

Questions

1.

You are given with an array arr which contains integer elements. Sort these elements in ascending order using insertion sort algorithm.

Example:

Input: 98, 23, 45, 14, 6, 67, 33, 42

Output: 6, 14, 23, 33, 45, 67, 98, 42

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd number

Run

Save

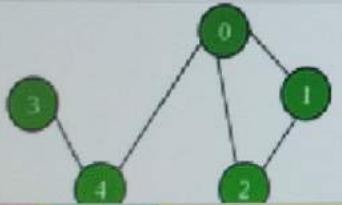
Your INPUT go's
like a=10

Questions

2.

You are given an undirected graph $G(V, E)$ with N vertices and M edges. We need to find the minimum number of edges between a given pair of vertices (u, v) .
Examples:

Input: For given graph G . Find minimum number of edges between $(1, 5)$.



Run

Save

1

Your INPUT go's here! Give only
like a=10

Sort Ascending

minimum edges

Fibonacci Sum

binary search

Sort Ascending

Palindrome or not

Frequency

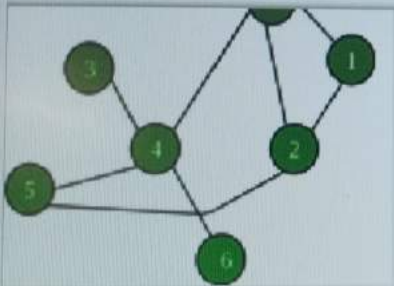
Missing Numbers

Linked List

bubble sort

odd nuber

Insert-delete



Output: 2
Explanation: (1, 2) and (2, 5) are the only edges resulting into shortest path between 1 and 5.

- | | |
|----------------|-------------------|
| Sort Ascending | minimum edges |
| Fibonacci Sum | binary search |
| Sort Ascending | Palindrome or not |
| Frequency | Missing Numbers |
| Linked List | bubble sort |
| odd number | Insert-delete |

Run Save

1

Your INPUT go's here! Give only like a=10

Questions

3.

Given the head of a singly linked list, return number of nodes present in a linked

Example 1:

1->2->3->5->8

Output 5

Run

Save

1

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd nuber

Your INPUT go's
like a=10

Questions

4.

Given a number n. the task is to print the Fibonacci series and the sum of the series using recursion.

input: n=10

output: Fibonacci series

0, 1, 1, 2, 3, 5, 8, 13, 21, 34

Sum: 88

Sort Ascending

> minimum

Fibonacci Sum

binary

Sort Ascending

Palindrome

Frequency

Missing

Linked List

bubble

odd number

insertion

Run**Save**

1

Your INPUT go's here! G
like a=10



Questions

5.

You are given an array arr in increasing order. Find the element x from arr using binary search.

Example 1: arr={ 1,5,6,7,9,10},X=6

Output : Element found at location 2

Example 2: arr={ 1,5,6,7,9,10},X=11

Output : Element not found at location 2

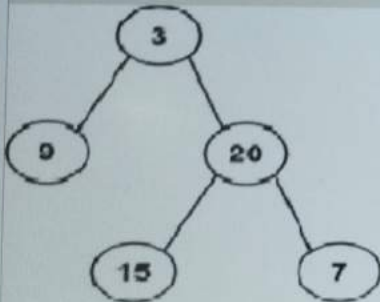
Run

Save

1

6.

Write a program to traverse the nodes present in the following tree in inorder and postorder traversal.



Run

Save

1

Your INPUT go's here!
like a=10

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd nuber

Questions

7.

Given a string s, sort it in ascending order and find the starting index of repeated character

Input: s = "tree"
Output: "eert", starting index 0

Input: s = "kkj"
Output: "jkk", starting index : 1

Example 2:
Input: s = "cccaaa"
Output: "aaaccc", starting index 0,3

Example 3:
Input: s = "Aabb"
Output: "bbAa", starting index 0,2

- | | |
|----------------|------------------|
| Sort Ascending | minimum edges |
| Fibonacci Sum | binary search |
| Sort Ascending | Palindrome or no |
| Frequency | Missing Number |
| Linked List | bubble sort |
| odd nuber | Insert-delete |

Run

Save

1

Your INPUT go's here! Give o
like a=10



Questions

8.

Given the head of a singly linked list, return true if it is a palindrome or false otherwise.

Example 1:

Input: head = [1,2,2,1]

Output: true

Example 2:

Input: head = [1,2]

Output: false

Input: R->A->D->A->R->NULL

Output: Yes

Input: C->O->D->E->NULL

Output: No

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd number

Run

Save

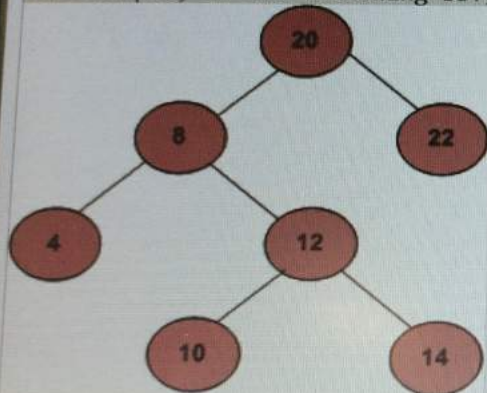
1

Your INPUT go
like a=10



9.

Given the root of a binary search tree and K as input, find Kth smallest element in BST.
For example, in the following BST,

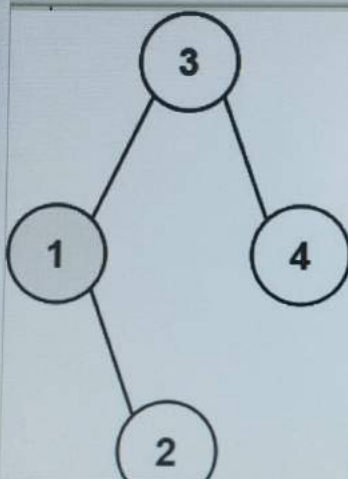


Run

Save

1

Your
like

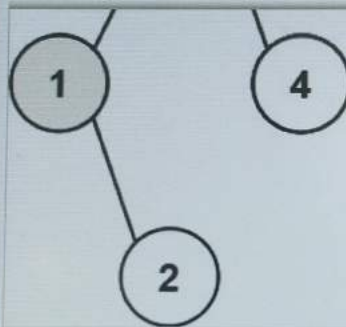


Run

Save

1





Input: root = [3,1,4,null,2], k = 1

Output: 1

Input: root = [5,3,6,2,4,null,null,1], k = 3

Output: 3

Run

Save

1

Questions

10.

Given a string s , find the frequency of characters

Example 1:

Input: $s = \text{"tree"}$

Output $t \rightarrow 1$, $r \rightarrow 1$, $e \rightarrow 2$

Run**Save**

1

**Questions**

11.

Given an unsorted array arr[] with both positive and negative elements, the task is to find the smallest positive number missing from the array.

Input: arr[] = {2, 3, 7, 6, 8, -1, -10, 15}

Output: 1

Input: arr[] = { 2, 3, -7, 6, 8, 1, -10, 15 }

Output: 4

Input: arr[] = {1, 1, 0, -1, -2}

Output: 2

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd nuber

Run

Save

1

Your INPUT go's
like a=10

Questions

12.

Given two integer arrays preorder and inorder where preorder is the preorder traversal of a binary tree and inorder is the inorder traversal of the same tree, construct and return the binary tree.

Input: preorder = [3,9,20,15,7], inorder = [9,3,15,20,7]

Output: [3,9,20,null,null,15,7]

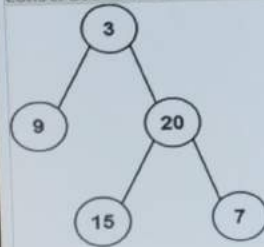
Run**Save**

1

Your INPUT goes here
like a=10

12.

Given two integer arrays preorder and inorder where preorder is the preorder traversal of a binary tree and inorder is the inorder traversal of the same tree, construct and return the binary tree.



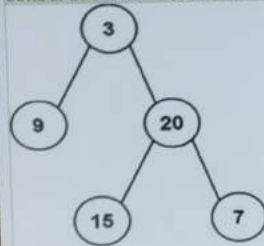
Run

Save

1

Your INPUT go's here
like a=10

Construct and return the binary tree.



Input: preorder = [3,9,20,15,7], inorder = [9,3,15,20,7]
Output: [3,9,20,null,null,15,7]

Run

Save

1

Sort Ascending

Fibonacci Sum

Sort Ascending

Frequency

Linked List

odd number

Your INPUT goes here
like a=10



Questions

13.

Write a program to create and display a linked list

Example 1:

Nodes : 6,7,8,9

Output: 6->7->8->9

Run

Save

1



SIMATS
ENGINEERING

Data Structure

Questions

14.

Write a program to sort the below numbers in descending order using bubble sort.

Input 4,7,9,1,2

Output:9,7,4,2,1

Run

Save

1

Sort Ascend

Fibonacci

Sort Ascend

Frequency

Linked

odd

Your ID

like a

Questions

15.

Given an array of size $N-1$ such that it only contains distinct integers in the range of 1 to N . Find the missing element.

Input:

 $N = 5$ $A[] = \{1, 2, 3, 5\}$

Output: 4

Input:

 $N = 10$ $A[] = \{6, 1, 2, 8, 3, 4, 7, 10, 5\}$

Output: 9

Run

Save

1

Questions

16.

Write a program to find odd number present in the data part of a node

Example Linked List 1->2->3->7

Output: 1,3,7

Run**Save**

1

Questions

17.

Write a program to perform insert and delete operations in a queue

Example : 12,34,56,78

After insertion of 60 content of the queue is 12,34,56,78,60

After deletion of 12 , the contents of the queue : 34,56,78,60

Run**Save**

1

Questions

18.

Given a string s containing just the characters '(', ')', '{', '}', '[' and ']', determine if the input string is valid. An input string is valid if:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.

Input: $s = "()"$

Output: true

Input: $s = "()[]{}"$

Output: true

Input: $s = "([])"$

Output: false

Input: $s = "([)]"$

Output: false

Input: $s = "{[]}"$

Output: true

Run

Save

1

Sort Ascend

Fibonacci

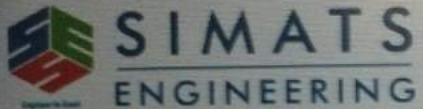
Sort Ascend

Frequent

Linked

odd

Your Input



Data Structure

19.

Given a number n , the task is to print the Fibonacci series and the sum of the series using Iterative procedure.

input $n=10$

output

Fibonacci series

0, 1, 1, 2, 3, 5, 8, 13, 21, 34

Sum: 88

Run

Save

20.

Given two strings needle and haystack, return the index of the first occurrence of needle in haystack, or -1 if needle is not part of haystack.

Example 1:

Input: haystack = "sadbutsad", needle = "sad"

Output: 0

Explanation: "sad" occurs at index 0 and 6.

The first occurrence is at index 0, so we return 0.

Input: haystack = "leetcode", needle = "leeto"

Output: -1

Explanation: "leeto" did not occur in "leetcode", so we return -1.

21.

Implement a first in first out (FIFO) queue using only two stacks. The implemented queue should support all the functions of a normal queue (push, peek, pop, and empty).

Implement the MyQueue class:

1. void push(int x) Pushes element x to the back of the queue.
2. int pop() Removes the element from the front of the queue and returns it.
3. int peek() Returns the element at the front of the queue.
4. boolean empty() Returns true if the queue is empty, false otherwise.

Input

```
["MyQueue", "push", "push", "peek", "pop", "empty"]
```

```
[[], [1], [2], [], [], []]
```

Output

```
[null, null, null, 1, 1, false]
```

Run

Save

Data Structure

22.

Given an array arr, sort the elements in descending order using bubblesort.

Arr=[9,10,-9,23,67,-90]

Output:[67,23,10,9,-9,-90]

Run

Save

odd nu

Fibonacci

descending

Min sta

singly link

Your INPUT
like a=10

23.

You have been given a positive integer N . You need to find and print the Factorial of this number without using recursion. The Factorial of a positive integer N refers to the product of all number in the range from 1 to N .

Input: $N=2$

Output: 2

Input: $N=4$

Output: 24

Run

Save

1

Your



Data Structure

24.

Given an array arr, sort the elements in ascending order using Bubble sort.

Arr=[9,10,-9,23,67,-90]

Output: [-90,-9,9,10,23,67]

Run

Save

1

Your INP
like a=1

Data Structure

Design a stack that supports push, pop, top, and retrieving the minimum element in constant time.

Implement the MinStack class:

1. MinStack() initializes the stack object.
2. void push(int val) pushes the element val onto the stack.
3. void pop() removes the element on the top of the stack.
4. int top() gets the top element of the stack.
5. int getMin() retrieves the minimum element in the stack.

Input

```
["MinStack","push","push","push","getMin","pop","top","getMin"]
```

```
[[],[-2],[0],[-3],[],[],[],[[]]]
```

Output

```
[null,null,null,null,-3,null,0,-2]
```

Explanation

```
MinStack minStack = new MinStack();
```

```
minStack.push(-2);
```

```
minStack.push(0);
```

Run**Save**

1

Your IN
like

5. int getMin() retrieves the minimum element in the stack.

Input

```
["MinStack","push","push","push","getMin","pop","top","getMin"]
```

```
[[[],[-2],[0],[-3],[[],[],[],[]]]]
```

Output

```
[null,null,null,null,-3,null,0,-2]
```

Explanation

```
MinStack minStack = new MinStack();
```

```
minStack.push(-2);
```

```
minStack.push(0);
```

```
minStack.push(-3);
```

```
minStack.getMin(); // return -3
```

```
minStack.pop();
```

```
minStack.top(); // return 0
```

```
minStack.getMin(); // return -2
```

Run**Save**

1

Your IN

26.

Find the factorial of a number using iterative procedure

Input : 3

Output: 6

Run

Save

1

odd n

Fibonac

descend

Min s

singly li

Your INPUT
like a=10

27.

Given the head of a linked list, insert the node in nth place and return its head.

Input: head = [1,3,2,3,4,5], p=3 n = 2

Output: [1,3,2,3,4,5]

Input: head = [1], p = 0, n = 1

Output: [0,1]

Input: head = [1,2], p=3, n = 3

Output: [1,2,3]

Run

Save

1

Your
like

28.

Given the head of a singly linked list and two integers left and right where $\text{left} \leq \text{right}$, reverse the nodes of the list between left and right (inclusive).

Input: head = [1, 2, 3, 4, 5], left = 2, right = 4

Output: [1, 4, 3, 2, 5]

Input: head = [5], left = 1, right = 1

Output: [5]

Input : [10,20,30,40,50,60,70], left = 3, right = 6

Output : [10,20,60,50,40,30,70]

Run

Save

1

Yo
li

Data Structure

and right where $\text{left} \leq \text{right}$, reverse the nodes of the list from position left to position right, and return the reversed list.

odd nub

Fibonacci S

descending

Min sta

singly link

Run

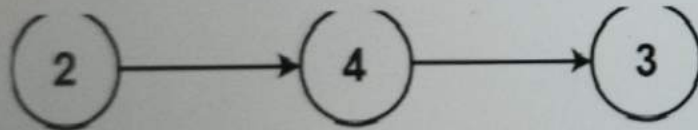
Save

Your INPUT
like a=10



29.

You are given with the following linked list

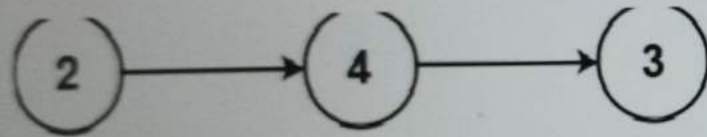


Run

Save

1

Your INPUT
like a=10



The digits are stored in the above order, you are asked to print the list in reverse order.

Run

Save

1

odd nu

Fibonacci

descending

Min st

singly link

Your INPUT
like a=10



Data Structure

30.

Given two sorted arrays nums1 and nums2 of size m and n respectively, return the sum of these two arrays

Example 1:

Input: nums1 = [1,3], nums2 = [2]

Output: 6

Example 2:

Input: nums1 = [1,2], nums2 = [3,4]

Output: 10

Run

Save

1

Your INPUT
like a=10