5. ARRAY IMPLEMENTATION OF QUEUE ADT

Preamble

A queue is a linear data structure where elements are stored in the FIFO (First In First Out) principle where the first element inserted would be the first element to be accessed. A queue is an Abstract Data Type (ADT) similar to stack, the thing that makes queue different from stack is that a queue is open at both its ends. The data is inserted into the queue through one end and deleted from it using the other end.

Queue is very frequently used in most programming languages. A real-world example of queue can be a single-lane one-way road, where the vehicle enters first, exits first. More real-world examples can be seen as queues at the ticket windows and bus-stops.

Steps

Step 1: Initialize the queue variables front =0 and rear = -1

Step 2: Read the queue operation type.

Step 3: Check the queue operations status.

i). If it is Insertion then do the following steps

Check rear < queue_size is true increment the rear by one and read the queue element and also display queue. otherwise display the queue is full.

Go to step 2

ii). If it is deletion then do the following steps

Check rear< front is true then display the queue is empty.

Move the elements to one step forward (i.e. move to previous index).

Decreases the rear value by one (rear=rear-1).

Display queue

Go to step2.

Implementation in C

```
#include<stdio.h>
#include<conio.h>
#define n 5
void main()
```

```
{
	int queue[n],ch=1,front=0,rear=0,i,j=1,x=n;
	//clrscr();
	printf("Queue using Array");
	printf("\n1.Insertion \n2.Deletion \n3.Display \n4.Exit");
	while(ch)
	{
		printf("\nEnter the Choice:");
		scanf("%d",&ch);
		switch(ch)
		{
			case 1:
				if(rear==x)
					printf("\n Queue is Full");
				else
				{
					printf("\n Enter no %d:",j++);
					scanf("%d",&queue[rear++]);
				}
				break;
			case 2:
				if(front==rear)
				{
					printf("\n Queue is empty");
				}
				else
				{
					printf("\n Deleted Element is
					%d",queue[front++]);
					x++;
				}
				break;
```

```
                case 3:
                        printf("\n Queue Elements are:\n ");
                        if(front==rear)
                                printf("\n Queue is Empty");
                        else
                        {
                                for(i=front; i<rear; i++)
                                {
                                        printf("%d",queue[i]);
                                        printf("\n");
                                }
                        break;
                case 4:
                        exit(0);
                        default:
                                printf("Wrong Choice: please see the options");
                }
        }
    }
    getch();
}
```

Sample Input and Output

```
Queue using Array
1.Insertion
2.Deletion
3.Display
4.Exit
Enter the Choice:1

 Enter no 1:56

Enter the Choice:1

 Enter no 2:90

Enter the Choice:1

 Enter no 3:67

Enter the Choice:1

 Enter no 4:67

Enter the Choice:1

 Enter no 5:23

Enter the Choice:3

 Queue Elements are:
 56
90
67
67
23

Enter the Choice:2

 Deleted Element is 56
Enter the Choice:2

 Deleted Element is 90
Enter the Choice:3

 Queue Elements are:
 67
67
23
```