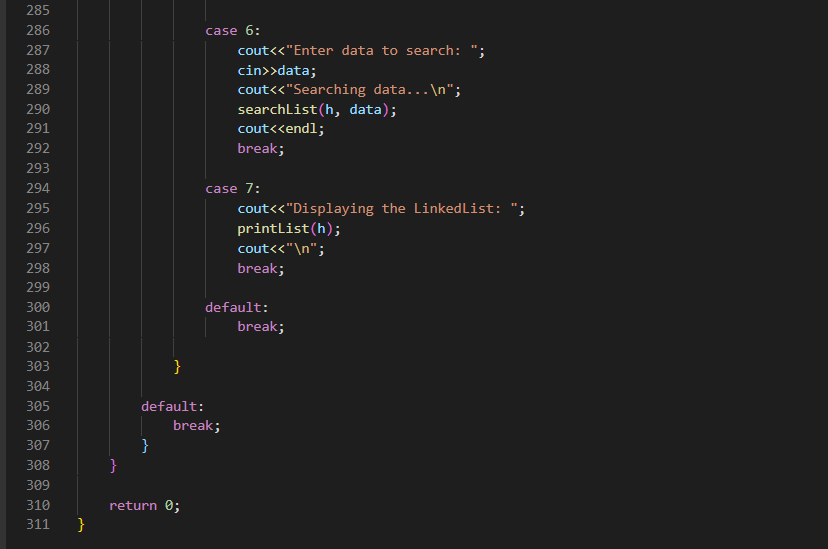
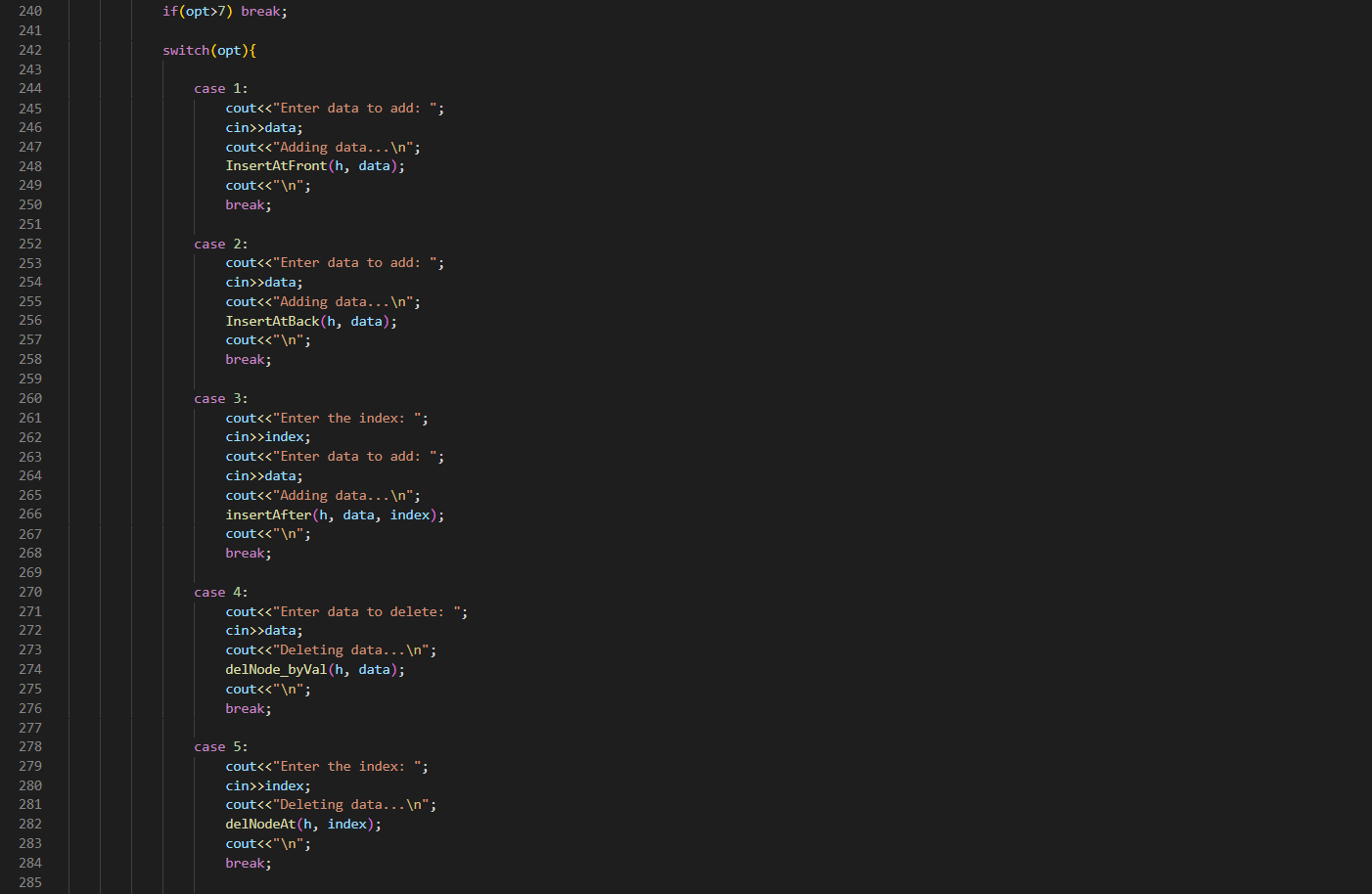
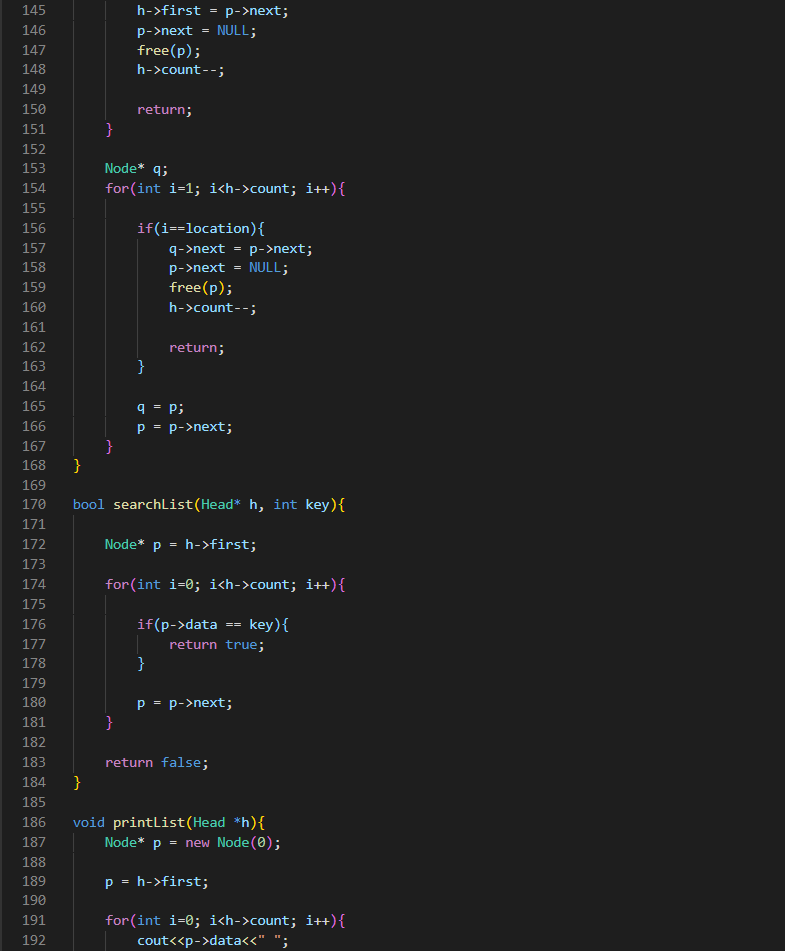
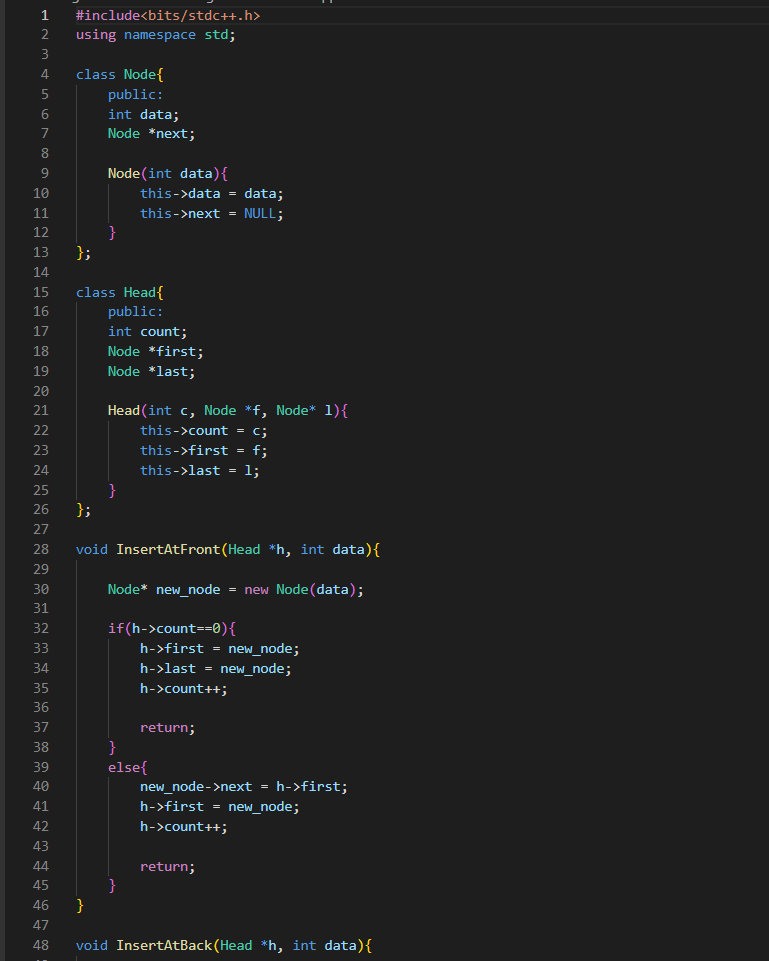
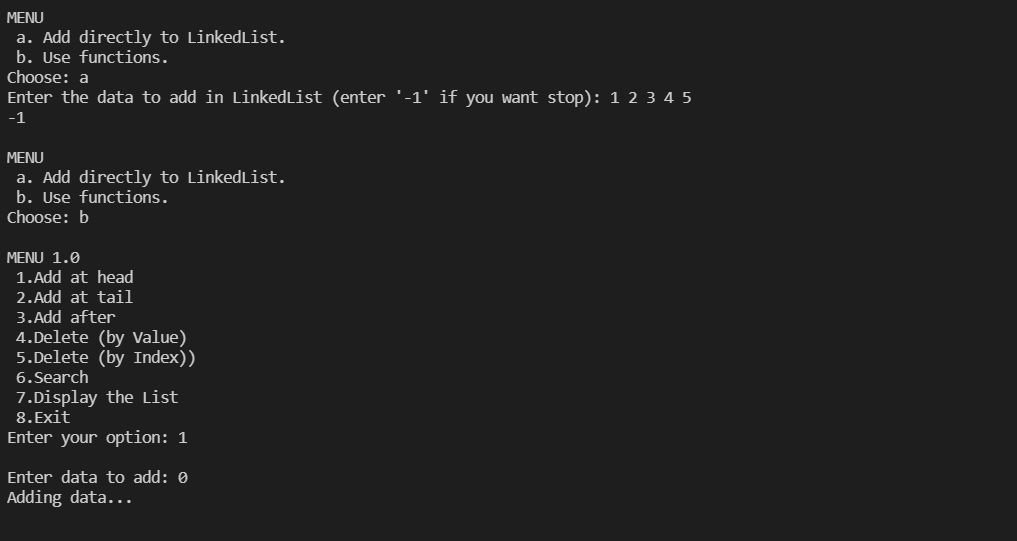
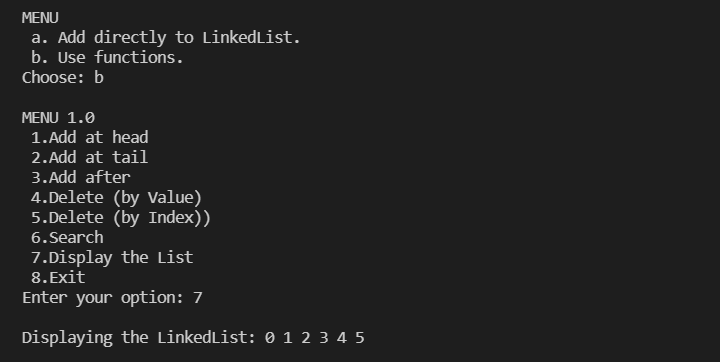
PRN: 2020BTEIT00041

Name: Om Vivek Gharge

Q1. Singly LinkedList

OUTPUT:

ALGORITHM:

ALGORITHMS:

1.Insertion In Singly Linked List A t Beginning:

Step 1: IF PTR = NULL

Write OVERFLOW

Go to Step 7

[END OF IF]

Step 2: SET NEW\_NODE = PTR

Step 3: SET PTR = PTR → NEXT

Step 4: SET NEW\_NODE → DATA = VAL

Step 5: SET NEW\_NODE → NEXT = HEAD

Step 6: SET HEAD = NEW\_NODE

Step 7: EXIT

2.Insertion in singly linked list after specified Node

STEP 1: IF PTR = NULL

WRITE OVERFLOW

GOTO STEP 12

END OF IF

STEP 2: SET NEW\_NODE = PTR

STEP 3: NEW\_NODE → DATA = VAL

STEP 4: SET TEMP = HEAD

STEP 5: SET I = 0

STEP 6: REPEAT STEP 5 AND 6 UNTIL I<loc< li=""></loc<>

STEP 7: TEMP = TEMP → NEXT

STEP 8: IF TEMP = NULL

WRITE "DESIRED NODE NOT PRESENT"

GOTO STEP 12

END OF IF

END OF LOOP

STEP 9: PTR → NEXT = TEMP → NEXT

STEP 10: TEMP → NEXT = PTR

STEP 11: SET PTR = NEW\_NODE

STEP 12: EXIT

3. Insertion in singly linked list at the tail:

Step 1: IF PTR = NULL Write OVERFLOW

Go to Step 1

[END OF IF]

Step 2: SET NEW\_NODE = PTR

Step 3: SET PTR = PTR - > NEXT

Step 4: SET NEW\_NODE - > DATA = VAL

Step 5: SET NEW\_NODE - > NEXT = NULL

Step 6: SET PTR = HEAD

Step 7: Repeat Step 8 while PTR - > NEXT != NULL

Step 8: SET PTR = PTR - > NEXT

[END OF LOOP]

Step 9: SET PTR - > NEXT = NEW\_NODE

Step 10: EXIT

4. Deletion in singly linked list at beginning:

Step 1: IF HEAD = NULL

Write UNDERFLOW

Go to Step 5

[END OF IF]

Step 2: SET PTR = HEAD

Step 3: SET HEAD = HEAD -> NEXT

Step 4: FREE PTR

Step 5: EXIT

5. Deletion in singly linked list after the specified node :

STEP 1: IF HEAD = NULL

WRITE UNDERFLOW

GOTO STEP 10

END OF IF

STEP 2: SET TEMP = HEAD

STEP 3: SET I = 0

STEP 4: REPEAT STEP 5 TO 8 UNTIL I<loc< li=""></loc<>

STEP 5: TEMP1 = TEMP

STEP 6: TEMP = TEMP → NEXT

STEP 7: IF TEMP = NULL

WRITE "DESIRED NODE NOT PRESENT"

GOTO STEP 12

END OF IF

STEP 8: I = I+1

END OF LOOP

STEP 9: TEMP1 → NEXT = TEMP → NEXT

STEP 10: FREE TEMP

STEP 11: EXIT

6. Deletion in singly linked list at the end:

Step 1: IF HEAD = NULL

Write UNDERFLOW

Go to Step 8

[END OF IF]

Step 2: SET PTR = HEAD

Step 3: Repeat Steps 4 and 5 while PTR -> NEXT!= NULL

Step 4: SET PREPTR = PTR

Step 5: SET PTR = PTR -> NEXT

[END OF LOOP]

Step 6: SET PREPTR -> NEXT = NULL

Step 7: FREE PTR

Step 8: EXIT

7. Searching in singly linked list:

Step 1: SET PTR = HEAD

Step 2: Set I = 0

STEP 3: IF PTR = NULL

WRITE "EMPTY LIST"

GOTO STEP 8

END OF IF

STEP 4: REPEAT STEP 5 TO 7 UNTIL PTR != NULL

STEP 5: if PTR → DATA = ITEM

write i+1

End of IF

STEP 6: I = I + 1

STEP 7: PTR = PTR → NEXT

[END OF LOOP]

STEP 8: EXIT

8. Reversing Singly Linked List:

STEP 1:Take 3 nodes as Node ptrOne,Node ptrTwo, Node prevNode

STEP 2:Initialize them as ptrOne = head; ptrTwo=head.next, prevNode = null.

STEP 3:Call reverseRecursion(head,head.next,null)

STEP 4: Reverse the ptrOne and ptrTwo

STEP 5: Make a recursive call for reverseRecursion(ptrOne.next,ptrTwo.next,null)