CJ Young – cty0008

Shanti Upaydhay – spu0004

Dr. Yilmaz - COMP 3270 – Introduction to Algorithms

**Empirical Programming Analysis**

**Algorithm-1**

|  |  |  |
| --- | --- | --- |
| Step | Cost of each execution | Total # of times executed |
| 1 | 1 | 1 |
| 2 | 1 | N+1 |
| 3 | 1 |  |
| 4 | 1 |  |
| 5 | 1 |  |
| 6 | 6 |  |
| 7 | 6 |  |
| 8 | 2 | 1 |

Multiply col.1 with col.2, add across rows and simplify

*We made the assumption that the Math.max(a,b) function has a cost of 6. We determined this by:*

1. *Read a*
2. *Read b*
3. *Compare the values (Cost is determined to be 2 from L5-Complexity, slide 19)*
4. *Return the result of the comparison*
5. *Make the assignment*

Multiply col.1 with col.2, add across rows and simplify

T1(n) =

Text, letter

Description automatically generated

**Algorithm-2**

|  |  |  |
| --- | --- | --- |
| Step | Cost of each execution | Total # of times executed |
| 1 | 1 | 1 |
| 2 | 1 | N + 1 |
| 3 | 1 | N |
| 4 | 1 |  |
| 5 | 6 |  |
| 6 | 6 |  |
| 7 | 2 | 1 |

Multiply col.1 with col.2, add across rows and simplify

T2(n) =

Text, letter

Description automatically generated

**Algorithm-3**

|  |  |  |
| --- | --- | --- |
| Step | Cost of each execution | Total # of times executed in any single recursive call |
| 1 | 4 | 1 |
| 2 | 11 | 1 |
| Steps executed when the input is a base case: **Step 1 or Step 2** | | |
| First recurrence relation: T(n=1 or n=0) = **T(0) = 4 and also T(1) = 11** | | |
| 3 | 5 | 1 |
| 4 | 2 | 1 |
| 5 | 1 | (n/2) + 1 |
| 6 | 6 | n/2 |
| 7 | 6 | n/2 |
| 8 | 2 | 1 |
| 9 | 1 | (n/2) +1 |
| 10 | 6 | n/2 |
| 11 | 6 | n/2 |
| 12 | 4 | 1 |
| 13 | 4 | (cost excluding the recursive call):1 |
| 14 | 5 | (cost excluding the recursive call): 1 |
| 15 | 18 | 1 |
| Steps executed when input is NOT a base case: **Steps 1-15** | | |
| Second recurrence relation: T(n>1) = **13n + 48** | | |
| Simplified second recurrence relation (ignore the constant term): T(n>1) = **13n** | | |

***We determined the max(x,y,z) by deducing our own understanding of the function as follows:***

***If a >= b and a>=c***

***Max = a***

***If b >=a and b>=c***

***Max = b***

***Else***

***Max = c***

***Return max***

***We determined that max(x,y,z) to be a cost of 10. The deduction is as follows:***

1. ***Read a***
2. ***Read b***
3. ***Read c***

***Comparison for a:***

1. ***Compare a to b (cost of 2)***
2. ***Compare a to c (cost of 2)***
3. ***Check if both are true (and operator = cost of 1)***
4. ***Set a to max***

***Comparison for b:***

1. ***Compare b to a (cost of 2)***
2. ***Compare b to c (cost of 2)***
3. ***Check if both are true (and operator = cost of 1)***
4. ***Set b to max***

***Comparison for c:***

1. ***Set c to max***

***4. Return max***

***Total cost = 1 + 1 + 1 + (2 + 2 + 1 + 1) + (2 + 2 + 1 + 1) + (1) + 1***

***= 3 + (6) + (6) + (1) + 1***

***= 4 + 13***

***= 17***

Solve the two recurrence relations using any method (recommended method is the Recursion Tree). Show your work below:

T3(n) =

Chart

Description automatically generated

**Algorithm-4**

|  |  |  |
| --- | --- | --- |
| Step | Cost of each execution | Total # of times executed |
| 1 | 1 | 1 |
| 2 | 1 | 1 |
| 3 | 1 | N+1 |
| 4 | 8 | N |
| 5 | 6 | N |
| 6 | 2 | 1 |

Multiply col.1 with col.2, add across rows and simplify

T4(n) =

Text, letter

Description automatically generated