

ADS Questions Week1

Day 1 : Algorithms Datastructure & Recursion

Day 2 : Array , Types of Array ,Operations

Day 3: Time & Space Complexity, Searching Techniques, Stack

Day 4: Stack Applications

Day 1 : Algorithms Datastructure & Recursion

- 1) What do you mean by Data Structures?
- 2) Define The Goals Of Data Structure?
- 3) What is the Need of DS
- 4) List out the areas in which data structures are applied extensively(real time examples)?
- 5) List different types of data structures.
- 6) What Does Abstract Data Type Mean?
- 7) What is Recursion?
- 8) List and Explain types of Recursion
- 9) Explain the data structures used to perform recursion?
- 10)List the examples where recursion is used
- 11)Explain the difference between Recursion and Iteration, justify which to use when,Tail recursion?
- 12)Difference between Primitive and Non Primitive DS
- 13)Difference between Linear and Non Linear DS
- 14)What are different characteristics of an Algorithm?types of Algorithm.
- 15)State Advantages and Dis advantages of Recursion.

Lab Questions :

1. Write a program to print a series of numbers with recursive Java methods
2. Write a program to sum a series of numbers with Java recursion
3. Write a program to calculate a factorial in Java with recursion
4. Write a program to print the Fibonacci series with Java and recursion
5. Write a program to implement a recursive Java palindrome checker

Problem 1

Recursive program to find the Sum of the series $1 - 1/2 + 1/3 - 1/4 \dots 1/N$

Given a positive integer N, the task is to find the sum of the series $1 - (1/2) + (1/3) - (1/4) + \dots (1/N)$ using recursion.

Examples:

Input: N = 3

Output: 0.8333333333333333

Explanation:

$1 - (1/2) + (1/3) = 0.8333333333333333$

Input: N = 4

Output: 0.5833333333333333

Explanation:

$1 - (1/2) + (1/3) - (1/4) = 0.5833333333333333$

Recursive program to print formula for GCD of n integers

Given a function `gcd(a, b)` to find GCD (Greatest Common Divisor) of two number. It is also known that GCD of three elements can be found by `gcd(a, gcd(b, c))`, similarly for four element it can find the GCD by `gcd(a, gcd(b, gcd(c, d)))`. Given a positive integer `n`. The task is to print the formula to find the GCD of `n` integer using given `gcd()` function.

Examples:

Input : `n = 3`

Output : `gcd(int, gcd(int, int))`

Input : `n = 5`

Output : `gcd(int, gcd(int, gcd(int, gcd(int, int))))`

Problem 2

Recursive Program to print multiplication table of a number

Given a number `N`, the task is to print its multiplication table using recursion.

Examples

Input: `N = 5`

Output:

`5 * 1 = 5`

`5 * 2 = 10`

`5 * 3 = 15`

`5 * 4 = 20`

`5 * 5 = 25`

`5 * 6 = 30`

`5 * 7 = 35`

`5 * 8 = 40`

`5 * 9 = 45`

`5 * 10 = 50`

Input: `N = 8`

Output:

`8 * 1 = 8`

`8 * 2 = 16`

`8 * 3 = 24`

`8 * 4 = 32`

`8 * 5 = 40`

`8 * 6 = 48`

`8 * 7 = 56`

`8 * 8 = 64`

`8 * 9 = 72`

`8 * 10 = 80`

Day 2 : Array , Types of Array ,Operations

Q1 What is an Array?

Q2 How do you declare an Array in java?

Q3 What is the default value of Array for different data types?

Q4 Can you change size of Array in java after creation?

Q5 Can you pass the negative number in Array size?

Q6 Can you declare an Array without Array size?

Q7 Where does Array stored in JVM memory ?

Q8 Given a primitive Array in java, where does in JVM it is stored?

Q9 What is ArrayStoreException ? When this exception is thrown ?

Q10 What is the difference between ArrayStoreException and ArrayOutOfBoundsException ?

Q11 What are the advantages of Array ?

Q12 What are the disadvantages of Array?

Q13 Can we use Generics with an Array?

Q14 What is an Anonymous Array in Java ? Give example?

Q15 Write a program to print elements of Array ?

Q16 Write a program to sort an Array in Java ?

Q17 Write a program to check whether two given Arrays are equal, given both contains same data type and same length ?

Q18 Which is legal `int[] arr` or `int arr[]` ?

Q19 Write a program to throw `ArrayOutOfBoundsException`?

Q20 Write a program to throw `ArrayStoreException` ?

Q23 How to find the missing number in a given Array from number 1 to 100 ?

Q24 What are jagged arrays in java?,

Q25 There are two arrays object one containing 100 elements and another containing 50

elements. Both are of same data type. Can we assign one Array to another Array.

Q26 What are the different ways to copy one Array from another Array?

Q27 Write a program to search a specific element in an Array?

Q28 What will happen if you do not initialize an Array?

Q29 How to find duplicate elements in a given Array?

Q30 What are the different ways to traverse an Array in java?

Q31 Is this a legal way to define arrays `int[] arr = new int [4]{1, 2, 3, 4};`

Q32 What is two dimensional Array in java?

Q33 How do you declare a two dimensional Array in java?

Q34 Can we make Array volatile using volatile keyword?

Q35 Are Array thread-safe ?

Q36 What is the time complexity $O(n)$ of different operations of an Array?

Q37 Given two arrays, find the intersection between them? [\(solution\)](#)

Q38 Find the missing number in an Array between 1 to 100. Given only one number is missing. [\(solution\)](#)

Q39 Find out smallest and largest number in a given Array?

Questions on Stack

1. Explain stack? Which are the applications of stack?
2. State with help of diagrammatic representation push, pop operations, stack full and empty condition
3. List the type of expressions and with example describe process to convert a infix expression into postfix and then evaluate it. (Write code)
4. How can we implement stack and queue using linked list? (Can attempt later)
5. State various realtime applications of stack
6. WAP to create a stack and perform various operations like push, pop
7. WAP to create stack and perform reverse operation on String
8. WAP to create stack and implement Infix to Postfix or Prefix operation

- Given an expression string exp, write a program to examine whether the pairs and the orders of “{”, “}”, “(”, “)”, “[”, “]” are correct in exp.

- **Example:**

- **Input:** exp = “[()]”

- Output:** Not Balanced

- **Input:** exp = “[()]{}{[()()]()}"

- Output:** Balanced

- Create a data structure twoStacks that represents two stacks. Implementation of twoStacks should use only one array, i.e., both stacks should use the same array for storing elements. Following functions must be supported by twoStacks.
- - push1(int x) → pushes x to first stack
 - push2(int x) → pushes x to second stack
- - pop1() → pops an element from first stack and return the popped element
 - pop2() → pops an element from second stack and return the popped element
- Implementation of twoStack should be space efficient.

