

ASSIGNMENT-06

/*Coffee Shop Line (Simple Queue):

Arrival: Customers arrive at the coffee shop and stand in line. Order Processing:

The first customer in line gets their order taken, and the barista starts making the coffee.

Serving: Once the first customer is served, they leave the queue, and the next customer in line moves forward to be served. Write a program to implement a simple queue*/

//simple queue implementation using array

```
#include<iostream>
#define SIZE 10
using namespace std;

class coffee_shop {
    int token_queue[SIZE];
    int rear;
    int front;

public:
    coffee_shop()
    {
        rear = -1;
        front = -1;
    }

    int isempty();
    int isfull();
    void enqueue(int t);
    int dequeue();
};

int coffee_shop::isfull()
{
    if (rear == SIZE - 1)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

int coffee_shop::isempty()
{
    if (front == SIZE || front == -1 || front > rear)
    {
        cout << "queue underflow" << endl;
        return 1;
    }
    else {
        return 0;
    }
}

void coffee_shop::enqueue(int t) {
    if (isfull())
    {
        cout << "queue is full" << endl;
    }
    else
    {
```

```

        rear = rear + 1;
        token_queue[rear] = t;
        if (front == -1) front = 0;
    }
}

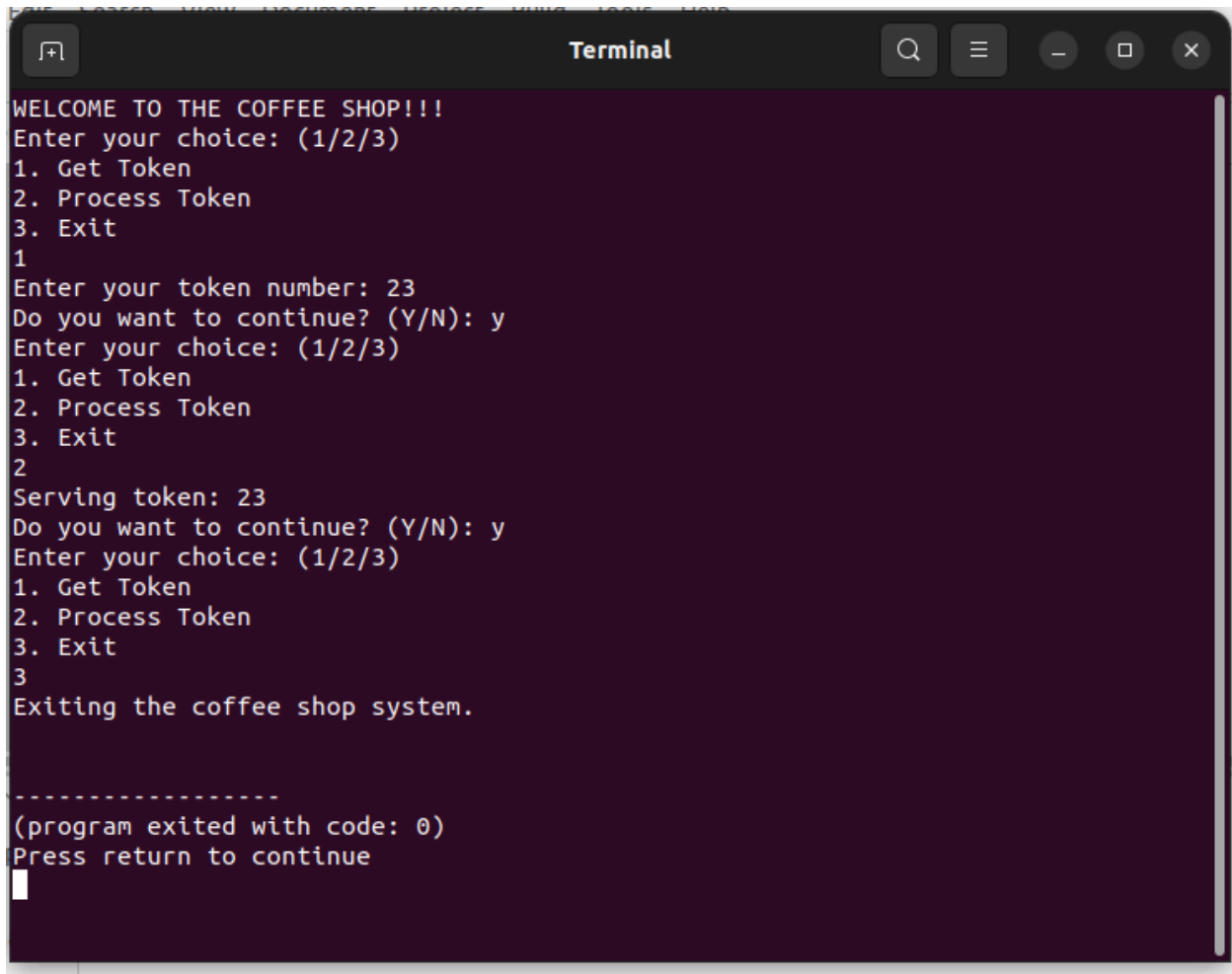
int coffee_shop::dequeue()
{
    if (isempty())
    {
        cout << "queue is empty" << endl;
        return -1;
    }
    else
    {
        int t = token_queue[front];
        front++;
        if (front > rear) {
            front = rear = -1;
        }
        return t;
    }
}

int main()
{
    coffee_shop obj;
    int choice, token;
    char continueChoice;
    cout << "WELCOME TO THE COFFEE SHOP!!! " << endl;
    do {
        cout << "Enter your choice: (1/2/3)" << endl;
        cout << "1. Get Token" << endl;
        cout << "2. Process Token" << endl;
        cout << "3. Exit" << endl;
        cin >> choice;
        switch (choice) {
            case 1:
                cout << "Enter your token number: ";
                cin >> token;
                obj.enqueue(token);
                break;
            case 2:
                {
                    int served = obj.dequeue();
                    if (served != -1)
                        cout << "Serving token: " << served << endl;
                    break;
                }
            case 3:
                cout << "Exiting the coffee shop system." << endl;
                break;
        }
        if (choice != 3) {
            cout << "Do you want to continue? (Y/N): ";
            cin >> continueChoice;
        }
        else {
            continueChoice = 'N';
        }
    } while (continueChoice == 'Y' || continueChoice == 'y');

    return 0;}

```

OUTPUT:

A screenshot of a macOS Terminal window titled "Terminal". The window has a dark background with light-colored text. The text inside shows a program execution for a coffee shop simulation. It starts with a welcome message, followed by a loop where the user enters choices. In the first iteration, choice 1 is entered, a token number 23 is generated, and the user confirms to continue. In the second iteration, choice 2 is entered, and the token 23 is served. In the third iteration, choice 3 is entered, and the program exits. At the bottom, it shows the program exited with code 0 and prompts the user to press return to continue.

```
WELCOME TO THE COFFEE SHOP!!!
Enter your choice: (1/2/3)
1. Get Token
2. Process Token
3. Exit
1
Enter your token number: 23
Do you want to continue? (Y/N): y
Enter your choice: (1/2/3)
1. Get Token
2. Process Token
3. Exit
2
Serving token: 23
Do you want to continue? (Y/N): y
Enter your choice: (1/2/3)
1. Get Token
2. Process Token
3. Exit
3
Exiting the coffee shop system.

-----
(program exited with code: 0)
Press return to continue
█
```

Github: