i-2;
    n = n+1;
  }
}
```

- ○ 50
- ◉ 51
- ○ 52
- ○ 53

5) If T(n) is $O(n^2 \sqrt{n})$ which of the following is **false**? ***2 points***

- ◉ T(n) is $O(n^2 \log n)$
- ○ T(n) is $O(n^3)$
- ○ T(n) is $O(n^3 \log n)$
- ○ T(n) is $O(n^4)$

You may submit any number of times before the due date. The final submission will be considered for grading.