here ✕ LINEAR.c ✕

```
24      printf("%d deleted.\n", q[front++]);
25      if (front > rear) {
26          // Reset queue when empty
27          front = rear = -1;
28      }
29  }
30
31  void display() {
32      if (front == -1 || front > rear) {
33          printf("Queue is empty.\n");
34          return;
35      }
36      printf("Queue: ");
37      for (int i = front; i <= rear; i++) {
38          printf("%d ", q[i]);
39      }
40      printf("\n");
41  }
42
43  int main() {
44      int ch;
45      while (1) {
46          printf("\n1.Insert 2.Delete 3.Display 4.Exit\nChoice: ");
47          scanf("%d", &ch);
48          switch (ch) {
49              case 1: insert(); break;
50              case 2: delete(); break;
51              case 3: display(); break;
```

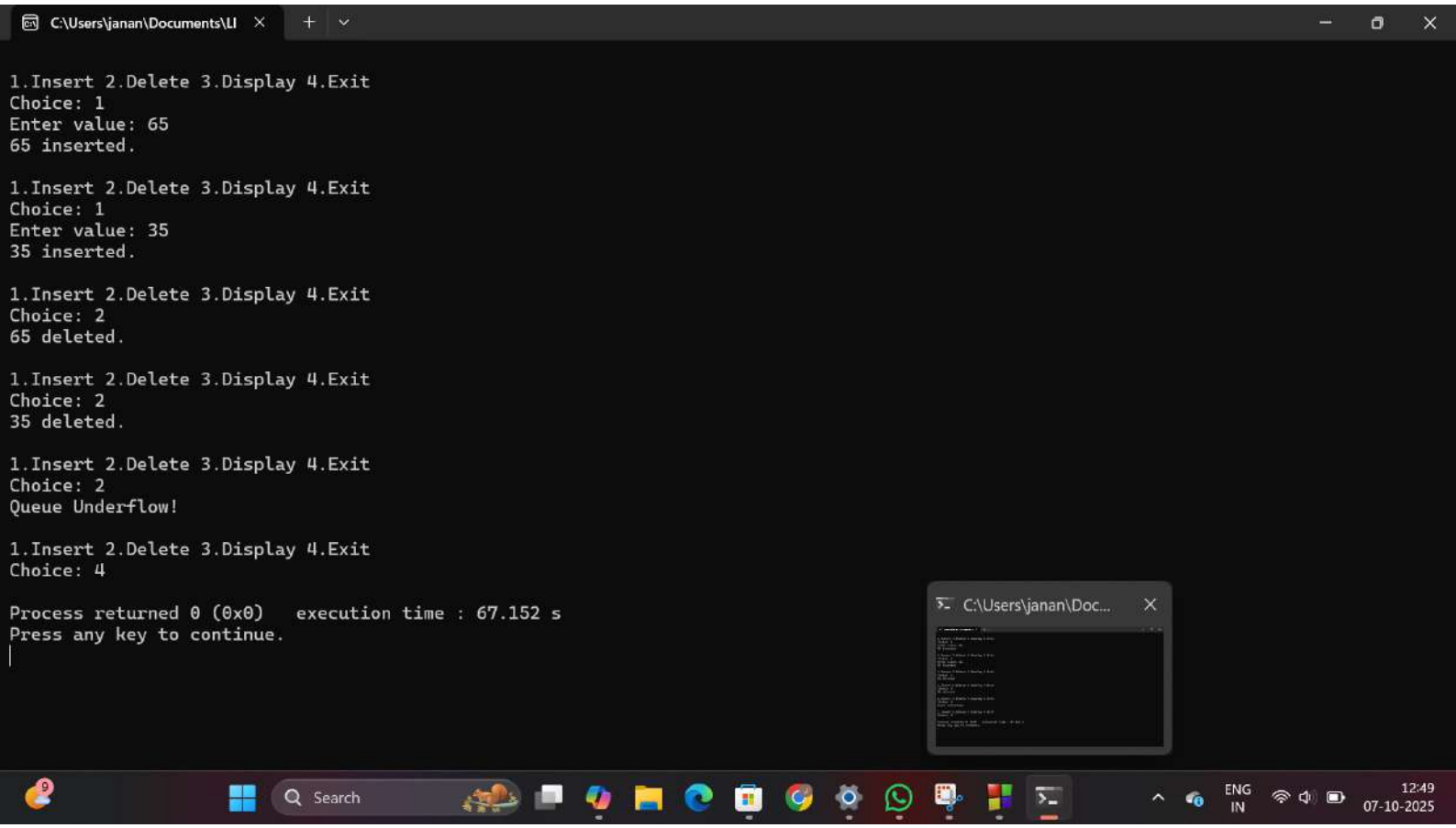
LINEAR.c

```c
#include <stdio.h>
#define SIZE 2

int q[SIZE], front = -1, rear = -1;

void insert() {
    if (rear == SIZE - 1) {
        printf("Queue Overflow!\n");
        return;
    }
    int x;
    printf("Enter value: ");
    scanf("%d", &x);
    if (front == -1) front = 0;
    q[++rear] = x;
    printf("%d inserted.\n", x);
}

void delete() {
    if (front == -1 || front > rear) {
        printf("Queue Underflow!\n");
        return;
    }
    printf("%d deleted.\n", q[front++]);
    if (front > rear) {
        // Reset queue when empty
        front = rear = -1;
    }
```


C:\Users\janan\Documents\LI
1.Insert 2.Delete 3.Display 4.Exit
Choice: 1
Enter value: 65
65 inserted.
1.Insert 2.Delete 3.Display 4.Exit
Choice: 1
Enter value: 35
35 inserted.
1.Insert 2.Delete 3.Display 4.Exit
Choice: 2
65 deleted.
1.Insert 2.Delete 3.Display 4.Exit
Choice: 2
35 deleted.
1.Insert 2.Delete 3.Display 4.Exit
Choice: 2
Queue Underflow!
1.Insert 2.Delete 3.Display 4.Exit
Choice: 4
Process returned 0 (0x0) execution time : 67.152 s
Press any key to continue.
C:\Users\janan\Doc...
Search
ENG
IN
12:49
07-10-2025