Started on	Saturday, 3 February 2024, 9:43 AM
State	Finished
Completed on	Saturday, 3 February 2024, 10:11 AM
Time taken	28 mins 16 secs
Marks	8.00/8.00
Grade	**10.00** out of 10.00 (**100**%)

Question 1

Correct

Mark 1.00 out of 1.00

What is the time complexity of the following code?

```
int i, j, k = 0;
for (i = N / 2; i <= N; i++) {
    for (j = 2; j <= N; j = j * 2) {
        k = k + N/ 2;
    }
}
```

- ○ a. O(N*log(N)) ✔
- ○ b. O(N*Sqrt(N))
- ○ c. O(N*N)
- ○ d. O(N)

The correct answer is:
O(N*log(N))

Question 2

Correct

Mark 1.00 out of 1.00

The Worst case occur in linear search algorithm when

○ a.   Item is the last element in the array

○ b.   Item is not in the array at all

◉ c.   Item is the last element in the array or is not there at all ✔

○ d.   Item is somewhere in the middle of the array

The correct answer is: Item is the last element in the array or is not there at all

Question 3

Correct

Mark 1.00 out of 1.00

What is the big O time complexity of the following code?

```
int a = 0;
int i = 1;
while( i < N)
{
    a = a + 2*i
    i = i * 3
}
```

Select one:

○ a. $O(N)$

◉ b. $O(lg(N))$ ✔

○ c. $O(N^3)$

○ d. $O(3^N)$

Your answer is correct.

In each iteration, the loop variable i is multiplied by 3 until it reaches N. Number of multiplications will be approximately $log_3(N)$. Hence the time complexity is O(lg(N))

The correct answer is:
$O(lg(N))$

Question 4
Correct
Mark 1.00 out of 1.00

Which of the following case does **not** exist in complexity theory

- a. Best Case
- b. Null Case ✔
- c. Average Case
- d. Worst Case

The correct answer is: Null Case

Question 5
Correct
Mark 1.00 out of 1.00

Express the function $\frac{n^3}{1000} - 100n^2 - 100n + 3$ in terms of Θ-notation.

- a. $\Theta(n^2)$
- b. $\Theta(\sqrt{n})$
- c. $\Theta(n^3)$ ✔
- d. $\Theta(\lg(n))$

The correct answer is: $\Theta(n^3)$

Question 6
Correct
Mark 1.00 out of 1.00

The worst case complexity of Bubble sort algorithm is

- a. $O(n)$
- b. $O(\log(n))$
- c. $O(n\log(n))$
- d. $O(n^2)$ ✔

The correct answer is: $O(n^2)$

Question 7

Correct

Mark 1.00 out of 1.00

Let $f(n) = 7n + 8$ and $g(n) = n$, find c (a suitable constant) such that $O(g(n)) = f(n)$ for $n \geq n_0$

    a.   8 ✔

    b.   7

    c.   There is no such constant

The correct answer is: 8

Question 8

Correct

Mark 1.00 out of 1.00

Following is the execution time measurement taken for a sorting algorithm to sort an array with a random permutation of elements.

No. of elements in the array (N)	Execution time (micro seconds)
1024	51
2048	202
4096	805
8192	3227
16384	12900
32768	51592

What can be the possible average case time complexity of this sorting algorithm?

Select one:

    a. $O(N^2)$ ✔

    b. $O(N)$

    c. $O(Nlg(N))$

    d. $O(4N)$

Your answer is correct.

The correct answer is:
$O(N^2)$