

ASSIGNMENTS

Course: Data Structure and Algorithms - Lecturer: Assoc. Prof. Trần Thị Thanh Hải

N0	Topics	Requirement
1	Introduction to DS &A	<ol style="list-style-type: none"> 1) Make a flowchart to show how to solve quadratic equations: $ax^2 + bx + c = 0$. 2) Given a array of N integers, make a flowchart to print all even numbers. 3) Make a flowchart to compute the sum of all odd number 4) Make a flowchart to check if a number is a prime
2	Algorithm complexity	<ol style="list-style-type: none"> 1) Compute the number of typical operations of this code ?What is the complexity of this algo <pre> int a = 0; for (i = 0; i < N; i++) { for (j = N; j > i; j--) { a = a + i + j; } } </pre> 2) Prove that $T(n) = a_0 + a_1n + a_2n^2 + a_3n^3$ is $O(n^3)$ using the formal definition of the Big-O notation 3) Determine dominants term and O in the following table

Expression	Dominant term(s)	$O(\dots)$
$5 + 0.001n^3 + 0.025n$		
$500n + 100n^{1.5} + 50n \log_{10} n$		
$0.3n + 5n^{1.5} + 2.5 \cdot n^{1.75}$		
$n^2 \log_2 n + n(\log_2 n)^2$		
$n \log_3 n + n \log_2 n$		
$3 \log_8 n + \log_2 \log_2 \log_2 n$		
$100n + 0.01n^2$		
$0.01n + 100n^2$		
$2n + n^{0.5} + 0.5n^{1.25}$		
$0.01n \log_2 n + n(\log_2 n)^2$		
$100n \log_3 n + n^3 + 100n$		
$0.003 \log_4 n + \log_2 \log_2 n$		

4) Fill the following table

		<table> <tr> <th>Statement</th><th>Is it TRUE or FALSE?</th><th>If it is FALSE then write the correct formula</th></tr> <tr> <td>Rule of sums: $O(f + g) = O(f) + O(g)$</td><td></td><td></td></tr> <tr> <td>Rule of products: $O(f \cdot g) = O(f) \cdot O(g)$</td><td></td><td></td></tr> <tr> <td>Transitivity: if $g = O(f)$ and $h = O(f)$ then $g = O(h)$</td><td></td><td></td></tr> <tr> <td>$5n + 8n^2 + 100n^3 = O(n^4)$</td><td></td><td></td></tr> <tr> <td>$5n + 8n^2 + 100n^3 = O(n^2 \log n)$</td><td></td><td></td></tr> </table>	Statement	Is it TRUE or FALSE?	If it is FALSE then write the correct formula	Rule of sums: $O(f + g) = O(f) + O(g)$			Rule of products: $O(f \cdot g) = O(f) \cdot O(g)$			Transitivity: if $g = O(f)$ and $h = O(f)$ then $g = O(h)$			$5n + 8n^2 + 100n^3 = O(n^4)$			$5n + 8n^2 + 100n^3 = O(n^2 \log n)$		
Statement	Is it TRUE or FALSE?	If it is FALSE then write the correct formula																		
Rule of sums: $O(f + g) = O(f) + O(g)$																				
Rule of products: $O(f \cdot g) = O(f) \cdot O(g)$																				
Transitivity: if $g = O(f)$ and $h = O(f)$ then $g = O(h)$																				
$5n + 8n^2 + 100n^3 = O(n^4)$																				
$5n + 8n^2 + 100n^3 = O(n^2 \log n)$																				
3	Array, Pointer	<ol style="list-style-type: none"> 1) Compute Gradient of an image 2) Reverse a string using stack data structure 3) Generate binary numbers between 1 to n using a queue 4) Note: For the first exercise, you can use lib as OpenCV for reading and writing an image. To compute Gradient of an image, please study what is the gradient of image and implement the filter to compute the Gradient according to x and y axis. Gradient of an image: https://en.wikipedia.org/wiki/Image_gradient Convolution: https://en.wikipedia.org/wiki/Kernel_(image_processing)#Convolution 																		
4	Linked list, Queue, Stack	<ol style="list-style-type: none"> 1) Declare a student structure with main information (name, dob, array of courses). Each course is a structure with name, code, score 2) Define a student list as single linked list 3) Initialize the list as empty 4) Create a node student 5) Insert a student at the head of the list 6) Insert a student at the tail of the list 7) Insert a student according to the alphabetical order 																		

		8) Search the student having highest average score. 9) Count the number of students in the list 10) Print out the list of students
5	Recursive Algorithm	1) Recursively find the greatest common divisor of a sequence of values 2) Find the sum of a sequence of numbers recursively 3) Write a recursive function that determines whether an array is a palindrome, where the array and its size are given as parameters. 4) Write a program in C to find the sum of digits of a number using recursion
6	Tree	1) Define Node structure for a binary search tree 2) Write functions to <ul style="list-style-type: none"> - Insert a node into the tree (BST) - Delete the node from the tree (BST) - Search a given node in the tree - Traversal the tree 3) Study balance tree, AVL tree and prepare slides to present in the next lecture.
7	Graph	1) Create a Graph $G = (V, E)$, directed, weighted by your self 2) Define a structure to represent this graph 3) Implement one algorithm of graph traversal. 4) Run and report the result with the graph created in (1)
8	Sorting	1) Manually sort the elements 77, 49, 25, 12, 9, 33, 56, 81 using: insertion sort, bubble sort, selection sort, quick sort, merge sort and heap sort. Represent the temporal result for each step of sorting 2) Write functions to sort an array of integers using one basic sort algorithm and one advanced sort algorithm. Test with the above example
9	Searching	1 Calculate hash values of keys: 1892, 1921, 2007, 3456 using different methods of hashing 2) What is collision ? Explain various techniques to resolve a collision. Which technique do you think better and why ?

		<p>3) Consider a hash table with size = 10, using linear probing, insert the key 27, 72, 63, 42, 36, 18, 29, 101 into the table</p> <p>4) Implement a list of students, each student is a record with following information: ID (value of 4 digits), name. We would like to directly search the student according to his ID, if the number of students $N = 500$, and table size $M = 550$.</p> <ul style="list-style-type: none"> - Choose a suitable method (open addressing / chained hashing table) - Define list structure, hash table - Define function to store a student - Define a function to search a student given his ID
--	--	---