#include <iostream>

using namespace std;

class Synchronization {

int a[10]; // Increased buffer size to 10

int mutex;

int empty;

int full;

int in;

int out;

void wait(int &x) {

if (x > 0) x--;

}

void signal(int &x) {

x++;

}

public:

Synchronization() : mutex(1), empty(10), full(0), in(0), out(0) {}

void producer() {

if (empty > 0 && mutex == 1) {

wait(empty);

wait(mutex);

cout << "Data to be produced: ";

int data;

cin >> data;

a[in] = data;

in = (in + 1) % 10; // Update for new buffer size

signal(mutex);

signal(full);

} else {

cout << "Buffer is full, cannot produce!" << endl;

}

}

void consumer() {

if (full > 0 && mutex == 1) {

wait(full);

wait(mutex);

cout << "Data consumed is: " << a[out] << endl; // Show consumed data

out = (out + 1) % 10; // Update for new buffer size

signal(mutex);

signal(empty);

} else {

cout << "Buffer is empty, cannot consume!" << endl;

}

}

};

int main() {

int fchoice;

Synchronization s;

do {

cout << "1. Producer\n2. Consumer\n3. Exit" << endl;

cout << "Enter your choice: ";

cin >> fchoice;

switch (fchoice) {

case 1: s.producer(); break;

case 2: s.consumer(); break;

case 3: break;

default: cout << "Invalid choice!" << endl; break;

}

} while (fchoice != 3);

return 0;

}

Output:

admin1@413-25:~$ g++ mutex.cpp

admin1@413-25:~$ ./a.out

1. Producer

2. Consumer

3. Exit

Enter your choice: 1

Data to be produced: 1101

1. Producer

2. Consumer

3. Exit

Enter your choice: 2

Data consumed is: 1101

1. Producer

2. Consumer

3. Exit

Enter your choice: 2

Buffer is empty, cannot consume!

1. Producer

2. Consumer

3. Exit

Enter your choice: 3

admin1@413-25:~$