Name : Parmesh Vala
Batch : SY IT
Roll No: 63

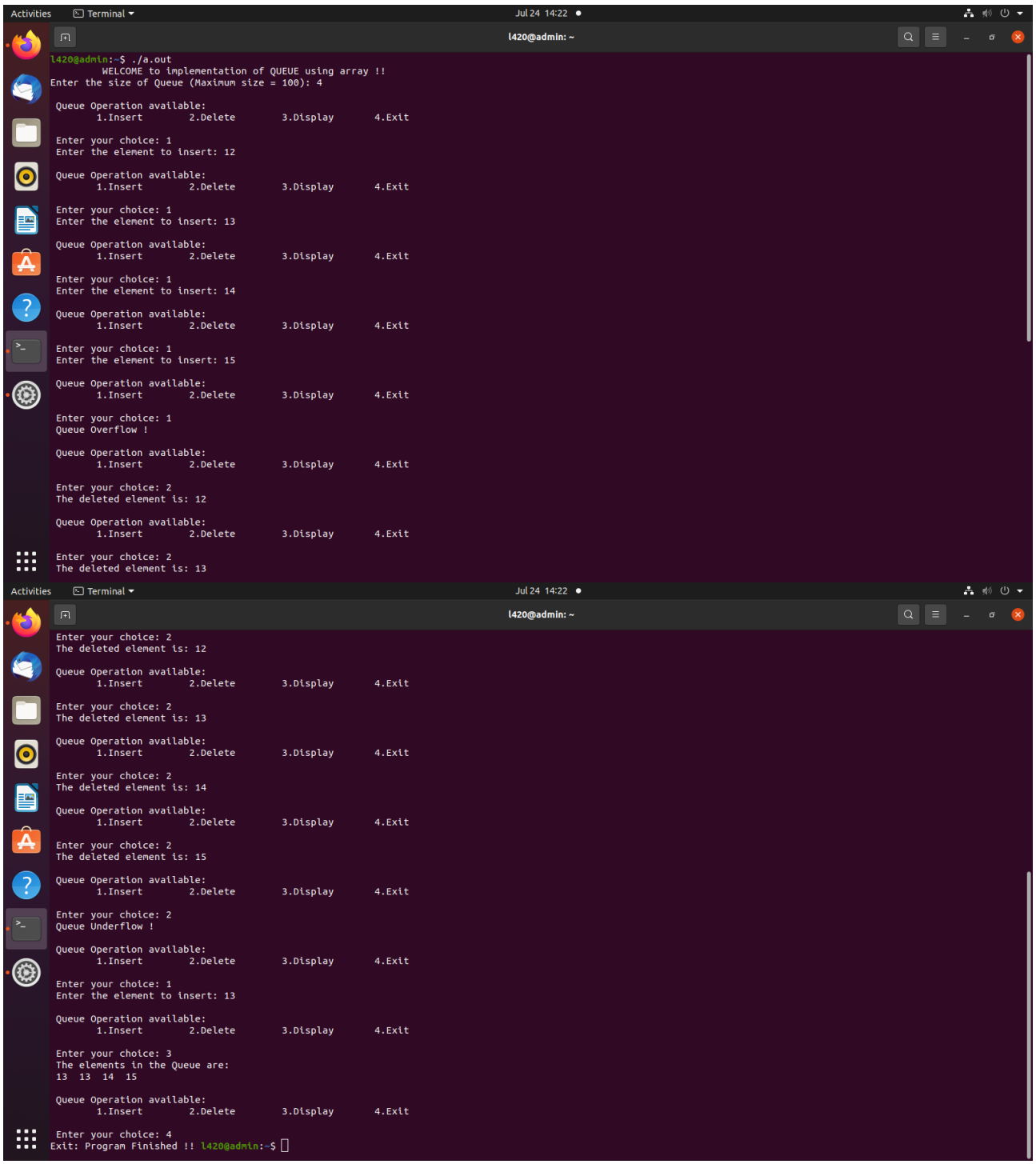
EXPERIMENT NO 2 : (Implementation of Queue using Array for real-world application.)

Code :

```
/***********************************
 Implementation of Queue Using Array
***********************************/
#include <stdio.h>
int Q[100], FRONT = -1, REAR = -1, i, n, x, choice;
void insert();
void delete ();
void display();

void main()
{
    printf("\t WELCOME to implementation of QUEUE using array !! \n");
    printf("Enter the size of Queue (Maximum size = 100): ");
    scanf("%d", &n);
    do
    {
        printf("\n Queue Operation available: \n");
        printf("\t1.Insert \t2.Delete \t3.Display \t4.Exit \n");
        printf("\n Enter your choice: ");
        scanf("%d", &choice);
        switch (choice)
        {
        case 1:
            insert();
            break;
        case 2:
            delete ();
            break;
        case 3:
            display();
            break;
        case 4:
            printf("Exit: Program Finished !! ");
            break;
        default:
            printf("Please enter a valid choice 1, 2, 3, 4 \n");
            break;
        }
    } while (choice != 4);
}

// Function to INSERT element
void insert()
{
    if (REAR >= n - 1)
    {
        printf(" Queue Overflow ! \n");
```

```
    }
    else
    {
        printf(" Enter the element to insert: ");
        scanf("%d", &x);
        REAR++;
        Q[REAR] = x;
        if (FRONT == -1)
        {
            FRONT = 0;
        }
    }
}

// Function to DELETE element
void delete ()
{
    if (FRONT == -1)
    {
        printf(" Queue Underflow ! \n");
    }
    else
    {
        printf(" The deleted element is: %d \n", Q[FRONT]);
        if (FRONT == REAR)
            FRONT = REAR = -1;
        else
            FRONT++;
    }
}

// Functiom to DISPLAY Queue
void display()
{
    if (REAR < 0)
    {
        printf(" Queue is empty ! \n");
    }
    else
    {
        printf(" The elements in the Queue are: \n");
        for (i = FRONT; i < n; i++)
        {
            printf(" %d ", Q[i]);
        }
        printf("\n");
    }
}
```

output :


Activities
Terminal
Jul 24 14:22
l420@admin: ~
l420@admin:~$ ./a.out
WELCOME to implementation of QUEUE using array !!
Enter the size of Queue (Maximum size = 100): 4
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Enter the element to insert: 12
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Enter the element to insert: 13
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Enter the element to insert: 14
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Enter the element to insert: 15
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Queue Overflow !
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
The deleted element is: 12
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
The deleted element is: 13

Activities
Terminal
Jul 24 14:22
l420@admin: ~
Enter your choice: 2
The deleted element is: 12
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
The deleted element is: 13
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
The deleted element is: 14
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
The deleted element is: 15
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 2
Queue Underflow !
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 1
Enter the element to insert: 13
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 3
The elements in the Queue are:
13 13 14 15
Queue Operation available:
1.Insert 2.Delete 3.Display 4.Exit
Enter your choice: 4
Exit: Program Finished !! l420@admin:~$