Experiment 4: Jay Patel
SYIT (42)

Code:

```
#include <stdio.h>
#define MAX 10

        int deque[MAX];
        int left = -1, right = -1;

        void input_deque(void);
        void output_deque(void);
        void insert_left(void);
        void insert_right(void);
        void delete_left(void);
        void delete_right(void);
        void display(void);

        int main() {
      int option;
                printf("\n ****MAIN MENU****");
                printf("\n 1.Input restricted deque");
                printf("\n 2.Output restricted deque");
                printf("\n Enter your option : ");
                scanf("%d", &option);

   switch (option) {
      case 1:
         input_deque();
         break;
      case 2:
         output_deque();
         break;
   }
   return 0;
}

void input_deque() {
   int option;
   do {
                printf("\n INPUT RESTRICTED DEQUE");
                printf("\n 1.Insert at right");
                printf("\n 2.Delete from left");
                printf("\n 3.Delete from right");
                printf("\n 4.Display");
                printf("\n 5.Quit");
                printf("\n Enter your option : ");
                scanf("%d", &option);

      switch (option) {
         case 1:
```

```
            insert_right();
            break;
         case 2:
            delete_left();
            break;
         case 3:
            delete_right();
            break;
         case 4:
            display();
            break;
      }
   } while (option != 5);
}

void output_deque() {
   int option;
   do {
                  printf("OUTPUT RESTRICTED DEQUE");
                  printf("\n 1.Insert at right");
                  printf("\n 2.Insert at left");
                  printf("\n 3.Delete from left");
                  printf("\n 4.Display");
                  printf("\n 5.Quit");
                  printf("\n Enter your option : ");
                  scanf("%d", &option);

                  switch (option) {
         case 1:
            insert_right();
            break;
         case 2:
            insert_left();
            break;
         case 3:
            delete_left();
            break;
         case 4:
            display();
            break;
      }
   } while (option != 5);
}

void insert_right() {
            int val;
            printf("\n Enter the value to be added: ");
            scanf("%d", &val);
   if ((left == 0 && right == MAX - 1) || (left == right + 1)) {
         printf("\n OVERFLOW");
      return;
   }
```

```
    if (left == -1) {

        left = 0;
        right = 0;
    } else {
        if (right == MAX - 1)
            right = 0;
        else
            right = right + 1;
    }
            deque[right] = val;
}

void insert_left() {
    int val;
    printf("\n Enter the value to be added: ");
    scanf("%d", &val);
    if ((left == 0 && right == MAX - 1) || (left == right + 1)) {
        printf("\n Overflow");
        return;
    }

    if (left == -1) {

        left = 0;
        right = 0;
    } else {
        if (left == 0)
            left = MAX - 1;
        else
            left = left - 1;
    }
            deque[left] = val;
}

void delete_left() {
    if (left == -1) {
        printf("\n UNDERFLOW");
        return;
    }
            printf("\n The deleted element is: %d", deque[left]);
    if (left == right) {
        left = -1;
        right = -1;
    } else {
        if (left == MAX - 1)
            left = 0;
        else
            left = left + 1;
    }
}
```

```
void delete_right() {
   if (left == -1) {
      printf("\n UNDERFLOW");
      return;
   }
   printf("\n The element deleted is: %d", deque[right]);
   if (left == right)
      {
      left = -1;
      right = -1;
   } else {
      if (right == 0)
         right = MAX - 1;
      else
         right = right - 1;
   }
}

void display() {
   int front = left, rear = right;
   if (front == -1) {
      printf("\n QUEUE IS EMPTY");
      return;
   }
   printf("\n The elements of the queue are: ");

   if (front <= rear) {
      while (front <= rear) {
         printf("%d ", deque[front]);
         front++;
        }
   } else {
      while (front <= MAX - 1) {
         printf("%d ", deque[front]);
         front++;
      }
            front = 0;
      while (front <= rear) {
         printf("%d ", deque[front]);
         front++;
      }
   }
          printf("\n");
}
```

Output:

```
****MAIN MENU****
1.Input restricted deque
2.Output restricted deque
Enter your option : 1

INPUT RESTRICTED DEQUE
1.Insert at right
2.Delete from left
3.Delete from right
4.Display
5.Quit
Enter your option : 1

Enter the value to be added: 42

INPUT RESTRICTED DEQUE
1.Insert at right
2.Delete from left
3.Delete from right
4.Display
5.Quit
Enter your option : 1

Enter the value to be added: 44

INPUT RESTRICTED DEQUE
1.Insert at right
2.Delete from left
3.Delete from right
4.Display
5.Quit
Enter your option : 4

The elements of the queue are: 42 44
```

```
The elements of the queue are: 42 44

INPUT RESTRICTED DEQUE
1.Insert at right
2.Delete from left
3.Delete from right
4.Display
5.Quit
Enter your option : 2

The deleted element is: 42
INPUT RESTRICTED DEQUE
1.Insert at right
2.Delete from left
3.Delete from right
4.Display
5.Quit
Enter your option : 5
```