An image processing application begins with two \( n \times n \) 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 \( O(n^2) \). This is followed by an edge detection phase that takes times \( O(n^3) \) for matrix \( C \). What is the most accurate and concise description of the complexity of the overall algorithm?

- \( O(n^2) \)
- \( O(n^3) \)
- \( O(n^2+n^3) \)
- \( O(n^5) \)

No, the answer is incorrect.

Score: 0

Feedback:

When there are multiple phases in sequence, the largest of the phases dominates the overall complexity. Here the second phase is \( O(n^2) \) and the first and third phases are \( O(n^3) \).

Accepted Answers:

\( O(n^2) \)

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 300 and 3,000, the function always returns well within one second, but for inputs of size 30,000 it sometimes takes about 1 second and for inputs of size 300,000 it sometimes takes 1-2 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.)
3) Suppose \( f(n) \) is \( 2n^3 + 4n + 5 \) and \( g(n) \) is \( 7n^5 + 5n^3 + 12 \). Let \( h(n) \) be a third, unknown function. 2 points Which of the following is not possible.

- \( h(n) \) 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 \( O(g(n)) \)

No, the answer is incorrect.  
Score: 0  
Feedback:  
Since \( f(n) \) is \( O(g(n)) \), if \( h(n) \) is \( O(f(n)) \) it must also be \( O(g(n)) \). All other combinations are possible.  
Accepted Answers:  
\( h(n) \) is \( O(f(n)) \) but \( h(n) \) is not \( O(g(n)) \)

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

```c
int i = 200, n = 80;
main(){
    while (i >= n){
        i = i-2;
        n = n+1;
    }
}
```

- 40
- 41
- 42
- 43

No, the answer is incorrect.  
Score: 0  
Feedback:  
After 40 iterations, \( i \) is 200-80=120 and \( n \) is 80+40=120. At this point, \( i==n \) so the 41st iteration succeeds  
The next test of the while condition fails and exits the loop. Hence, overall the while condition is checked 42 times.  
Accepted Answers:  
42

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 \( O(n^4) \)
No, the answer is incorrect.
Score: 0
Feedback:
- √ n is not O(log n)
Accepted Answers:
- T(n) is O(n^2 log n)