**2.1)**

Both have the same growth rates.

**2.2)** A is True.

**2.6 A)**

**B)** O (log log (D))

**2.7)**

**1: A)** T(N) = O(N)

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | 0 | 0 |
| 1000 | 0 | 0 |
| 10,000 | .001 | 10 |
| 100,000 | .001 | 100 |

**C)** RatioshouldIncrease by which seems it does.

**2: A)** T(N) = O()

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | 0 | 0 |
| 1000 | .008 | 8 |
| 10,000 | .572 | 5720 |
| 100,000 | 48.15 | 481500 |

**C)** RatioshouldIncrease by but I don’t think it does.

**3: A)** T(N) = O()

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | .007 | .7 |
| 1000 | 7.576 | 7576 |
| 10,000 | Took too long | n/a |
| 100,000 | Took too long | n/a |

**C)** Should increase by

**4: A)** T(N) = O()

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | 0 | 0 |
| 1000 | 0.003 | 3 |
| 10,000 | .252 | 2520 |
| 100,000 | 22.287 | 2228700 |

**C)** Should increase by

**5: A)** T(N) = O()

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | 4.538 | 453.8 |
| 1000 | Took too long | n/a |
| 10,000 | Took too long | n/a |
| 100,000 | Took too long | n/a |

**C)** RatioshouldIncrease by which seems it does.

**6: A)** T(N) = O()

**B)**

|  |  |  |
| --- | --- | --- |
| N | Time in Seconds | Ratio |
| 10 | 0 | 0 |
| 100 | .064 | 64 |
| 1000 | n/a | n/a |
| 10,000 | n/a | n/a |
| 100,000 | n/a | n/a |

**C)** RatioshouldIncrease by which seems it does.

**2.11 A)**

**B)**

**C)**

**D)**

**2.13 A)**

**B)** O log(N!)

**2.15)** The most efficient Algorithm to determine if an integer i exist in an array of integers is the Binary Search. The Binary Search every time cuts it in half resulting in Olog(N)

**2.16)** Turned in cpp file for this one.