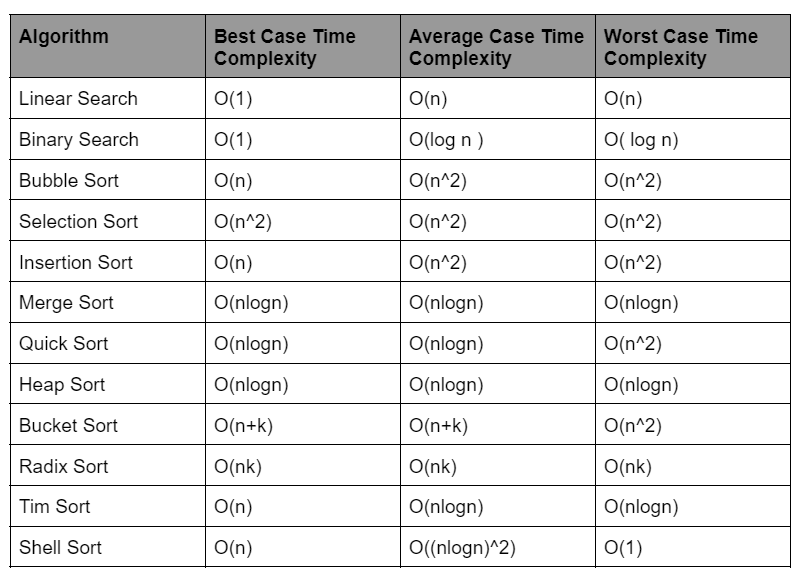
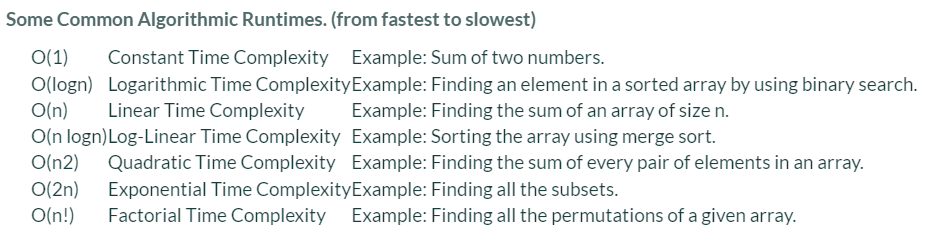
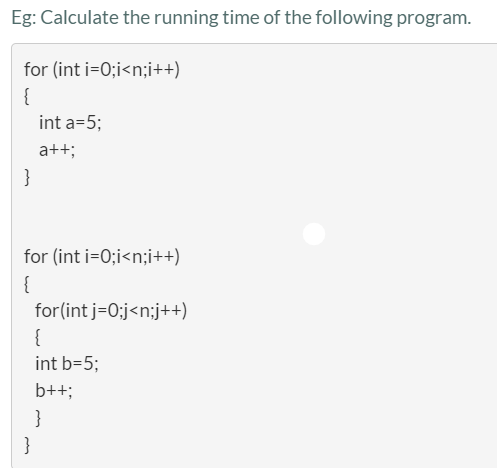
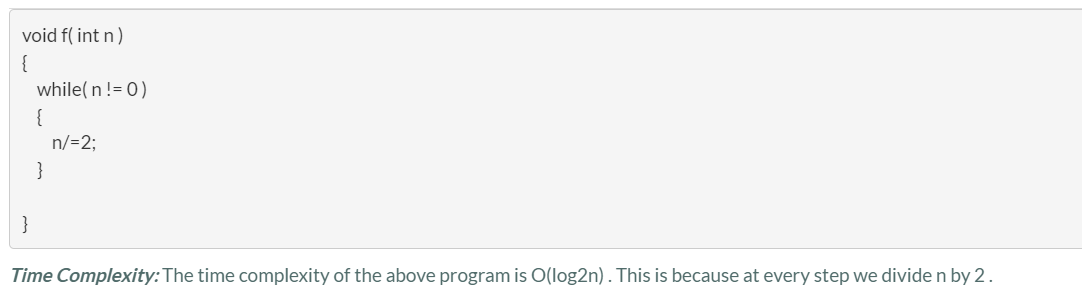
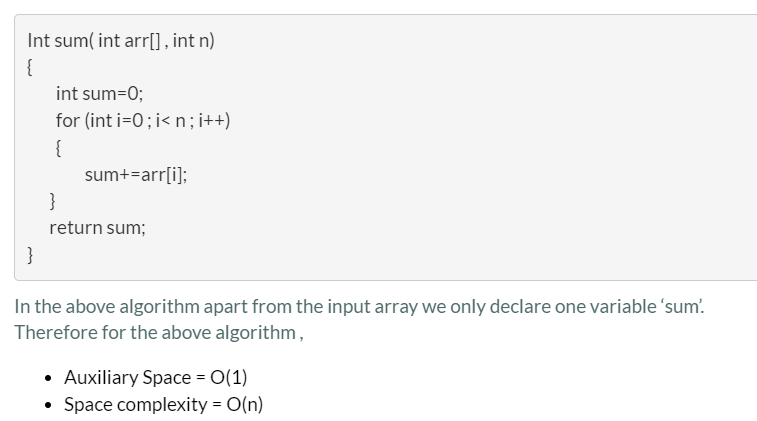
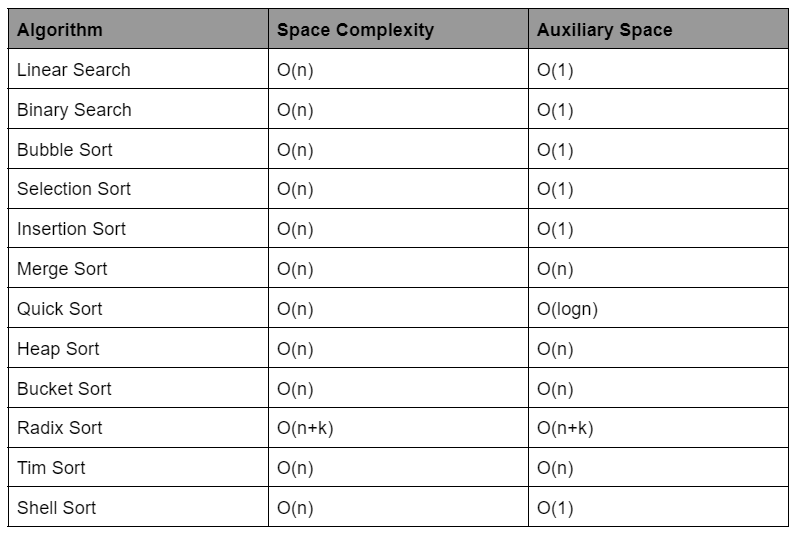
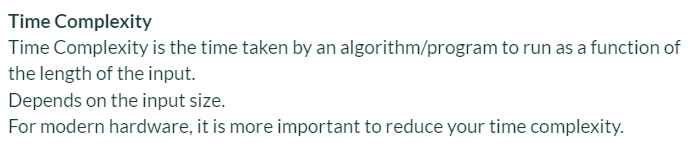
**TIME COMPLEXITY**

* 
* ****
* ****

the time complexity of the first fragment would be O(n), as the loop would run n times, the time complexity of the statements inside the loop is O(1).  
Whereas the time complexity of the second fragment would be O(n2).  
  
Time complexity of the program= O(n) + O(n2)= O(n2)

* 

**SPACE COMPLEXITY**

* vector<int> V;  
  for (int i = 0; i < N; i++) V.push\_back(  
    
  The code snippet ends up creating a vector of size N. So, space complexity of the code is O(n)
* **Auxiliary space** is the extra space used by the algorithm apart from the input space.
* 
* 
*   
   **while**    
  