

**Code**

import java.util.LinkedList;

import java.util.Queue;

import java.util.Scanner;

class BoundedBuffer {

private final int capacity;

private final Queue<Integer> buffer;

private int mutex;

private int full;

private int empty;

private int x;

public BoundedBuffer(int capacity) {

this.capacity = capacity;

this.buffer = new LinkedList<>();

this.mutex = 1;

this.full = 0;

this.empty = capacity;

this.x = 0;

}

private void wait(int[] semaphore) {

while (semaphore[0] <= 0) {}

semaphore[0]--;

}

private void signal(int[] semaphore) {

semaphore[0]++;

}

public void produce() {

wait(new int[]{empty});

wait(new int[]{mutex});

x++;

buffer.add(x);

System.out.println("Produced: " + x);

signal(new int[]{mutex});

signal(new int[]{full});

}

public void consume() {

wait(new int[]{full});

wait(new int[]{mutex});

int item = buffer.poll();

System.out.println("Consumed: " + item);

signal(new int[]{mutex});

signal(new int[]{empty});

}

}

public class ProducerConsumerProblem {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the capacity of the bounded buffer: ");

int capacity = scanner.nextInt();

BoundedBuffer boundedBuffer = new BoundedBuffer(capacity);

while (true) {

System.out.println("\nMenu:");

System.out.println("1. Produce Item");

System.out.println("2. Consume Item");

System.out.println("3. Exit");

System.out.print("Choose an option: ");

int choice = scanner.nextInt();

switch (choice) {

case 1:

boundedBuffer.produce();

break;

case 2:

boundedBuffer.consume();

break;

case 3:

System.out.println("Exiting...");

scanner.close();

return;

default:

System.out.println("Invalid choice. Please try again.");

}

}

}

}

**Output:**

Enter the capacity of the bounded buffer: 45

Menu:

1. Produce Item

2. Consume Item

3. Exit

Choose an option: 1

Produced: 1

Menu:

1. Produce Item

2. Consume Item

3. Exit

Choose an option: 1

Produced: 2

Menu:

1. Produce Item

2. Consume Item

3. Exit

Choose an option: 1

Produced: 3

Menu:

1. Produce Item

2. Consume Item

3. Exit