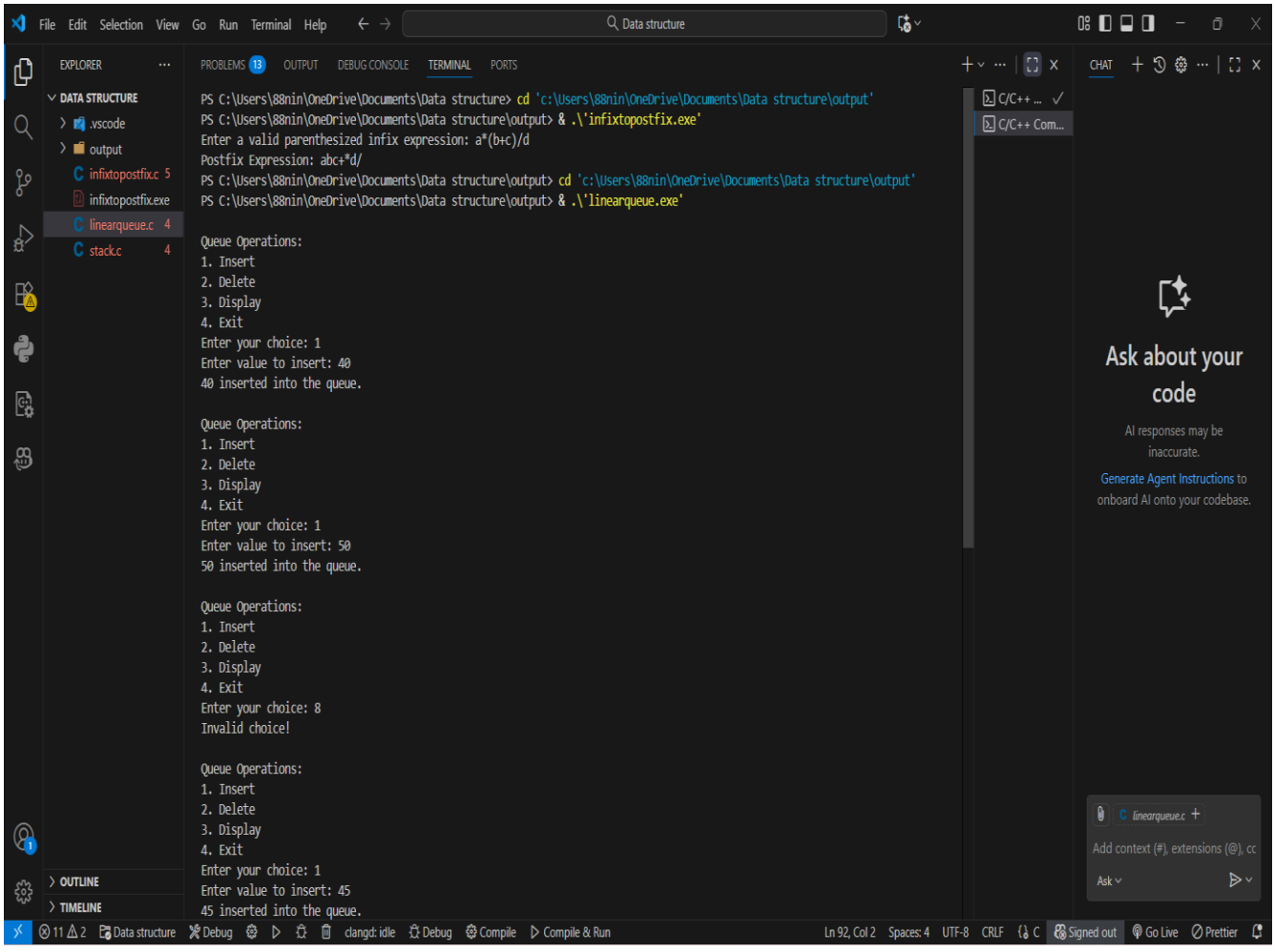
```
/*WAP to simulate the working of a queue of integers using an array. Provide the following
operations: Insert, Delete, Display The program should print appropriate messages for queue
empty and queue overflow conditions*/
#include <stdio.h>
#define MAX 5
int queue[MAX];
int front = -1, rear = -1;

void insert(int value)
{
if (rear == MAX - 1)
{
printf("Queue Overflow!");
}
else
{
if (front == -1)
front = 0;
rear++;
queue[rear] = value;
printf("%d inserted into the queue.\n", value);
}
}

void delete()
{
if (front == -1 || front > rear)
{
```

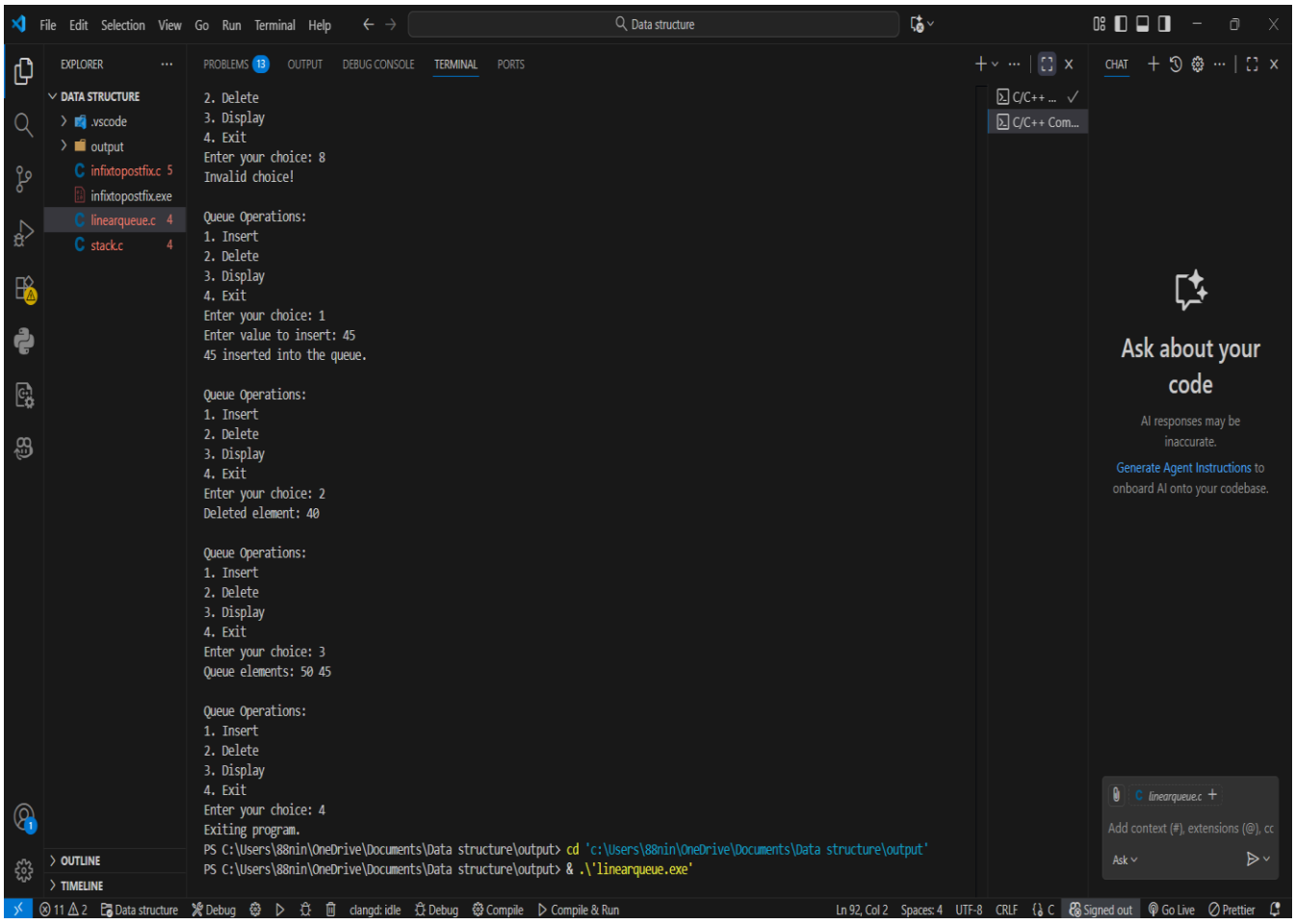
```
printf("Queue Underflow! \n");
}
else
{
printf("Deleted element: %d\n", queue[front]);
front++;
if (front > rear)
{
front = rear = -1;
}
}
}

void display()
{
if (front == -1 || front > rear)
{
printf("Queue is empty.\n");
}
else
{
printf("Queue elements: ");
for (int i = front; i <= rear; i++)
{
printf("%d ", queue[i]);
}
printf("\n");
}
```

```
}

int main()
{
int choice, value;
while (1)
{
printf("\nQueue Operations:\n");
printf("1. Insert\n");
printf("2. Delete\n");
printf("3. Display\n");
printf("4. Exit\n");
printf("Enter your choice: ");
scanf("%d", &choice);
switch (choice)
{
case 1:
printf("Enter value to insert: ");
scanf("%d", &value);
insert(value);
break;
case 2:
delete();
break;
case 3:
display();
break;
case 4:
```

```
printf("Exiting program.\n");
return 0;
default:
printf("Invalid choice!\n");
}
}
return 0;
}
```

Output:-




File Edit Selection View Go Run Terminal Help
Data structure
EXPLORER
DATA STRUCTURE
.vscode
output
infixtopostfix.c 5
infixtopostfix.exe
linearqueue.c 4
stack.c 4
PROBLEMS 13
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
2. Delete
3. Display
4. Exit
Enter your choice: 8
Invalid choice!
Queue Operations:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 1
Enter value to insert: 45
45 inserted into the queue.
Queue Operations:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 2
Deleted element: 40
Queue Operations:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 3
Queue elements: 50 45
Queue Operations:
1. Insert
2. Delete
3. Display
4. Exit
Enter your choice: 4
Exiting program.
PS C:\Users\88nin\OneDrive\Documents\Data structure\output> cd 'c:\Users\88nin\OneDrive\Documents\Data structure\output'
PS C:\Users\88nin\OneDrive\Documents\Data structure\output> & .\'linearqueue.exe'
C/C++ ...
C/C++ Com...
CHAT
Ask about your code
AI responses may be inaccurate.
Generate Agent Instructions to onboard AI onto your codebase.
linearqueue.c
Add context (#), extensions (@), cc
Ask
OUTLINE
TIMELINE
11 2
Data structure
Debug
clangd: idle
Debug
Compile
Compile & Run
Ln 92, Col 2
Spaces: 4
UTF-8
CRLF
C
Signed out
Go Live
Prettier