

## 1 Asymptotics

Select **all valid** bounds for each implementation of `f0` below. Note that `Math.random` returns a random number between 0 and 1.

(a) 

```
public void f0(int n) {  
    if (n < 100) {  
        return;  
    }  
    f0(n - 2);  
}
```

- |  |  |  |
|--|--|--|
| <input type="checkbox"/> $\Omega(1)$               | <input type="checkbox"/> $\Theta(1)$               | <input type="checkbox"/> $O(1)$                      |
| <input type="checkbox"/> $\Omega(\log(n))$         | <input type="checkbox"/> $\Theta(\log(n))$         | <input type="checkbox"/> $O(\log(n))$                |
| <input type="checkbox"/> $\Omega(n)$               | <input type="checkbox"/> $\Theta(n)$               | <input type="checkbox"/> $O(n)$                      |
| <input type="checkbox"/> $\Omega(n \cdot \log(n))$ | <input type="checkbox"/> $\Theta(n \cdot \log(n))$ | <input type="checkbox"/> $O(n \cdot \log(n))$        |
| <input type="checkbox"/> $\Omega(n^2)$             | <input type="checkbox"/> $\Theta(n^2)$             | <input type="checkbox"/> $O(n^2)$                    |
| <input type="checkbox"/> $\Omega(n^3)$             | <input type="checkbox"/> $\Theta(n^3)$             | <input type="checkbox"/> $O(n^3)$                    |
| <input type="checkbox"/> $\Omega(3^n)$             | <input type="checkbox"/> $\Theta(3^n)$             | <input type="checkbox"/> The function never finishes |
| <input type="checkbox"/> $\Omega(n!)$              | <input type="checkbox"/> $\Theta(n!)$              |  |

```
(b) public void f0(int n) {
    if (n < 100) {
        return;
    }
    if (Math.random() < 0.5) {
        return;
    }
    f0(n / 2);
}
```

- |  |  |  |
|--|--|--|
| <input type="checkbox"/> $\Omega(1)$               | <input type="checkbox"/> $\Theta(1)$               | <input type="checkbox"/> $O(1)$                      |
| <input type="checkbox"/> $\Omega(\log(n))$         | <input type="checkbox"/> $\Theta(\log(n))$         | <input type="checkbox"/> $O(\log(n))$                |
| <input type="checkbox"/> $\Omega(n)$               | <input type="checkbox"/> $\Theta(n)$               | <input type="checkbox"/> $O(n)$                      |
| <input type="checkbox"/> $\Omega(n \cdot \log(n))$ | <input type="checkbox"/> $\Theta(n \cdot \log(n))$ | <input type="checkbox"/> $O(n \cdot \log(n))$        |
| <input type="checkbox"/> $\Omega(n^2)$             | <input type="checkbox"/> $\Theta(n^2)$             | <input type="checkbox"/> $O(n^2)$                    |
| <input type="checkbox"/> $\Omega(n^3)$             | <input type="checkbox"/> $\Theta(n^3)$             | <input type="checkbox"/> $O(n^3)$                    |
| <input type="checkbox"/> $\Omega(3^n)$             | <input type="checkbox"/> $\Theta(3^n)$             | <input type="checkbox"/> The function never finishes |
| <input type="checkbox"/> $\Omega(n!)$              | <input type="checkbox"/> $\Theta(n!)$              |  |

## 2 Best and Worst Case

What is the best and worst case runtimes of the following definitions of `f0` in  $\Theta(\cdot)$  notation?

(a) 

```
public int f0(int[] arr) {
    for(int i = 0; i < arr.length; i++) {
        if (arr[i] == 49) {
            return i;
        }
    }
    return -1;
}
```

Best Case:  $\Theta(\quad)$ , Worst Case:  $\Theta(\quad)$

(b) Note that we are trying to find the runtime of `f0`.

```
public int f0(int n) {
    if (n == 0) {
        return 0;
    }
    bloop(n);
    return f0(n / 3) + f0(n / 3) + f0(n / 3);
}
```

```
public int bloop(int n) {
    for (int i = 0; i < n; i += 1) {
        System.out.println("Ah, loops too");
    }
    return n;
}
```

Best Case:  $\Theta(\quad)$ , Worst Case:  $\Theta(\quad)$