

Practical no: 11

Objective: Write a program in C to implement queue using

1. Array
2. Linked List

Program Codes: Following is the code of this problems in C:-

1. Practical11a.c

```
#include <stdio.h>
#define MAX 6
int queue[MAX], front = -1, rear = -1, choice = 0;

void enqueue(){
    int element;
    printf("\nEnter value to enqueue: ");
    scanf("%d", &element);
    if(rear == MAX-1){
        printf("\nQueue is full");
        return;
    }
    if(front == -1){
        front++;
    }
    rear++;
    queue[rear] = element;
}

void dequeue(){
    if(front == -1){
        printf("\nQueue is empty");
        return;
    }
    printf("\nDequeued value: %d", queue[front]);
    if(front == rear){
        front = rear = -1;
        return;
    }
    front++;
}
```

```

void value_at_front(){
    if(front == -1){
        printf("\nQueue is empty");
        return;
    }
    printf("\nValue at front is: %d", queue[front]);
}

```

```

void display(){
    int i = 0;
    if (front == -1){
        printf("\nQueue is empty");
        return;
    }
    printf("\nValues are: ");
    for(i = front; i ≤ rear; i++){
        printf("%5d", queue[i]);
    }
}

```

```

int main(){
    while(1){
        printf("\nSelect the option from menu"
            "\n1. Enqueue"
            "\n2. Dequeue"
            "\n3. Peek at front"
            "\n4. Display"
            "\n0. exit"
            "\n enter choice: ");
        scanf("%d", &choice);

        switch(choice){
            case 1: enqueue(); break;
            case 2: dequeue(); break;
            case 3: value_at_front(); break;
            case 4: display(); break;
            case 0: return 0;
            default: printf("\nWrong choice retry");
        }
    }
}

```

2. Practical11b.c

```
#include <stdlib.h>
#include <stdio.h>
typedef struct node node;

struct node{
    int data;
    node *link;
};

node *head= NULL, *tail=NULL;

void enqueue(){
    node *new_node = (node*)malloc(sizeof(node));
    printf("Enter data to enqueue:");
    scanf("%d", &new_node->data);
    new_node->link = NULL;

    if(head==NULL){
        tail = head = new_node;
        return;
    }
    tail->link = new_node;
    tail = tail->link;
}

void dequeue(){
    if(head == NULL){
        printf("Queue is empty");
        return;
    }
    node *temp = head;
    if(head == tail){
        tail = NULL;
    }
    head = head->link;
    printf("Dequeued data is: %d", temp->data);
    free(temp);
}

void value_at_front(){
    if(head == NULL){
```

```

        printf("Queue is empty");
        return;
    }
    printf("Value at front is: %d", head→data);
}

void display(){
    node *temp=head;
    while(temp≠NULL){
        printf("%5d", temp→data);
        temp= temp→link;
    }
}

int main(){
    int choice;
    while(1){
        printf("\n\nSelect the option from menu"
            "\n1. Enqueue"
            "\n2. Dequeue"
            "\n3. Peek at front"
            "\n4. Display"
            "\n0. exit"
            "\n enter choice: ");
        scanf("%d", &choice);

        switch(choice){
            case 1: enqueue(); break;
            case 2: dequeue(); break;
            case 3: value_at_front(); break;
            case 4: display(); break;
            case 0: return 0;
            default: printf("\nWrong choice retry");
        }
    }
}

```

Output: Following is the output of the program:-

```
C:\Users\lenovo\Desktop>gcc queue_impl.c && a.exe

Select the option from menu
1. Enqueue
2. Dequeue
3. Peek at front
4. Display
0. exit
enter choice: 1

Enter value to enqueue: 101

Select the option from menu
1. Enqueue
2. Dequeue
3. Peek at front
4. Display
0. exit
enter choice: 1

Enter value to enqueue: 102
Select the option from menu
1. Enqueue
2. Dequeue
3. Peek at front
4. Display
0. exit
enter choice: 3

Value at front is: 101
Select the option from menu
1. Enqueue
2. Dequeue
3. Peek at front
4. Display
0. exit
enter choice: 4

Values are: 101 102
```

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Dequeued value: 101

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Dequeued value: 102

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Queue is empty

```
C:\Users\lenovo\Desktop>gcc queue_impl_ll.c && a.exe
```

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 1

Enter value to enqueue: 101

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 1

Enter value to enqueue: 102

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 3

Value at front is: 101

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 4

Values are: 101 102

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Dequeued value: 101

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Dequeued value: 102

Select the option from menu

- 1. Enqueue
- 2. Dequeue
- 3. Peek at front
- 4. Display
- 0. exit

enter choice: 2

Queue is empty