Bangabandhu Sheikh Mujibur Rahman Science And Technology University

Sheikh Hasina Institute of ICT

Assignment Name: CSE201 – Data Structure

Course Code: CSE201
Course Title: Data Structure
Date Of Submission: 10 / 10 / 2021

Submitted By	Submitted To
Name: Israt Jahan Reshma **ID:** 18ICTCSE041 **Year:** Second Year **Semester:** 1st Semester **Session:** 2018 – 19 Department Of Computer Science and Engineering, SHIICT, BSMRSTU	**Mr. Md. Jamal Uddin** Assistant Professor Department Of CSE BSMRSTU, Gopalgonj - 8100

1. What is data structure? What are the type of data structures? What are the notation of performance of an algorithm?

Ans: Data structure can be defined as the group of data elements which provides an efficient way of storing and organizing data in the computer so that it can be used efficiently. Example: Array, Linked list, Stack, queue etc.

2 types of Data structure

(i) Linear Data Structure:

A data structure is called linear if all elements of its are arranged in the linear order. In linear structure, the elements are stored non-hierarchical way where each element has the successor and predecessors except the 1st and last element. example: Array, Linked list, stack, queues etc.

(ii) Non Linear Data Structure:

This data structure does not form a sequence i.e. each item or elements are connected with two or more other item in a non linear arrangement. The data elements are not arranged in sequential structure. Example: Graph, Tree etc.

2nd Part: To measure the Performance of an algorithm, we generally used three asymptotic notation.

These are:

(i) Big Oh Notation (O)

(ii) Omega (Ω) Notation

(iii) Theta (Θ) Notation

(i) Big Oh Notation (O):

Def: $O(g(n)) = \{f(n)$: there exist positive constants c and n_0 such that $0 \leq f(n) \leq c\, g(n)$ for all $n \geq n_0\}$

(ii) Omega (Ω) Notation:

Def: $\Omega(g(n)) = \{f(n)$: there exist positive constants c and n_0 such that $0 \leq c\, g(n) \leq f(n)$ for all $n \geq n_0\}$

(iii) Theta (Θ) Notation:

Def: $\Theta(g(n)) = \{f(n)$: there exist positive constants c_1, c_2 and n_0 such that $0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$ for all $n \geq n_0\}$

2. What is Bubble sort? Write an algorithm for sorting an array of N elements using Bubble sort.
Sort this array using Bubble sort - 15, 18, 4, 5, 2

Ans: Bubble sort:

Bubble sort is a sorting algorithm that compares two adjacent elements and swaps them until they are not in the intended order.

Bubble Sort Algorithm:

1. Repeat step 2 to 3 for i=1 to N-1
2. Set J=1
3. Repeat while J ≤ N-i
   (a) If Data[J] > Data[j+1]
   the interchange Data[J] and Data[J+1]
   (b) set J = J+1
   [End of inner loop]
   [End of outer loop]
4. Exit.

2nd Part: Given Array 15, 18, 4, 5, 2
total 5 elements are here. So, the size of array is 5.
Let, Arr[5] = {15, 18, 4, 5, 2}
Now Discuss the Procedure of Bubble sort:

15 18 4 5 2

compare them. if 15 > 18 then swap else not.

15 18 4 5 2

compare them. since 18 > 4. swap them.

15 4 18 5 2

18 > 5 then swap them

15 4 5 18 2

18 > 2. swap them

15 4 5 2 18 [and finaly 18 is stored permanently at n-1 th position.

Again perform the overall operation among the 1st index to 3th index and then 2, 1 and so on.

15 4 5 2 18

4 15 5 2 18

4 5 15 2 18

4 5 2 15 18

4 5 2 15 18

4 2 5 15 18

2 4 5 15 18 and this is the sorted array.

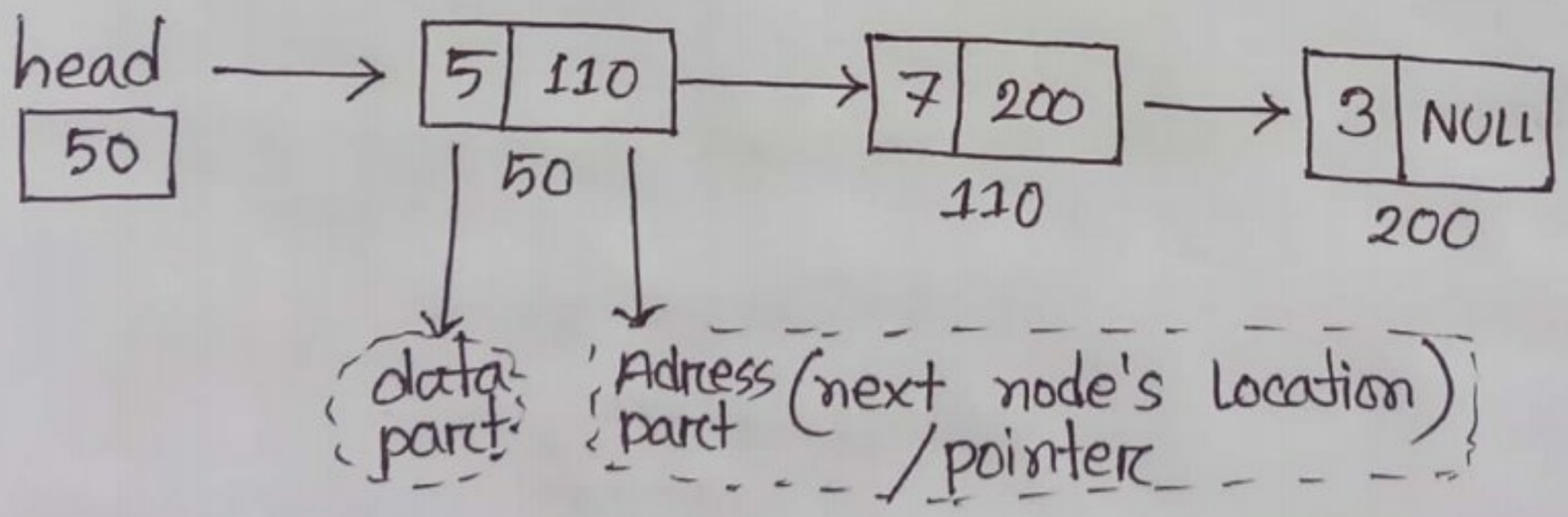
3. What are the type of linked list? Explain with diagram. Write an Algorithm to delete element (node) from "pos" position in the list.

Ans: Linked list is a linear data structure, in which the elements are not stored at contiguous memory location. The elements are linked using pointer.

Types of linked list:

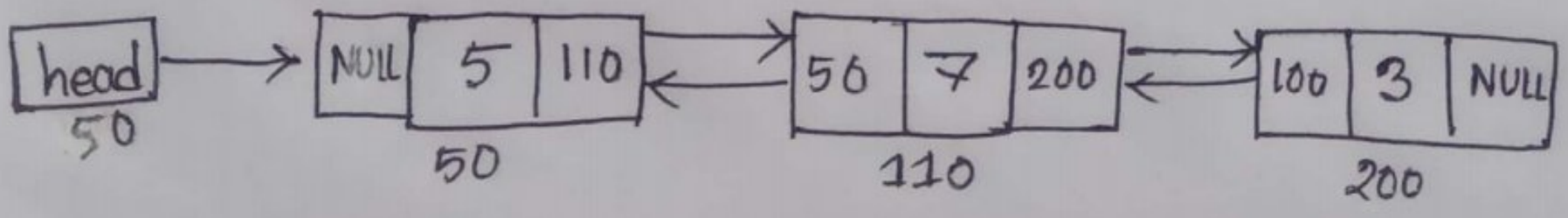
Singly linked list:

It is commonly used in program. The singly linked list contain two part. one is data part and another is adress part. which contain the adress of next node. The adress of next node is also known as pointer.

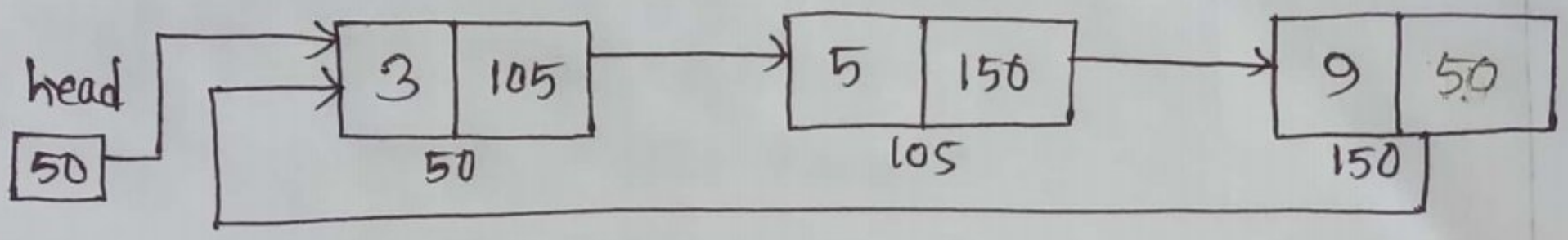


Doubly linked list:

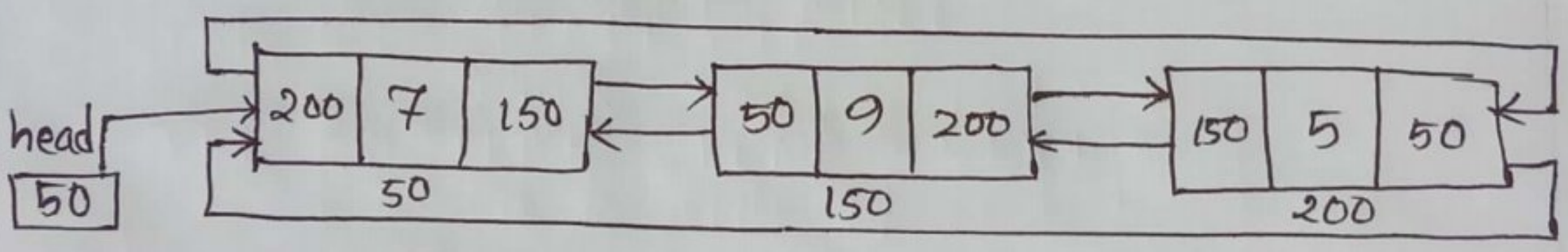
The doubly linked list contains two pointer. It has three part where one data part and two adress part which points to it's previous nodes and next nodes.



Circular linked list: It's a variation of singly linked list. The only difference is that the last node does connect to the first node. So the link part of the last node holds the first node's address.



Doubly Circular linked list: It's the combination of doubly linked list and doubly linked list.



2nd Part: An Algorithm to delete element from "pos" position:

STEP-1 : IF HEAD = NULL
Write underflow
Go to step 11
END of If

STEP-2 : SET TEMP = HEAD

STEP-3 : SET I = 0

STEP-4 : REPEAT STEP 5 TO 8
UNTIL I< POS

STEP-5 : TEMP1 = TEMP

STEP-6 : TEMP = TEMP→NEXT

STEP-7 : IF TEMP = NULL
Write "NOT Found"
GO to STEP 11
END of IF

STEP-8 : I=I+1
END OF LOOP

STEP-9 : TEMP1→NEXT = TEMP→NEXT

STEP-10 : FREE TEMP

STEP-11 : EXIT

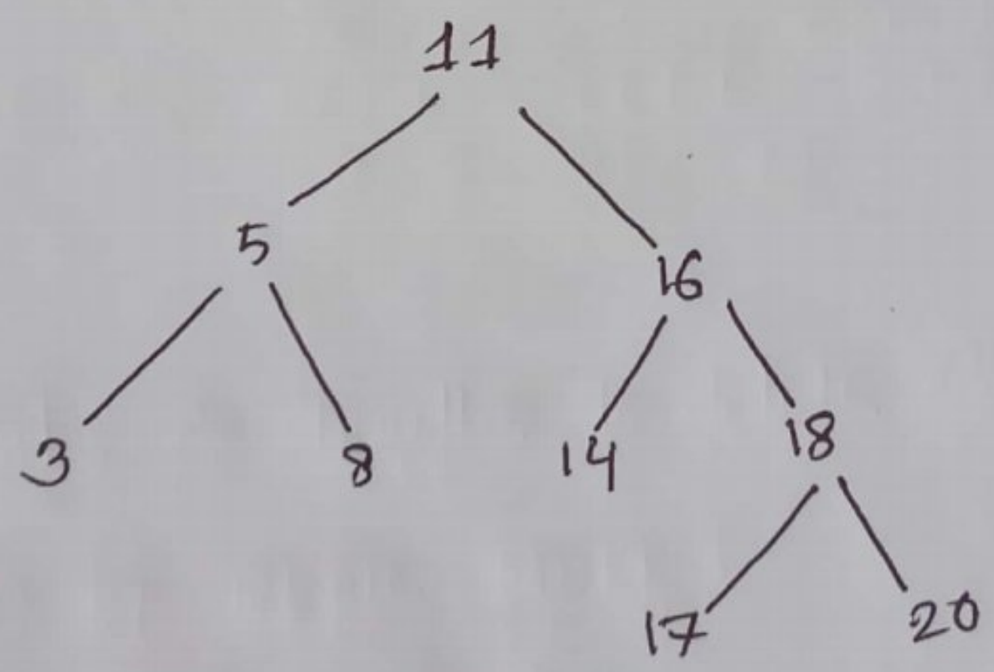
4. Define the Properties of complete Binary tree. Create a tree from given orders of traversal pre order =

{11, 5, 3, 8, 16, 14, 18, 17, 20}

in order = {3, 5, 8, 11, 14, 16, 17, 18, 20}

Ans: If T is any Binary tree, then T is called complete Binary tree if

(i) Each node of T has atmost 2 children.
(ii) All nodes at the last level as far left as possible.



5. Convert the infix equation to postfix using stack –

$(A + B \wedge D) / (E - F) * G$

Ans: Let Given equation, $P = (A + B \wedge D) / (E - F) + G$

Character Scanned	Stack	Postfix Expression
(	(	
A	(	A
+	(+	A
B	(+	AB
∧	(+∧	AB
D	(+∧	ABD
)		ABD∧+
/	/	ABD∧+
(	/(	ABD∧+
E	/(	ABD∧+E
–	/(–	ABD∧+E
F	/(–	ABD∧+EF
)	/	ABD∧+EF–
+	+	ABD∧+EF–/
G	+	ABD∧+EF–/G
		ABD∧+EF–/G+

6. What is the difference between sequential (linear) Search and Binary Search? Show the step of searching element '5' using Binary Search Technique in this array - 2, 5, 8, 15, 20.

Ans:

Discription: Linear Search is a technique where the searching the element sequentially until the element is found. But binary Search is a method that finds the middle element in the list recursively untill the middle element is matched with the searched element.

Sorted Data: In linear Search, the elements can be any order. but the precondition of binary search is that the array must be sorted.

Approach: linear search is a sequential approach but binary search is the divide and conquer approach.

Size: linear Search is preferable for small size and binary search fore large size.

Efficiency: Binary search is more efficient in the case of large array.

Implementation: linear search can be implemented on any linear data structure such as array, linked list, stack, etc. On the other hand, the implementation of binary search is limited as it can be implemented only on those data structures that have two-way traversal.

2nd Part: Searching '5' from the given array using binary search technique. Let, Array = { 2, 5, 8, 15, 20}

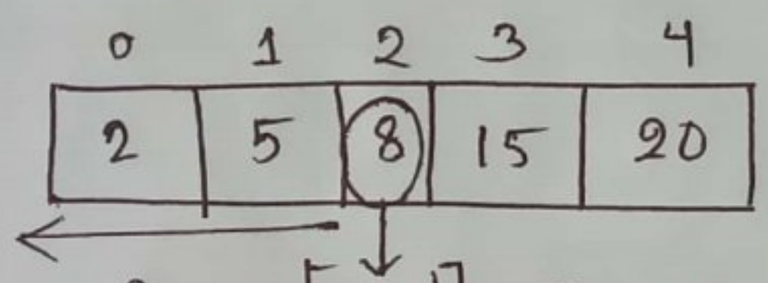
Applying divide and conquer, we will find 5. Let F = 5

So, 1st ly, divide the array and every time find the middle element and compare with the searched element. If

mid_element == searched element ; "Successfully find" return index.

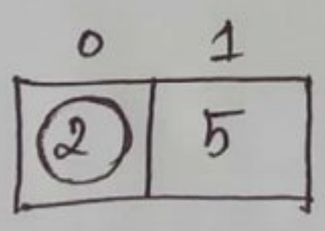
mid_element < searched element ; find the from left ,

else find from the right



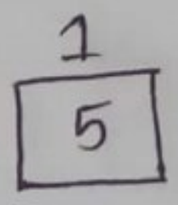
here, mid = (0+4)/2 = 2

Array [mid] > F so, searched element is within left elements.



here, mid = (0+1)/2 = 0

Array [mid] < F so, searched element is within right elements.



here, mid = (1+1)/2 = 1

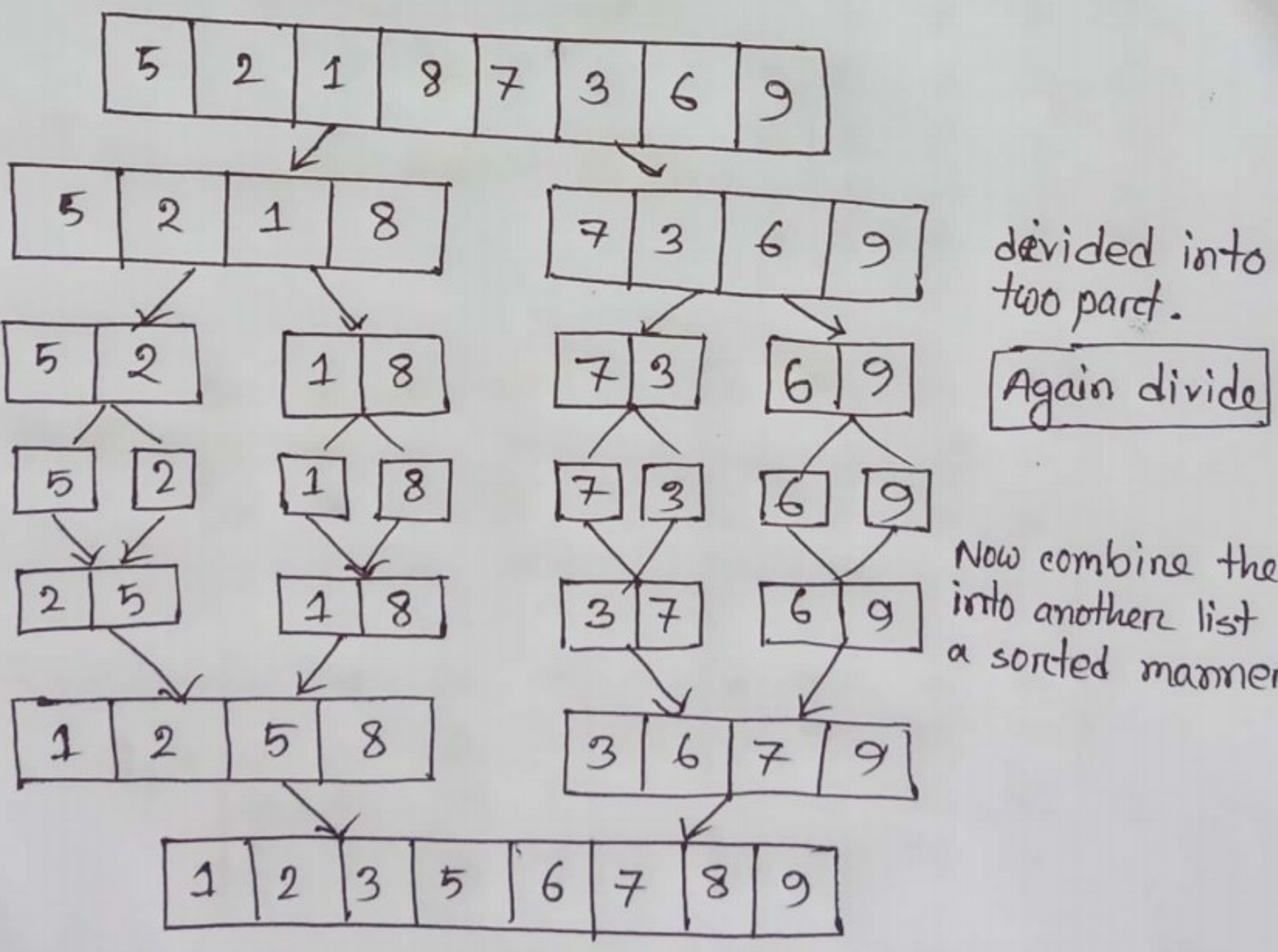
Array [mid] == F

∴ the index of 5 is 1.

7. Explain merge sort using an example.

Ans: Merge sort is a sorting technique based on devide and conquer technique. In merge sort, first devide the array into equals halves and then combine them in a sorted manner.

Let's A = { 5, 2, 1, 8, 7, 3 6, 9 } is an array. which is unsorted.



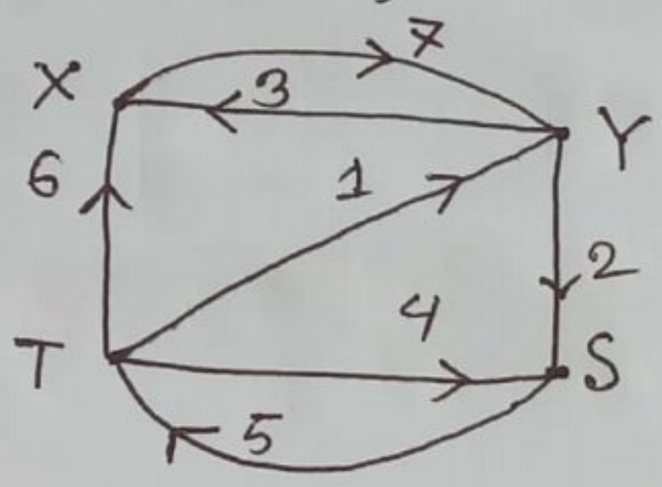
And this is the final sorted element of Array A.

8. Consider the weighted graph G_1 in Fig-1. Suppose the nodes are stored in an array Data as follows:

DATA : X, Y, S, T

(a) Find the weight matrix of G_1

(b) Find the matrix Q of shortest paths using Warshall Algo.



Ans: (a) the weighted matrix of $G_1 =$

	x	y	s	T
x	0	7	0	0
y	3	0	2	0
s	0	0	0	5
T	6	1	4	0

(b) $Q_0 = \begin{bmatrix} \infty & 7 & \infty & \infty \\ 3 & \infty & 2 & \infty \\ \infty & \infty & \infty & 5 \\ 6 & 1 & 4 & \infty \end{bmatrix}$ $\begin{bmatrix} - & XY & - & - \\ YX & - & YS & - \\ - & - & - & ST \\ TX & TY & TS & - \end{bmatrix}$

$Q_1 = \begin{bmatrix} \infty & 7 & \infty & \infty \\ 3 & 10 & 2 & \infty \\ \infty & \infty & \infty & 5 \\ 6 & 1 & 4 & \infty \end{bmatrix}$ $\begin{bmatrix} - & XY & - & - \\ YX & YXY & YS & - \\ - & - & - & ST \\ TX & TY & TS & - \end{bmatrix}$

$Q_2 = \begin{bmatrix} 10 & 7 & 9 & \infty \\ 3 & 10 & 2 & \infty \\ \infty & \infty & \infty & 5 \\ 4 & 1 & 3 & \infty \end{bmatrix}$ $\begin{bmatrix} XYX & XY & XYS & - \\ YX & YXY & YS & - \\ - & - & - & ST \\ TYX & TY & TYS & - \end{bmatrix}$

$$Q_3 = \begin{bmatrix} 10 & 7 & 9 & 14 \\ 3 & 10 & 2 & 7 \\ \alpha & \alpha & \alpha & 5 \\ 4 & 1 & 3 & 8 \end{bmatrix} \qquad \begin{bmatrix} xYx & xY & xYS & xYST \\ Yx & YxY & YS & YST \\ - & - & - & ST \\ TYx & TY & TYS & TYST \end{bmatrix}$$

$$Q_4 = \begin{bmatrix} 10 & 7 & 9 & 14 \\ 3 & 8 & 2 & 7 \\ 9 & 6 & 8 & 5 \\ 4 & 1 & 3 & 8 \end{bmatrix} \qquad \begin{bmatrix} xYx & xY & xYS & xYST \\ Yx & YSTY & YS & YST \\ STx & STY & STYS & ST \\ TYx & TY & TYS & TYST \end{bmatrix}$$

$$\begin{cases} Q_1[2,2] = MIN[Q_0(2,2),\ Q_0(2,1)+Q_0(1,2)] = MIN(\alpha, 3+7) = 10 \end{cases}$$

$$\begin{cases} Q_2[1,1] = MIN[Q_1(1,1),\ Q(1,2)+Q_1(2,1)] = MIN(\alpha, 7+3) = 10 \\ Q_2[4,1] = MIN(Q_1(1,4),\ Q_1(4,2)+Q_1(2,1) = MIN(6, 1+3) = 4 \\ Q_2[4,3] = MIN(Q_1[4,3),\ Q_1(4,2)+Q_1(2,3) = MIN(4, 1+2) = 3 \end{cases}$$

$$\begin{cases} Q_3[1,4] = MIN[Q_2(1,4), Q_2(1,3)+Q_2(3,4)] = MIN(\infty, 9+5) = 14 \\ Q_3[2,4] = MIN[Q_2(2,4), Q_2(2,3)+Q_2(3,4)] = MIN(\alpha, 2+5) = 7 \\ Q_3[4,4] = MIN[Q_2(4,4), Q_2(4,3)+Q_2(3,4)] = MIN(\alpha, 3+5) = 8 \end{cases}$$

$$\begin{cases} Q_4[2,2] = MIN[Q_3(2,2),\ Q_3(2,4)+Q_3(4,2)] = MIN(10, 7+1) = 8 \\ Q_4[3,1] = MIN[Q_3[3,1],\ Q_3(3,4)+Q_3(4,1)] = MIN(\alpha, 5+4) = 9 \\ Q_4[3,2] = MIN[Q_3(3,2),\ Q_3(3,4)+Q_3(4,2)] = MIN(\alpha, 5+1) = 6 \\ Q_4[3,3] = MIN[Q_3(3,3),\ Q_3(3,4)+Q_3(4,3)] = MIN(\alpha, 5+3) = 8 \end{cases}$$