

queue-using-array.c

//write a program to implement straight-line queue using array.

```
#include <stdio.h>
```

```
#define max 10
```

```
int queue[max], rear = -1, front = -1;
```

```
void enqueue(int data){
    queue[++rear] = data;
    if(front == -1){
        front++;
    }
}
```

```
int dequeue(){
    return queue[front++];
}
```

```
void display(){
    if(rear == front && front != max-1){
        printf("Empty Queue.\n");
        return;
    }
    int i = front;
    while(i <= rear){
        printf("%d",queue[i]);
        i++;
        if(i > rear){
            break;
        }
        printf(" -> ");
    }
    printf("\n");
}
```

```
int isEmpty(){
    return (rear == -1 || front == max);
}
```

```
int isFull(){
    return rear >= max-1;
}
```

```
int len(){
    if(rear == -1){
        return 0;
    }
    return rear - front + 1;
}
```

```
void main(){
    if(isEmpty()){
        printf("The Queue is empty.\n");
    }else{
        printf("The Queue is not empty.\n");
    }
    printf("the length of the Queue is %d\n",len());
    enqueue(10);
    printf("the length of the Queue is %d\n",len());
}
```

```

enqueue(102);
enqueue(15);
enqueue(13);
enqueue(12);
display();
printf("the length of the Queue is %d\n",len());
if(isFull()){
    printf("The Queue is Full.\n");
}else{
    printf("The Queue is not Full.\n");
}
printf("Removed %d\n",dequeue());
printf("Removed %d\n",dequeue());
printf("Removed %d\n",dequeue());
display();
}

```

OUTPUT

PS S:\Workspace\CollegeWork\DataStructure\Temp> gcc .\queue-using-array.c

PS S:\Workspace\CollegeWork\DataStructure\Temp> ./a

The Queue is empty.

the length of the Queue is 0

the length of the Queue is 1

10 -> 102 -> 15 -> 13 -> 12

the length of the Queue is 5

The Queue is not Full.

Removed 10

Removed 102

Removed 15

13 -> 12

PS S:\Workspace\CollegeWork\DataStructure\Temp>