

**ICS 202 – Data Structures and Algorithms**  
**Spring Semester 2024/2025 (242)**

**Assignment #1**

**Question I. (10 points)** Let Array  $A[1: 50] = 19, 21, 23, \dots, 115, 117$ . How many element comparisons are performed by Algorithm BINARYSEARCH when searching for the following values of  $x$ ? You need to show how you get your final answer. (a) 121 (b) 57 (c) 19 (d) 11

**Question II (15 points):** Consider the following algorithm, where the input  $n$  is a positive power of 3:

```
for (i = 0; i <= n; i += 3) {  
    for (j = 1; j <= n; j *= 3) {  
        for (k = 1; k <= i; k++) {  
            x = x + 1; // Statement 1  
        }  
    }  
}
```

1. (12 points) Determine the number of times **Statement 1** gets executed, showing all the steps clearly.
2. (3 points) Write the cost of the algorithm using Big  $\Theta()$  notation

**Question III (15 points):** Write the function  $T(n) = n^{1/3} \log^2(n^2) + \sqrt{n \log n}$  in terms of Big  $\Theta()$  notation. Prove your answer.

**Question IV (20 points):** Consider the following algorithm, where the input is two arrays of sizes  $n$  and  $m$ , respectively.

```
boolean arraysComparison(int[] arr1, int[] arr2){  
    int n = arr1.length;  
    int m = arr2.length;  
  
    for(int i = 0; i < n; i++){  
        for(int j = 0; j < m; j++){  
            if(arr1[i] > arr2[j] && arr1[i] - arr2[j] > 5)  
                return true;  
        }  
    }  
  
    return false;  
}
```

1. (6 points) What is the best case time complexity of this algorithm? Justify your answer, indicating which statement did you use to find the complexity.
2. (8 points) What is the worst case time complexity of this algorithm? Justify your answer, indicating which statement did you use to find the complexity.
3. (6 points) What is the space complexity of this algorithm? Justify your answer.

**Question V (10 points):** Determine whether the following statements are true or false. If true, prove your answer. If false, provide a counter example.

1. If  $f(n)$  is  $\Theta(g(n))$ , then  $e^{f(n)}$  is  $\Theta(e^{g(n)})$ , where  $e$  is a mathematical constant
2.  $f(n) + g(n)$  is  $\Omega(\max(f(n), g(n)))$  for all functions  $f(n)$  and  $g(n)$  where they are both positive and non-decreasing.
3.  $b^{na}$  is  $O(b^n)$ , where  $a, b$  are constants.
4.  $n^{\log n}$  is  $O(2^n)$ , where  $n$  is a positive integer, but only when  $n$  is sufficiently large.

**Definition:**  $f(n)$  is said to be in small  $o(g(n))$  if  $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0$ .

The small  $o()$  notation is usually used to determine whether two functions belong to two different complexity classes or not.

**Question VI (15 points):** Fill in the following blanks with either T for true or F for false:

$f(n)$	$g(n)$	$f = O(g)$	$f = \Omega(g)$	$f = \Theta(g)$	$f = o(g)$
$1000n^{3/2}$	$10 + n^2 + n$				
$\sqrt{n} \log^2 n$	$n \log n$				
$2^n$	$n^n$				
$n^n$	$3^{3n}$				

**Question VII (15 points):** Given the code segment below and that  $n$  and  $x$  are the input, where  $n$  is a power of 3, answer the following questions:

```
// . . .
int total = 0;

for (int i = 0; i < n * n; i++)
    total++;           // statement1

if (x < 3) {
    for (int i = 1; i <= n; i++)
        for (int j = 0; j < n - i; j++)
            total++;    // statement2
}
else {
    for (int i = 1; i <= n * n; i *= 5)
        total++;        // statement3
}

for (int i = 1; i <= n * n; i *= 4)
    for (int j = 1; j <= i / 2; j++)
        total++;        // statement4
```

- (13 points) Find the number of times `statement1`, `statement2`, `statement3` and `statement4` get executed, showing all the details of your solution, if
  - $x = 2$
  - $x = 5$
- (1 points) Determine the Big- $\Theta()$  complexity of this program fragment in the best case.
- (1 points) Determine the Big-  $\Theta()$  complexity of this program fragment in the worst case.

### IMPORTANT NOTE REGARDING THIS HOMEWORK SUBMISSION

This homework must be submitted to GRADESCOPE. Your homework submission must be a pdf file. If you do not type your homework and just solve by hand, make sure you clearly scan the pages (using a scanner or a mobile scanning software like Office Lens or Cam Scanner) of your homework and convert it to **A SINGLE pdf file** or using the GRADESCOPE app directly on iOS/Android. Make sure to map each answer to the corresponding question. Failure to do the mapping will result in losing 10 points.