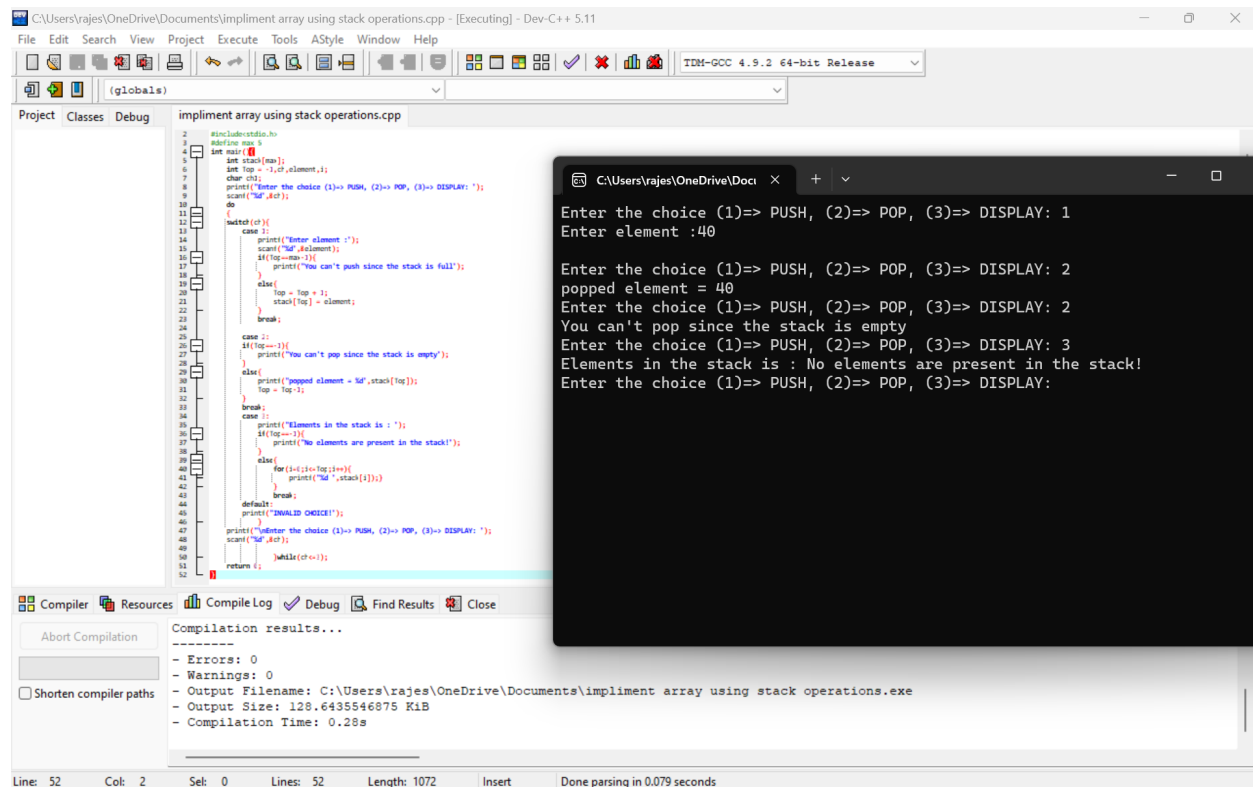


22) write a c programme to implement array using stack operations



```
1 //C program to implement array using stack operations
2 #include<stdio.h>
3 #define max 5
4 int main()
5 {
6     int stack[max];
7     int top = -1;
8     char ch;
9     printf("Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: ");
10    scanf("%d",&ch);
11    do
12    {
13        switch(ch)
14        {
15            case 1:
16                printf("Enter element :");
17                scanf("%d",&element);
18                if(top==max-1)
19                {
20                    printf("You can't push since the stack is full!");
21                }
22                else
23                {
24                    top = top + 1;
25                    stack[top] = element;
26                }
27                break;
28            case 2:
29                if(top==0)
30                {
31                    printf("You can't pop since the stack is empty!");
32                }
33                else
34                {
35                    printf("Popped element = %d",stack[top]);
36                    top = top - 1;
37                }
38                break;
39            case 3:
40                printf("Elements in the stack is : ");
41                if(top==0)
42                {
43                    printf("No elements are present in the stack!");
44                }
45                else
46                {
47                    for(i=0;i<=top;i++)
48                    {
49                        printf("%d ",stack[i]);
50                    }
51                }
52                break;
53            default:
54                printf("INVALID CHOICE!");
55        }
56        printf("Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: ");
57        scanf("%d",&ch);
58    } while(ch!=0);
59    return 0;
60 }
```

Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: 1
Enter element : 40

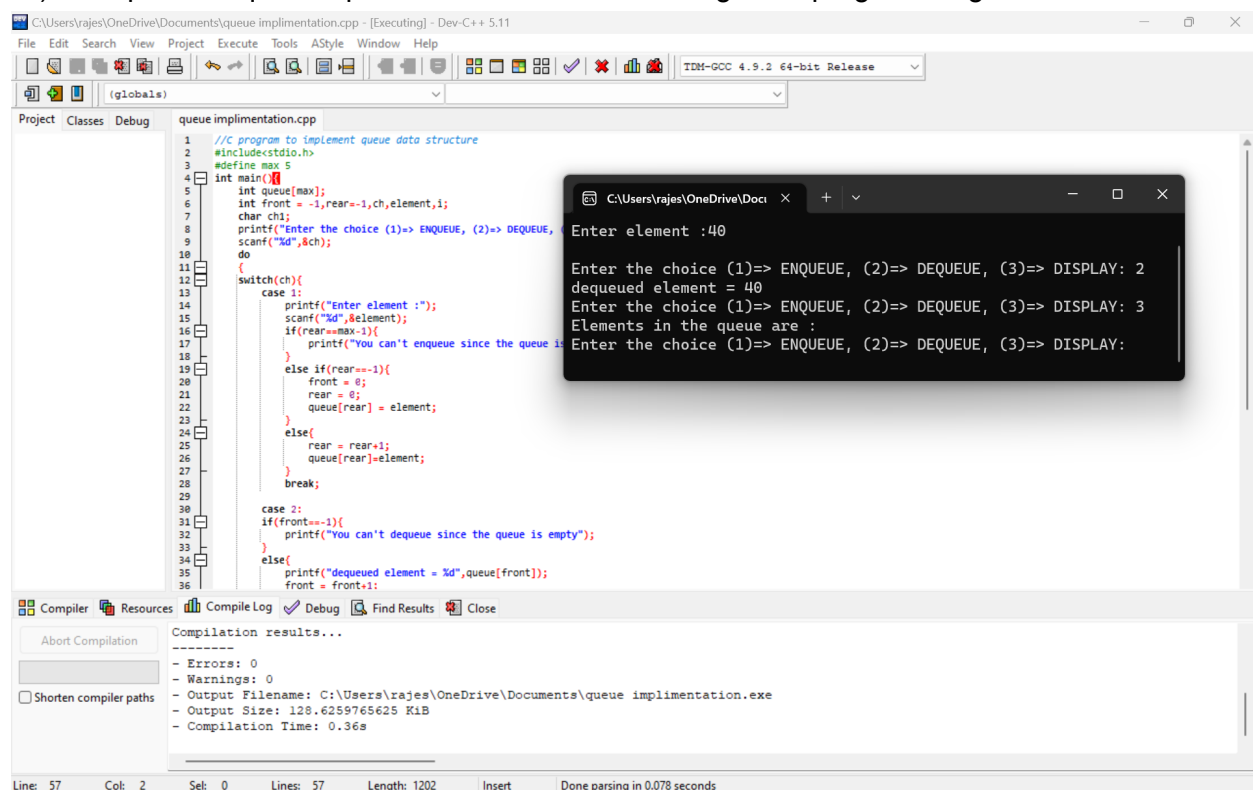
Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: 2
popped element = 40
Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: 2
You can't pop since the stack is empty
Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY: 3
Elements in the stack is : No elements are present in the stack!
Enter the choice (1)=> PUSH, (2)=> POP, (3)=> DISPLAY:

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\rajes\OneDrive\Documents\implitment array using stack operations.exe
- Output Size: 128.6435546875 KiB
- Compilation Time: 0.28s

Line: 52 Col: 2 Sel: 0 Lines: 52 Length: 1072 Insert Done parsing in 0.079 seconds

23) To implement queue operation data structures using the c programming



```
1 //C program to implement queue data structure
2 #include<stdio.h>
3 #define max 5
4 int main()
5 {
6     int queue[max];
7     int front = -1, rear = -1, ch, element;
8     char ch;
9     printf("Enter the choice (1)=> ENQUEUE, (2)=> DEQUEUE, (3)=> DISPLAY: ");
10    scanf("%d",&ch);
11    do
12    {
13        switch(ch)
14        {
15            case 1:
16                printf("Enter element :");
17                scanf("%d",&element);
18                if(rear==max-1)
19                {
20                    printf("You can't enqueue since the queue is full!");
21                }
22                else if(rear==-1)
23                {
24                    front = 0;
25                    rear = 0;
26                    queue[rear] = element;
27                }
28                else
29                {
30                    rear = rear + 1;
31                    queue[rear] = element;
32                }
33                break;
34            case 2:
35                if(front==-1)
36                {
37                    printf("You can't dequeue since the queue is empty!");
38                }
39                else
40                {
41                    printf("dequeued element = %d",queue[front]);
42                    front = front + 1;
43                }
44                break;
45            case 3:
46                printf("Elements in the queue are : ");
47                if(front==0)
48                {
49                    printf("No elements are present in the queue!");
50                }
51                else
52                {
53                    for(i=front;i<=rear;i++)
54                    {
55                        printf("%d ",queue[i]);
56                    }
57                }
58                break;
59            default:
60                printf("INVALID CHOICE!");
61        }
62        printf("Enter the choice (1)=> ENQUEUE, (2)=> DEQUEUE, (3)=> DISPLAY: ");
63        scanf("%d",&ch);
64    } while(ch!=0);
65    return 0;
66 }
```

Enter element : 40

Enter the choice (1)=> ENQUEUE, (2)=> DEQUEUE, (3)=> DISPLAY: 2
dequeued element = 40
Enter the choice (1)=> ENQUEUE, (2)=> DEQUEUE, (3)=> DISPLAY: 3
Elements in the queue are :
Enter the choice (1)=> ENQUEUE, (2)=> DEQUEUE, (3)=> DISPLAY:

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\rajes\OneDrive\Documents\queue implimentation.exe
- Output Size: 128.6259765625 KiB
- Compilation Time: 0.36s

Line: 57 Col: 2 Sel: 0 Lines: 57 Length: 1202 Insert Done parsing in 0.078 seconds