**#4. Queue**

**Roll Number:CB.EN.P2EBS22001**

**Date of Submission:21-11-2022**

**Aim:**

To perform following Queue operations using C Programming:

1. Queue Creation
2. Add an element
3. Remove an element
4. Check the status of the queue full/empty
5. View the entire queue after each of the above operation

**Tools Required:**

Text editor with C Compiler.

**Experiment:**

Code

#include <stdio.h>

#include <stdlib.h>

typedef struct Queue{

     unsigned int size;

     int front;

     int rear;

     int \*array;

}Q1;

Q1\* create(unsigned int size){

    Q1 \*queue;

    queue= (struct Queue\*)malloc(sizeof(struct Queue));

    queue->array=(int\*)malloc(size\*sizeof(int));

    queue->size=size-1;

    queue->front=-1;

    queue->rear=-1;

    return queue;

}

int isFull(Q1\*queue){

    if(queue->rear==queue->size){

        printf("Queue is Full\n");

    }

    return queue->rear!=queue->size;

}

int isEmpty(Q1 \*queue){

    if(queue->front==-1){

        printf("Queue is Empty\n");

    }

    return queue->front==-1;

}

void enqueue(Q1 \*queue){

    int queueElement;

    if(isFull(queue)){

        if(queue->front==-1)

        queue->front=0;

        printf("Enter the number to be saved in the queue:");

        scanf("%d",&queueElement);

        queue->array[++queue->rear]=queueElement;

    }

}

void dequeue(Q1 \*queue){

    if(queue->front==queue->rear){

    queue->array[queue->front++]= NULL;

     queue->front=queue->rear=-1;

    }

    if(!isEmpty(queue)){

        queue->array[queue->front++]= NULL;

    }

}

void peek(Q1 \*queue){

    printf("%d",queue->array[queue->front]);

}

void view(Q1 \*queue){

     int i,value;

     value=(queue->front!=-1)?queue->front:0;

     for(i=value;i<=queue->size;i++){

        printf("\n--------------\n");

        printf("|\t");

        if(!queue->array[i])

        printf("\t");

        if(queue->array[i])

        printf("%d\t",queue->array[i]);

        printf("|\n");

        printf("-------------\n");

    }

}

int main() {

   Q1 \*queue;

   unsigned int size, userInput;

   printf("Enter the number of queue elements:");

   scanf("%d",&size);

   queue=create(size);

  while(1){

     printf("========OPTIONS========");

      printf("\n1-Enqueue\n2-Dequeue\n3-isFull()\n4-isEmpty()\n5-peek()\n6-view()\n7-exit\n");

      printf("Enter the option:");

    scanf("%d",&userInput);

    switch(userInput){

        case 1:enqueue(queue);

               break;

        case 2:dequeue(queue);

               break;

        case 3:isFull(queue);

               break;

        case 4:isEmpty(queue);

               break;

        case 5:peek(queue);

               break;

        case 6:view(queue);

               break;

        case 7: exit(1);

        default: printf("Enter a valid number");

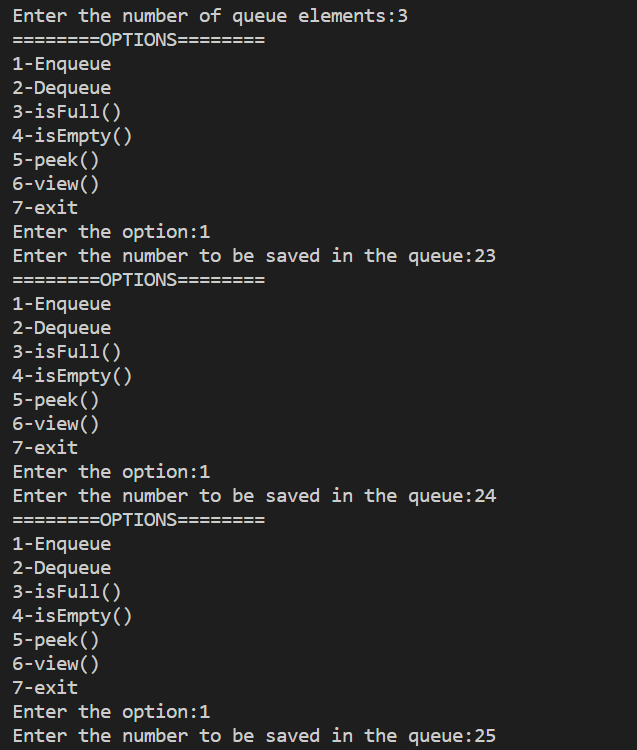
    }

  }

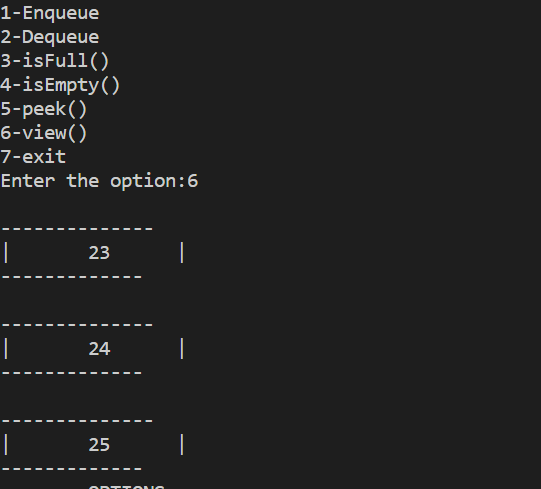
    return 0;

}

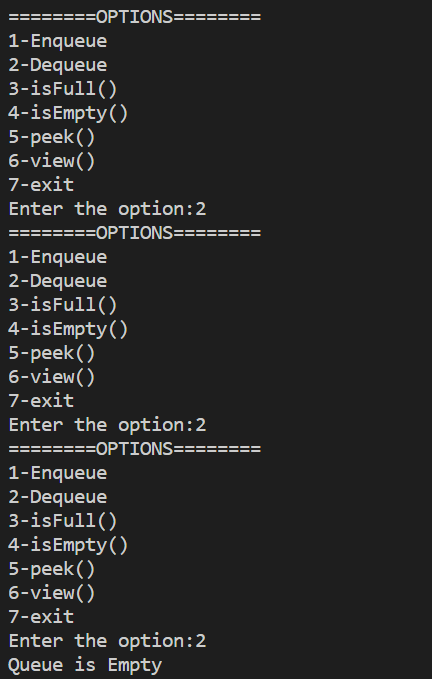
OUTPUT



VIEW



**Dequeue**

****

**Inference and Result:**

Queue is implemented in C and queue operations are performed and result is observed.