
Attempt Score 20 out of 20 points

1. What is the big O for the code below

```
public int method3(int n) {
  int sum = 0;
  for (int j = 0; j < n; j++) {
    for (int k = 0; k < n; k++) {
      for (int l = 0; l < n; l++) {
        sum += j * k / (l + 1);
      }
    }
  }
  return sum;
}
```

- a. O(n)
- b. $O(n^2)$
- c. $O(n^3)$
- d. O(n log n)

QUESTION 2

1. What is the big O for the code below

```
public int method4(int n) {
  for (int i = 0; i < n; i++) {
   for (int j = n; j > 0; j=j/2) {
     sum += 1;
   }
  }
  return sum;
}
```

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

QUESTION 3

1. What is the big O for the code below

```
public int method1(int n) {

 if (n > 1) {
 return n;
 }
 else {
 return 0;
 }
}
```

- a. O(n)
- b. O(log n)
- c. O(1)
- d. O(n log n)

QUESTION 4

What is the big O notation of the following function:

$$n^{log2} + \log n^n + n \log n!$$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$n^{\log(2)} + \log(n^n) + n\log(n!)$

$$n^{\log(2)} + \log(n^n) + n\log(n!)$$

- ○ O(log n)
- ○ O(n)
- ○ O(n log n)
- ◉ O(n log n!)

QUESTION 5

1. What is the big O for the code below

```
public int method2(int n) {
 int sum = 0;
 for (int j = 0; j < n; j++) {
     sum += j;
  }
  return sum;
 }
```

a. O(log n)

b. O(n)

c. $O(n^2)$

d. O(1)

QUESTION 7

What is the big O notation of the following function:

$$n^2 + n \log n2^n$$

- $O(n^2 \log n)$
- $O(n^2)$
- $O(n \log n)$
- $O(2^n)$

For Blank 3

QUESTION 8

Fill in the blanks with the correct frequency for every line and the total big O:

```
1 int i, j, sum = 0;          [ 1 ]
2 for (i = 0; i < n; i++)     [ n+1 ]
3     if (i != 0)             [ n ]
4         sum += i;           [ n-1 ]
5     else for (j = 0; j < n; j++)   [ n+1 ]
6         sum += j;           [ n ]
```

O([n])

QUESTION 6

Fill in the blanks with the correct frequency in the corresponding line:

```
1 public int method5(int n) {
2       int res = 0;
3       for (int i = 5; i <= n + 3; i++) {   n
4             res += i;   n-1
5             for (int j = n - 3; j < n - 2; j++) {
   2n-2
6                   res += j;   n-1
7             }
8       }
9       return res;
10 }
```