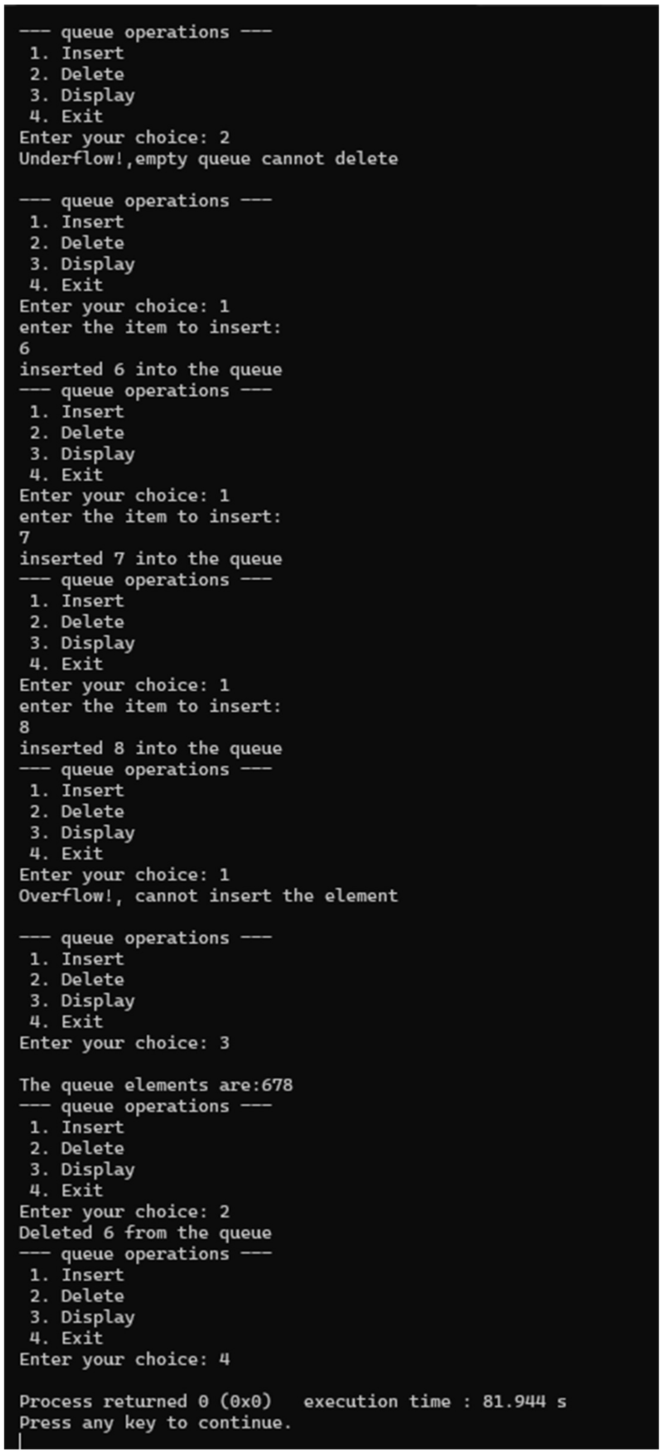


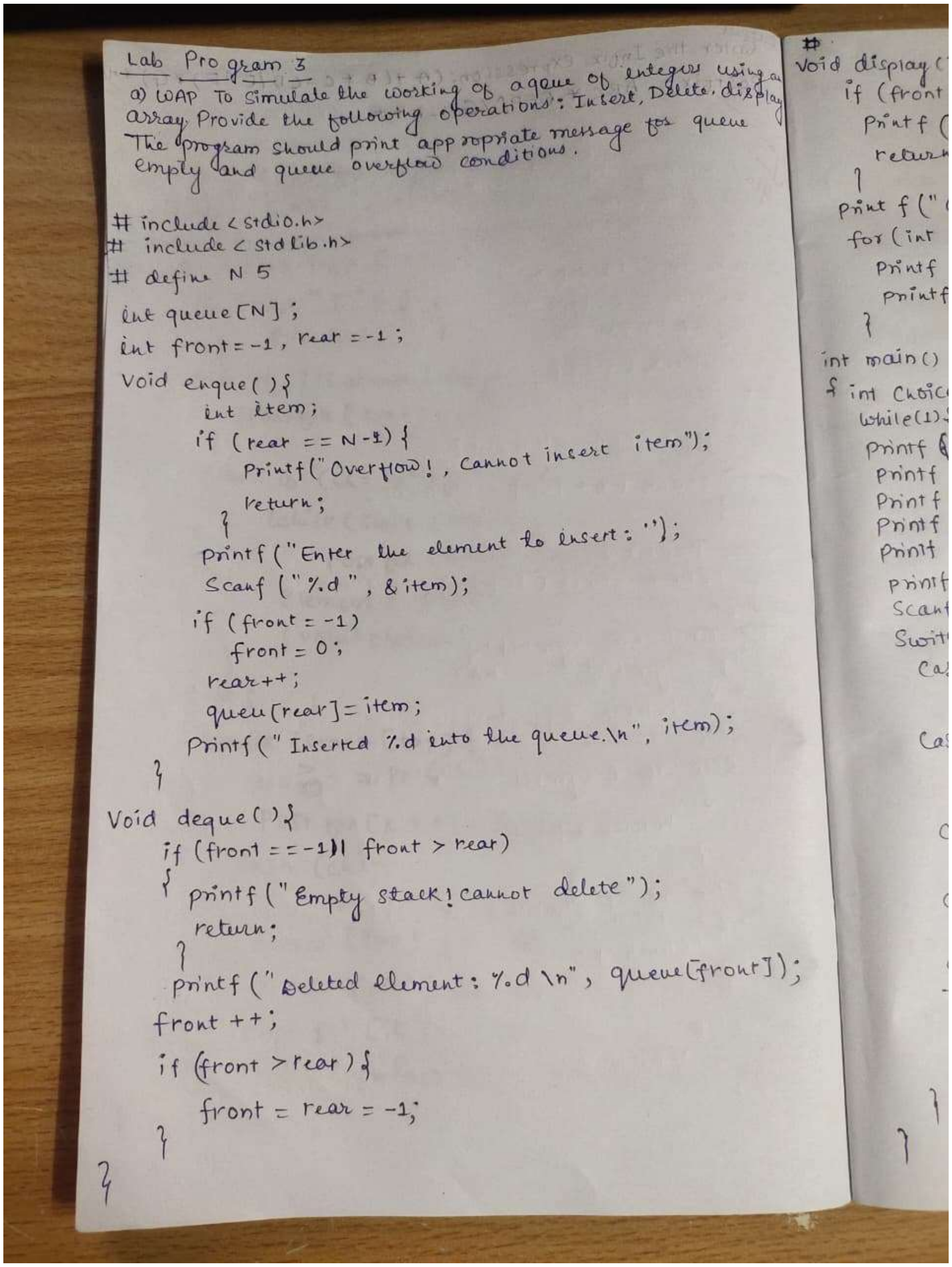
```
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 2
Underflow!,empty queue cannot delete

--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 1
enter the item to insert:
6
inserted 6 into the queue
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 1
enter the item to insert:
7
inserted 7 into the queue
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 1
enter the item to insert:
8
inserted 8 into the queue
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 1
Overflow!, cannot insert the element

--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 3

The queue elements are:678
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 2
Deleted 6 from the queue
--- queue operations ---
 1. Insert
 2. Delete
 3. Display
 4. Exit
Enter your choice: 4

Process returned 0 (0x0)   execution time : 81.944 s
Press any key to continue.
```



Lab Program 3

a) WAP To Simulate the working of a qeue of integers using an array. Provide the following operations: Insert, Delete, display. The program should print appropriate message for queue empty and queue overflow conditions.

```
# include < stdio.h>
# include < stdlib.h>
# define N 5
int queue[N];
int front=-1, rear =-1;
Void enque(){
      int item;
     if (rear == N-1) {
         Printf("Overflow!, Cannot insert  item");
         return;
       }
     printf("Enter the element to insert: ");
     Scanf ("%d", &item);
    if (front = -1)
        front = 0;
     rear++;
     queu[rear]= item;
   Printf("Inserted %d into the queue.\n", item);
  }
Void deque(){
   if (front ==-1)|| front > rear)
   {
    printf("Empty stack! cannot delete");
     return;
    }
   printf("Deleted element: %d \n", queue[front]);
  front ++;
  if (front >rear){
     front = rear = -1;
  }
}
```

```
#
Void display(
   if (front
     Printf (
     return
   }
  print f("
  for (int
     Printf
     printf
   }
int main()
{ int Choic
   while(1)
   printf
    Printf
    Print f
    Printf
    Print
     printf
     Scanf
     Swit
       Ca

       Ca

        C

        C

        }
    }
```

tegers using an
Delete. display
for queue

n");

;

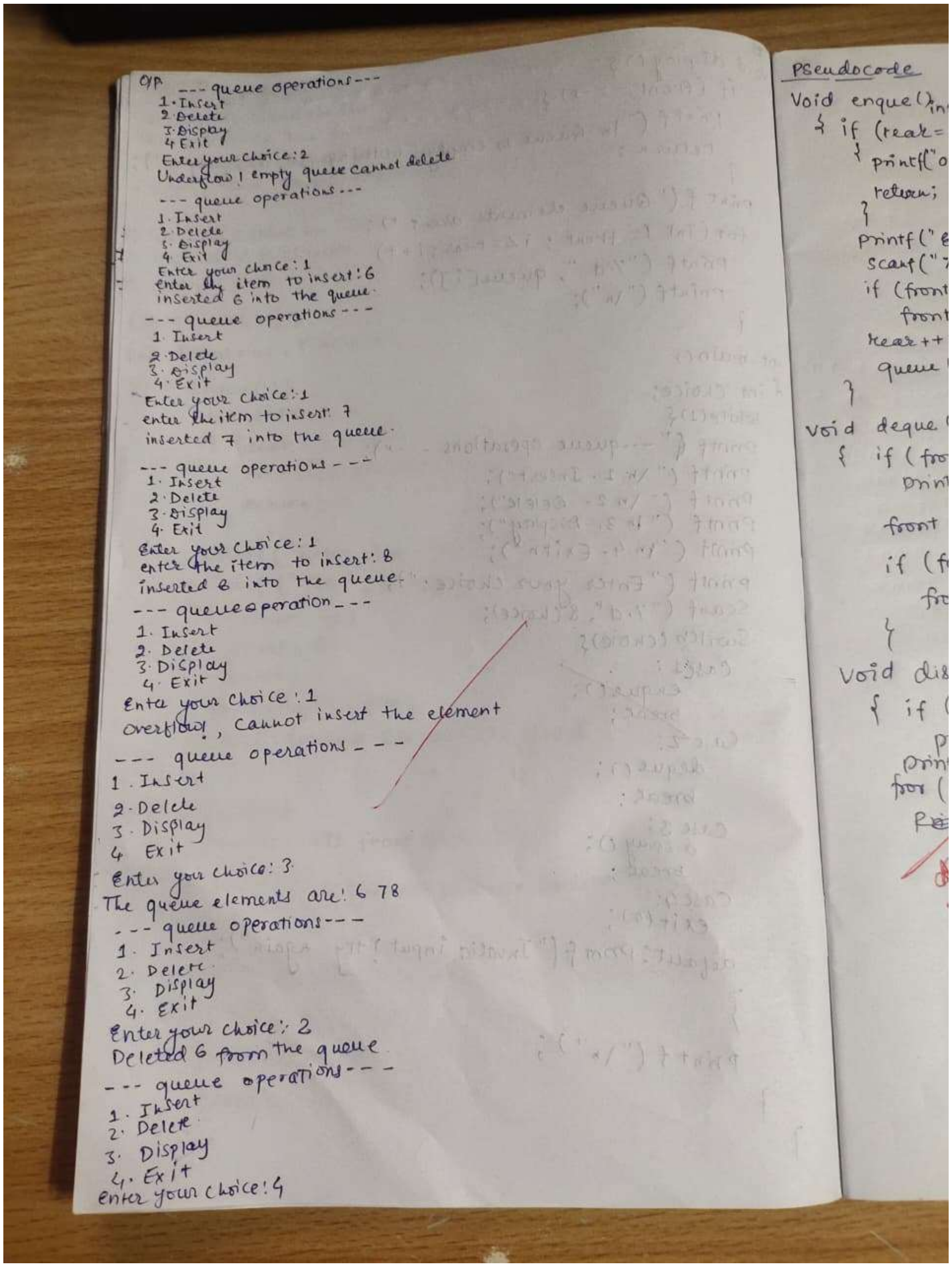
m);

ront]);

```
#.
void display(){
  if (front = = -1) {
    Printf ("\n Queue is empty, nothing to display\n");
    return;
  }
  print f("Queue elements are : ");
  for (int i= front ; i<=rear; i++)
    Printf ("%d ", queue[i]);
    printf ("\n");
  }
int main()
{ int Choice;
  while(1){
  printf ("---queue operations ---");
   printf ("\n 1. Insert");
   Print f ("\n 2. Delete");
   Printf ("\n 3. Display");
   Print ("\n 4. Exit\n");
   print ("Enter your choice:");
   Scanf ("%d", &Choice);
   Switch (choic){
     Case1 :
        ~~enque();~~
        ~~break;~~
     Case 2:
       deque ();
       break;
      Case 3:
        display ();
         break;
      Case4:
        exit(0);
      defaut: Print f ("Invalid input ! try again");

      }
      print f ("\n");

   }
}
```



```
O/P  --- queue operations---
1. Insert
2. Delete
3. Display
4 Exit
Enter your choice: 2
Underflow! empty queue cannot delete
--- queue operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
enter the item to insert: 6
inserted 6 into the queue.
--- queue operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
enter the item to insert: 7
inserted 7 into the queue.
--- queue operations ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
enter the item to insert: 8
inserted 8 into the queue.
--- queue operation ---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
overflow!, Cannot insert the element
--- queue operations ---
1. Insert
2. Delete
3. Display
4 Exit
Enter your choice: 3
The queue elements are: 6 7 8
--- queue operations---
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted 6 from the queue
--- queue operations---
1. Insert
2. Delete
3. Display
4. Exit
enter your choice: 4
```

Pseudocode

```
Void enque()in
 { if (rear=
   { printf("o
     return;
   }
   printf("E
   scanf("%
   if (front
      front
   rear++
   queue
 }
void deque
 { if (fro
      print
   front
   if (f
      fro
   }
Void dis
 { if (
      p
   print
   for (
     ~~Re~~
```

Pseudocode

```
Void enque()int item ;
 { if (rear= N-1)
     { printf("over flow")
       return;
     }
     printf("Enter the item to inser")
     scanf("%d", & item)
     if (front = =-1)
        front=0;
     rear++;
     queue [rear]= item.
 }
void deque ()
 { if ( front ==-1 & front > rear)
      Printf ("underflow")

    front ++;
    if (front >rear)
       front =rear = -1 ;
 }
void display ()
 { if (front = = - 1)
      printf(" Empty queue)',
   printf ("Queue elements are):
 for (int i= front; i≤rear; i++):
   ~~Print~~ printf (" %d"; queue(i));
```

[illegible]
8/10