CSC 4102 - Analysis of Algorithms and Computing

Table of Algorithms alongwith their Time and Space complexities (Step count, variable count and asymptotic notations)

Sr.No	Algorithm	Big - Ω (Best case)	Big - Θ (Average Case)	Big - O (Worst case)	Space Complexity	Worst Case Step count	Best Case Step count
1.	Selection sort	$\Omega(n^2)$	$\Theta(n^2)$	$O(n^2)$	$n+3$	$3n^2+5n$	$2n^2+3n$
2.	Heap sort	**BuildMaxHeap:** $\Omega(log_2(n))$. **MaxHeapify:** $\Omega(1)$	**BuildMaxHeap:** $\Theta(nlog_2(n))$. **MaxHeapify:** $\Theta(log_2 n)$	**BuildMaxHeap:** $O(nlog_2(n))$. **MaxHeapify:** $O(log_2 n)$	?	?	?
3.	Insertion sort	$\Omega(n)$	$\Theta(n^2)$	$O(n^2)$	$n+4$	$n(n-1)/2$	$n-1$
4.	Count sort	$\Omega(n)$	$\Theta(n)$	$O(n)$	$2n+k+2$	$4k+5n+2$ where k is the maximum element in the array and n is the size of the array	Same as worst case
5.	Magic square	?	?	?	?	?	?
6.	Ternary search	$\Omega(1)$	$\Theta(log_3 n)$	$O(log_3 n)$	$(n+5)(log_3 n+1)$	$5+log_3(n)c$	5
7.	Binary Search	$\theta(1)$	$\Theta(log_2 n)$	$O(log_2 n)$	$(n+4)(log_2 n+1)$	$4+log_2(n)c$	4
8.	Fibonacci	$\Omega(n)$	$\Theta(n)$	$O(n)$	5. Recursive Fibonacci = $4n-6$	$5n-6$. Recursive Fibonacci = ?	Same as worst case
9.	Factorial	$\Omega(n)$	$\Theta(n)$	$O(n)$	3. Recursive factorial = $2(n-1)$	$2n-1$. Recursive Factorial: $2n-2$	Same as worst case
10.	Maximum	$\Omega(n)$	$\Theta(n)$	$O(n)$	$n+3$	$3n$	$2n+1$
11.	Matrix addition	$\Omega(n^2)$	$\Theta(n^2)$	$O(n^2)$	For square matrix: $3n^2+3$. For rectangle matrix: $3mn+4$	For square matrix : $2n^2+2n+2$. For rectangle matrix : $2mn+2m+2$. Where m is # of rows and n is # of columns	Same as worst case
12.	Matrix multiplication	$\Omega(n^3)$	$\Theta(n^3)$	$O(n^3)$	For square matrix: $3n^2+4$. For rectangle matrix: $mn+mo+on+6$ Where m is # of rows in matA, n is # of columns in matB and o is # of columns in matA (and rows in matB).	For square matrix : $2n^3+3n^2+2n+2$ .For rectangle matrix : $2mno+3mn+2m+2$ Where m is # of rows in matA, n is # of columns in matB and o is # of columns in matA (and rows in matB).	Same as worst case
13.	Power	$\Omega(n)$	$\Theta(n)$	$O(n)$. Using Exponentiation by squaring : $O(log_2 n)$	3	$2n+1$. Recursive Power = $log_2(n)$	1
14.	isPrime	$\Omega(n)$	$\Theta(n)$	$O(n)$	3	$2n+1$	1
15.	Sum	$\Omega(n)$	$\Theta(n)$	$O(n)$	$n+3$	$2n+3$. Recursive Sum: $2n+2$	Same as worst case
16.	Tower of Hanoi	$\Omega(2^n)$	$\Theta(2^n)$	$O(2^n)$	$8n-7$	2^n-1	Same as worst case
17.	Sum of squares Equivalent	$\Omega(n)$	$\Theta(n)$	$O(n)$	4	no of digits in n + 7	7
18.	Sum of divisors	$\Omega(n)$	$\Theta(n)$	$O(n)$	$2n+3$	$n+5$	$(n+12)/2$
19.	Merge sort	$\Omega(nlog_2 n)$	$\Theta(nlog_2 n)$	$O(nlog_2 n)$		?	?
20.	Quick sort						