Name :- Haider Ali

SAP ID :- 53109

Programme :- BSCS 3-1

Q1

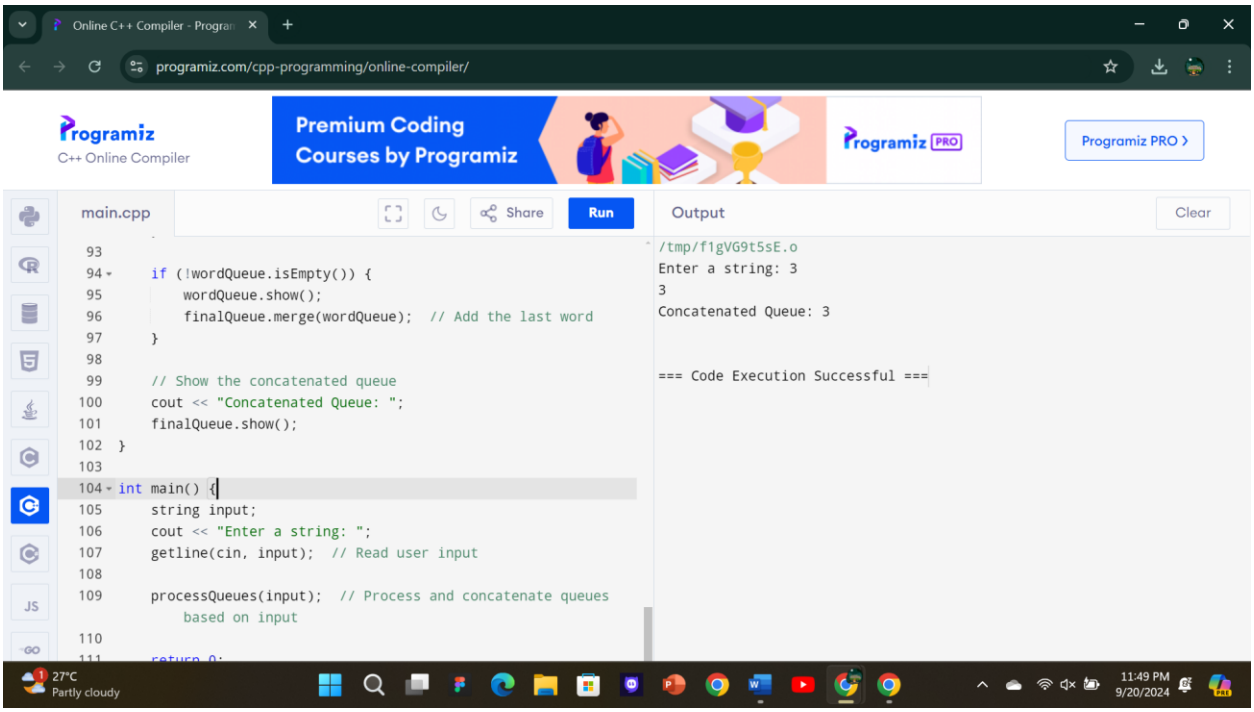
```
#include <iostream>
#include <string>
using namespace std;

class MyQueue {
private:
    char* data;
    int start;
    int end;
    int total;
    int size;

public:
    MyQueue(int s = 100) {
        data = new char[s]; // Allocates memory dynamically, no check for allocation success
        total = s;
        start = 0;
        end = -1;
        size = 0;
```

```
}
~MyQueue() {
    delete[] data;
}
void add(char element) {
    if (size == total) {
        cout << "Queue is full!" << endl;
        return;
    }
    end = (end + 1) % total;
    data[end] = element;
    size++;
}
char remove() {
    if (isEmpty()) {
        cout << "Queue is empty!" << endl;
        return '\0';
    }
    char item = data[start];
    start = (start + 1) % total;
    size--;
    return item;
}
bool isEmpty() {
    return size == 0;
```

```
    }
    void show() {
        if (isEmpty()) {
            cout << "Queue is empty!" << endl;
            return;
        }

        int idx = start;
        for (int i = 0; i < size; i++) {
            cout << data[idx] << " ";
            idx = (idx + 1) % total;
        }
        cout << endl;
    }
    void merge(MyQueue& other) {
        while (!other.isEmpty()) {
            add(other.remove());
        }
    }
};
void processQueues(string input) {
    MyQueue finalQueue(500);
    MyQueue wordQueue;

    for (char ch : input) {
```

```
        if (ch != ' ') {
            wordQueue.add(ch);
        } else {
            wordQueue.show();
            finalQueue.merge(wordQueue);
            wordQueue = MyQueue();
        }
    }

    if (!wordQueue.isEmpty()) {
        wordQueue.show();
        finalQueue.merge(wordQueue);
    }
    cout << "Concatenated Queue: ";
    finalQueue.show();
}

int main() {
    string input;
    cout << "Enter a string: ";
    getline(cin, input);

    processQueues(input);

    return 0;
```

}


Online C++ Compiler - Progra
programiz.com/cpp-programming/online-compiler/
Programiz
C++ Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO
main.cpp
Share
Run
Output
Clear
93
94 if (!wordQueue.isEmpty()) {
95 wordQueue.show();
96 finalQueue.merge(wordQueue); // Add the last word
97 }
98
99 // Show the concatenated queue
100 cout << "Concatenated Queue: ";
101 finalQueue.show();
102 }
103
104 int main() {
105 string input;
106 cout << "Enter a string: ";
107 getline(cin, input); // Read user input
108
109 processQueues(input); // Process and concatenate queues based on input
110
/tmp/f1gVG9t5sE.o
Enter a string: 3
3
Concatenated Queue: 3
=== Code Execution Successful ===
27°C
Partly cloudy
11:49 PM
9/20/2024