Quiz Submissions - Algorithm Analysis Homework Quiz

Chork Hieng (username: gt9182iu)

Attempt 1

Written: Mar 17, 2022 11:19 PM - Mar 17, 2022 11:56 PM

Submission View

Your quiz has been submitted successfully.

Tilde approximations and orders of growth.

For each question in this section, provide the tilde approximation and the order of growth for the given function. Put the tilde approximation in the first blank, and the order of growth in the second.

Since D2L does not provide an equation editor for student answers, use `^` to designate exponentiation, and `lg N` to designate a logarithm to the base 2. So if the function is given as:

$$f(N) = 4N\ lg\ N + 3$$

Your answers should be

4N lg N

N lg N

Question 1 — 2 / 2 points

$$f(N) = \frac{N^2}{2}$$

Answer for blank # 1: (N^2)/2 ✗ **(N^2 / 2, N^2/2)**

Answer for blank # 2: N^2 ✓ (50 %)

Question 2 — 2 / 2 points

$$f(N) = \frac{N^2 + N}{2}$$

Answer for blank # 1: (N^2)/2 ✗ **(N^2 / 2, N^2/2)**

Answer for blank # 2: N^2 ✓ (50 %)

Question 3 **2 / 2 points**

$$f(N) = lgN + N + 2lgN$$

Answer for blank # 1: N ✓ (50 %)

Answer for blank # 2: N ✓ (50 %)

Question 4 **2 / 2 points**

$$f(N) = 3N^2 + NlgN$$

Answer for blank # 1: 3(N^2) ✗ **(3N^2, 3 N^2)**

Answer for blank # 2: N^2 ✓ (50 %)

Question 5 **2 / 2 points**

$$f(N) = N\ lg\ N + 5N^2$$

Answer for blank # 1: 5(N^2) ✗ **(5N^2, 5 N^2)**

Answer for blank # 2: N^2 ✓ (50 %)

Question 6 **2 / 2 points**

$$f(N) = 4N\ lg\ N + 2N$$

Answer for blank # 1: 4N lg N ✓ (50 %)

Answer for blank # 2: N lg N ✓ (50 %)

Question 7 **2 / 2 points**

$$f(N) = 4\ lg\ N + 2N$$

Answer for blank # 1: 2N ✓ (50 %)

Answer for blank # 2: N ✓ (50 %)

Question 8 **2 / 2 points**

$$f(N) = 4\ lg\ N$$

Answer for blank # 1: 4 lg N ✔ (50 %)

Answer for blank # 2: lg N ✔ (50 %)

Question 9 **2 / 2 points**

$$f(N) = 5$$

Answer for blank # 1: 5 ✔ (50 %)

Answer for blank # 2: 1 ✔ (50 %)

Question 10 **2 / 2 points**

$$f(N) = N^2 + 2N^2\ lg\ N$$

Answer for blank # 1: 2(N^2) lg N ✘ **(2 N^2 lg N, 2N^2 lgN, 2N^2lgN, 2 N^2 lgN)**

Answer for blank # 2: N^2 lg N ✔ (50 %)

Question 11 **0 / 2 points**

Extra credit:

$$f(N) = 2\ lg\ N^2$$

Answer for blank # 1: 2 lg N^2 ✘ **(4 lg N, 4lgN, 4 lgN)**

Answer for blank # 2: lg N^2 ✘ **(lg N, lgN)**

Question 12 **0 / 2 points**

Extra credit:

$$f(N) = \frac{lg\ N^2}{lg\ N}$$

Answer for blank # 1: (lg N^2)/(lg N) ✘ **(2)**

Answer for blank # 2: (lg N^2)/(lg N) ✘ **(1)**

Analyzing Algorithms

In this section, each question will contain an implementation of an algorithm you need to analyze to find the order of growth.

Assume that **`Arrays.sort()`** is an

$$N\ lg\ N$$

algorithm.

Question 13 — 2 / 2 points

```
public static double accumulate(double[] a) {
    double sum = 0.0;
    for (int i = 0; i < a.length; i++)
        sum += a[i];
    return sum;
}
```

- ○ 1
- ○ $lg\ N$
- ✓ ● N
- ○ $N\ lg\ N$
- ○ N^2
- ○ $N^2\ lg\ N$

Question 14 — 2 / 2 points

```
public static double innerProduct(double[] a, double[] b) {
// assume a.length == b.length
    double sum = 0;
    for (int i = 0; i < a.length; i++)
        sum += a[i] * b[i];
    return sum;
}
```

- ○ 1
- ○ $lg\ N$

- ✔ N
- $N \ lg \ N$
- N^2
- $N^2 \ lg \ N$

Question 15 — 2 / 2 points

```
public static int twoSum(int[] a) {
    int count = 0;
    for (int i = 0; i < a.length; i++)
        for (int j = i + 1; j < a.length; j++)
            if (a[i] + a[j] == 0)
                count += 1;
    return count;
}
```

- 1
- $lg \ N$
- N
- $N \ lg \ N$
- ✔ N^2
- $N^2 \ lg \ N$

Question 16 — 0 / 2 points

```
public static int twoSumFastUnique(Integer[] a) {
    Arrays.sort(a);
    int count = 0;
    for (int i = 0; i < a.length; i++) {
        int idx = ArrayUtils.upperBound(a, i+1, a.length, -a[i]);
```

```
            if (idx != a.length && a[idx] == -a[i])
                count++;
        }
        return count;
    }
```

- 1
- $lg\ N$
- N
- ➡ $N\ lg\ N$
- ✖ N^2
- $N^2\ lg\ N$

Question 17 — 0 / 2 points

```
public static int twoSumFast(Integer[] a) {
    Arrays.sort(a);
    int count = 0;
    for (int i = 0; i < a.length; i++) {
        int lidx = ArrayUtils.lowerBound(a, i+1, a.length, -a[i]);
        int uidx = ArrayUtils.upperBound(a, i+1, a.length, -a[i]);
        count += uidx - lidx;
    }
    return count;
}
```

- 1
- $lg\ N$
- N
- ➡ $N\ lg\ N$
- ✖

N^2

$N^2 \ lg \ N$

Question 18 — 0 / 2 points

```
public static int twoSumReallyFastUnique(int[] a) {
    Arrays.sort(a);
    int lo = 0;
    int hi = a.length-1;
    int count = 0;
    while (lo < hi) {
        int sum = a[lo] + a[hi];
        if (sum < 0)
            lo++;
        else if (sum > 0)
            --hi;
        else {
            count++;
            lo++;
            --hi;
        }
    }
    return count;
}
```

1

$lg \ N$

➡ N

✖ $N \ lg \ N$

N^2

$N^2 \ lg \ N$

Question 19 — 2 / 2 points

```
public static int threeSumFast(Integer[] a) {
    Arrays.sort(a);
    int count = 0;
```

```
    for (int i = 0; i < a.length; i++) {
        for (int j = i+1; j < a.length; j++) {
            int lidx = ArrayUtils.lowerBound(a, j+1, a.length, -a[i]);
            int uidx = ArrayUtils.upperBound(a, j+1, a.length, -a[i]);
            count += uidx - lidx;
        }
    }
    return count;
}
```

- 1
- $lg\ N$
- N
- $N\ lg\ N$
- N^2
- ✔ $N^2\ lg\ N$

Question 20 — 2 / 2 points

```
public static double accumulateSorted(double[] a) {
    Arrays.sort(a);
    double sum = 0.0;
    for (int i = a.length-1; i >= 0; i--)
        sum += a[i];
    return sum;
}
```

- 1
- $lg\ N$
- N
- ✔ $N\ lg\ N$

N^2

$N^2 \ lg \ N$

Question 21 — 2 / 2 points

```
public static int multiply(int a, int b) {
    int result = 0;
    while (b != 0) {
        if (b % 2 != 0)
            result += a;
        result += result;
        b /= 2;
    }
    return result;
}
```

1

✔ $lg \ N$

N

$N \ lg \ N$

N^2

$N^2 \ lg \ N$

Question 22 — 2 / 2 points

```
public static int power(int a, int b) {
    int result = 1;
    while (b != 0) {
        if (b % 2 != 0)
            result *= a;
        result *= result;
        b /= 2;
    }
    return result;
}
```

- ○ 1
- ✓ ● $lg\ N$
- ○ N
- ○ $N\ lg\ N$
- ○ N^2
- ○ $N^2\ lg\ N$

Attempt Score: 34 / 44 - 77.27 %

Overall Grade (highest attempt): 34 / 44 - 77.27 %

Done