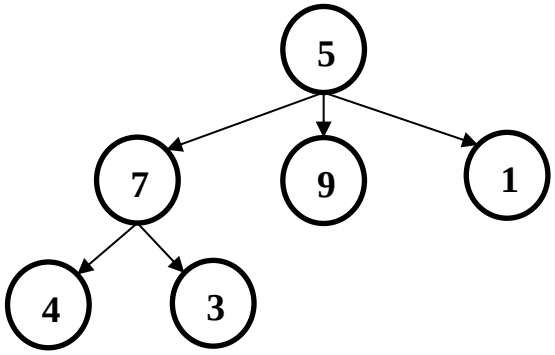
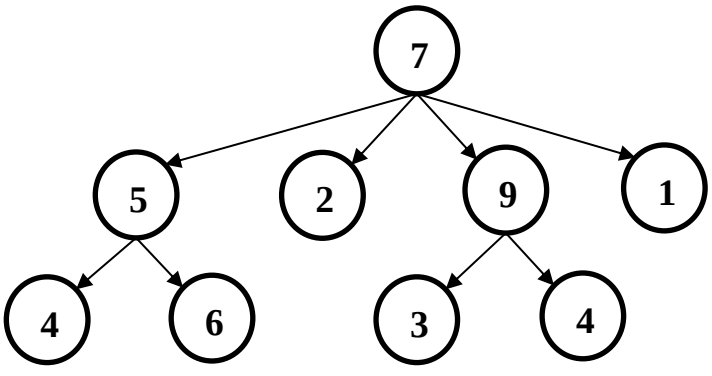
| Lab | Type | Practical |
|---|---|---|
| LAB-1 | | **Practice decision making statements in C.** |
| | A | 1. WAP to find out largest number from given two numbers. |
| | A | 2. WAP to find out largest number from given three numbers. |
| | B | 3. WAP to perform Addition, Subtraction, Multiplication and Division of 2 numbers as per user's choice. |
| | B | 4. WAP to read marks of five subjects. Calculate percentage and print class accordingly. Fail if percentage is below 35, Pass Class if percentage between 35 to 45, Second Class if percentage is between 45 to 60, First Class if percentage is between 60 to 70, and Distinctions if percentage more than 70. |
| LAB-2 | | **Practice while loop in C.** |
| | A | 5. WAP to print 1 to 10 numbers. |
| | A | 6. WAP to print 1 to N numbers. |
| | B | 7. WAP to print odd numbers between 1 to N. |
| | B | 8. WAP to print numbers between two given numbers which is divisible by 2 but not divisible by 4. |
| | C | 9. WAP to print sum of series 1 + 4 + 9 + 16 + 25 + 36 + …n. |
| | C | 10. WAP to print sum of series 1 – 2 + 3 – 4 + 5 – 6 + 7 … n. |
| | C | 11. WAP to print sum of series 1+1/2+1/3+1/4+⋯+1/n. |
| LAB-3 | | **Practice do while loop in C.** |
| | A | 12. WAP to print 1 to 10 numbers. |
| | A | 13. WAP to print 1 to n numbers. |
| | B | 14. WAP to print odd numbers between 1 to n. |
| | B | 15. WAP to print numbers between two given numbers which is divisible by 2 but not divisible by 4. |
| | C | 16. WAP to print sum of series 1 + 4 + 9 + 16 + 25 + 36 + …n. |
| | C | 17. WAP to print sum of series 1 – 2 + 3 – 4 + 5 – 6 + 7 … n. |
| | C | 18. WAP to print sum of series 1+1/2+1/3+1/4+⋯+1/n. |
| LAB-4 | | **Practice for loop in C.** |
| | A | 19. WAP to print 1 to 10 numbers. |
| | A | 20. WAP to print 1 to n numbers. |
| | B | 21. WAP to print odd numbers between 1 to n. |
| | B | 22. WAP to print numbers between two given numbers which is divisible by 2 but not divisible by 4. |
| | C | 23. WAP to print sum of series 1 + 4 + 9 + 16 + 25 + 36 + …n. |
| | C | 24. WAP to print sum of series 1 – 2 + 3 – 4 + 5 – 6 + 7 … n. |
| | C | 25. WAP to print sum of series 1+1/2+1/3+1/4+⋯+1/n. |

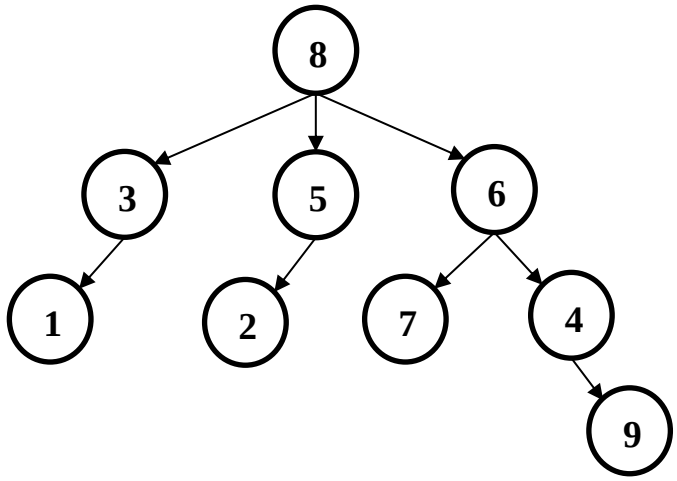
| | | |
|---|---|---|
| LAB-5 | | **Use array in C.** |
| | A | 26. WAP to read n numbers in an array and print them. |
| | A | 27. WAP to read n numbers in an array and print them in reverse order. |
| | B | 28. WAP to find Maximum, Minimum, Sum, and Average of given numbers in an array. |
| | C | 29. WAP to count numbers higher than average of an array. |
| LAB-6 | | **Use structure in C.** |
| | A | 30. Write a program to create structure of book with book title, author name, publication, and price. Read data of n books and display them. |
| | C | 31. Write a program to read data of students in array of structure and print it. |
| LAB-7 | | **Implement PUSH POP and Display operations on Stack.** |
| | A | 32. Perform following operations on a stack (Consider current stack is empty and stack size is 5):<br>a. Push: 15, 75, 32<br>b. Pop<br>c. Push: 14, 16<br>d. Display<br>e. Pop<br>f. Push: 28, 39, 49<br>g. Display |
| | A | 33. WAP to implement following operation on stack: PUSH, POP, and Display. |
| LAB-8 | | **Implement PEEP and CHANGE operations on Stack.** |
| | A | 34. Perform following operations on a stack (Consider current stack is S[15, 75, 14, 28, __] and stack size is 5):<br>a. Change 4th element from top to 46.<br>b. Peep: 3rd element from top.<br>c. Change 2nd element from top to 95.<br>d. Peep: 2nd element from top. |
| | A | 35. WAP to implement following operation on stack: PEEP and CHANGE |

| | | |
|---|---|---|
| LAB-9 | A | **Practice conversion of infix notations to postfix notations.**<br>36. Convert following expression to postfix notations:<br>a. A - B / ( C * D ^ E )<br>b. A + B ^ C ^ D - E * F / G<br>c. ( A + B * C / D - E + F / G / ( H + I ) )<br>d. ( A + B ) * C + D / ( B + A * C ) + D<br>e. A + B - C * D / E + F $ G / ( I + J )<br>f. ( a + b ^ c ^ d ) * ( e + f / d ) |
| LAB-10 | A | **Practice conversion of infix notations to prefix notations.**<br>37. Convert following expression to prefix notations:<br>a. A - B / ( C * D ^ E )<br>b. A + B ^ C ^ D - E * F / G<br>c. ( A + B * C / D - E + F / G / ( H + I ) )<br>d. ( A + B ) * C + D / ( B + A * C ) + D<br>e. A + B - C * D / E + F $ G / ( I + J )<br>f. ( a + b ^ c ^ d ) * ( e + f / d ) |
| LAB-11 | A | **Practice evaluation of postfix and prefix notations.**<br>38. Evaluate following expression:<br>a. 5 6 2 + * 1 2 4 / - +<br>b. A B + C D / * G H * + (where A=2,B=4,C=6,D=3,G=8,H=7)<br>c. 5 4 6 + * 4 9 3 / + *<br>d. + 8 * 3 7<br>e. - + 7 / 4 2 5<br>f. - + 4 * 2 6 + 4 3 |
| LAB-12 | A<br><br><br><br><br><br><br><br>A | **Implement simple queue.**<br>39. Perform following operations on a simple Queue (Consider current Queue is empty and queue size is 5):<br>a. Enqueue: 15, 75, 32<br>b. Dequeue<br>c. Enqueue: 14, 16<br>d. Display<br>e. Dequeue<br>f. Enqueue: 28, 39, 46<br>g. Display<br>40. WAP to implement following operation on queue: Enqueue, Dequeue, and Display. |

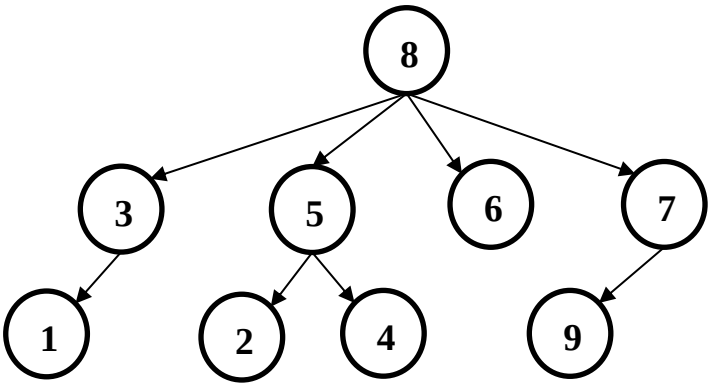
| | | |
|---|---|---|
| LAB-13 | | **Implement circular queue.** |
| | A | 41. Perform following operations on a circular Queue (Consider current Queue is empty and queue size is 5):<br>a. Enqueue: 15, 75, 32<br>b. Dequeue<br>c. Enqueue: 14, 16<br>d. Display<br>e. Dequeue<br>f. Enqueue: 28, 39, 46<br>g. Display |
| | A | 42. WAP to implement following operation on circular queue: Enqueue, Dequeue, and Display. |
| | C | 43. WAP to implement following operation on double ended queue: Enqueue, Dequeue, and Display. |
| LAB-14 | | **Implement insert and display operations on linked list.** |
| | A | 44. Perform following operations on a link list (Consider current link list is empty):<br>a. Insert first: 15, 75, 32<br>b. Display<br>c. Insert last: 78, 37, 28<br>d. Display |
| | A | 45. WAP to implement following operation on link list: Insert at first, Insert at last, and Display. |
| | C | 46. WAP to implement following operation on link list: Insert at specified location. |
| LAB-15 | | **Implement delete operations on linked list.** |
| | A | 47. Perform following operations on a link list (Consider current link list is: Start→15→75→32→78→37→28):<br>a. Delete first<br>b. Display<br>c. Delete first<br>d. Delete first<br>e. Delete first<br>f. Display |
| | A | 48. WAP to implement following operation on link list: Delete from first, Delete from last. |
| | C | 49. WAP to implement following operation on link list: Delete from specified position. |

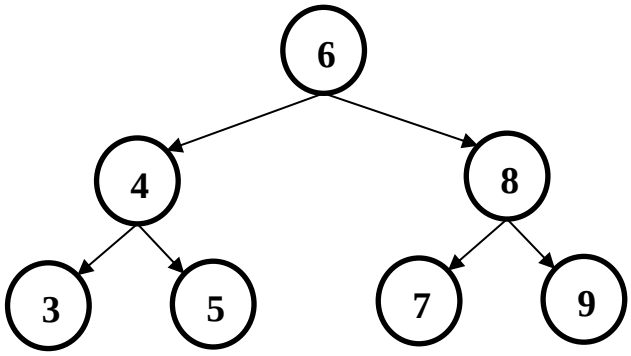
| | | |
|---|---|---|
| LAB-16 | | **Implement search operation on linked list.** |
| | A | 50. Perform following operations on a link list (Consider current link list is: Start→15→75→32→78→37→28):<br>a. Search (75)<br>b. Search (94)<br>c. Search (78) |
| | A | 51. WAP to implement following operation on link list: Search node from link list. |
| LAB-17 | | **Implement count node operation on linked list.** |
| | A | 52. Count the number of node in a link list (Consider current link list is: Start→15→75→32→78→37→28). |
| | A | 53. WAP to implement following operation on link list: Count number of node in a link list. |
| LAB-18 | | **Practice conversion of general tree to binary tree.** |
| | A | 54. Convert following general tree to binary tree:<br>  <br><br>a.<br>  <br><br>b. |

c.

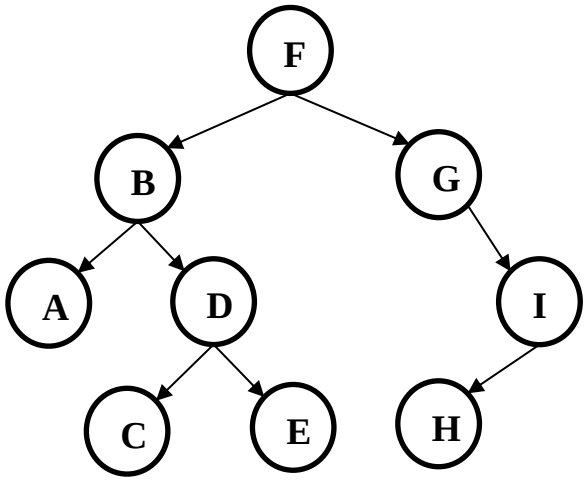



d.

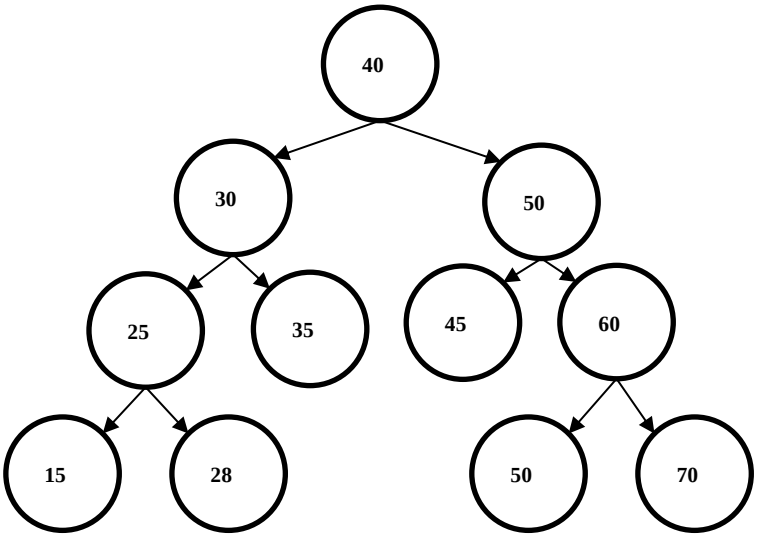


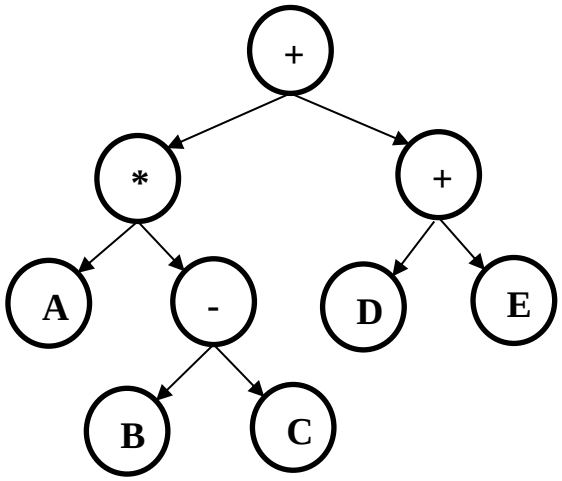
| | | |
|---|---|---|
| LAB-19 | A | **Practice insertion and deletion operation on binary search tree.**<br>55. Insert and Delete node from following binary tree as per instructions.<br>a. Tree 1 : Insert (21), Insert (28), Insert (14), Insert (32), Insert (25), Insert (18), Insert (11), Insert (30), Insert (19), Insert (15)<br>b. Tree 2 : Insert (80), Insert (88), Insert (64), Insert (52), Insert (85), Insert (38), Insert (19), Insert (22), Insert (34), Insert (92)<br>c. Tree 3 : Insert (74), Insert (28), Insert (96), Insert (83), Insert (75), Insert (35), Insert (48), Insert (31), Insert (16), Insert (4)<br>d. Tree 4 : Insert (59), Insert (38), Insert (45), Insert (76), Insert (66), Insert (19), Insert (26), Insert (55), Insert (72), Insert (49), Delete (26), Delete (19), Delete (49), Delete (72)<br>e. Tree 5 : Insert (57), Insert (34), Insert (38), Insert (66), Insert (86), Insert (99), Insert (36), Insert (25), Insert (42), Insert (40), Delete (42), Delete (86), Delete (66)<br>f. Tree 6 : Insert (56), Insert (49), Insert (65), Insert (68), Insert (38), Insert (50), Insert (63), Insert (64), Insert (76), Insert (66), Delete (65), Delete (68), Delete (56) |
| LAB-20 | A | **Practice tree traversal.**<br>56. Find out In order, Pre order and Post order traversal of tree given below:<br>a.  |

b.




c.



| | | |
|---|---|---|
| | |  <br>d. |
| LAB-21 | | **Implement linear search algorithm.** |
| | A | 57. Search element 64 in Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using linear search algorithm. |
| | A | WAP to implement linear search algorithm. |
| LAB-22 | | **Implement binary search algorithm.** |
| | A | 58. Search element 64 in Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using binary search algorithm (Note: sort given array to use binary search). |
| | A | WAP to implement linear search algorithm. |
| LAB-23 | | **Implement bubble sort algorithm.** |
| | A | 59. Sort Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using bubble sort algorithm. |
| | A | 60. WAP to implement bubble sort algorithm. |
| LAB-24 | | **Implement selection sort algorithm.** |
| | A | 61. Sort Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using selection sort algorithm. |
| | A | 62. WAP to implement selection sort algorithm. |
| LAB-25 | | **Implement insertion sort algorithm.** |
| | A | 63. Sort Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using insertion sort algorithm. |
| | A | 64. WAP to implement insertion sort algorithm. |

| | | |
|---|---|---|
| LAB-26 | | **Implement merge sort algorithm.** |
| | A | 65. Sort Array A [75, 2, 34, 19, 73, 64, 85, 26, 15, 10, 28] using merge sort algorithm. |
| | A | 66. WAP to implement merge sort algorithm. |