Assignment 1: Investor-Producer Synchronization Problem

Version 1.0

Dr. Mosaddek Hossain Kamal Tushar and Dr. Mamun Rashid

September 30, 2019

Due Date: 15 Days from the uploaded date **September 23, 2019**

Contents

1 Investor-Producer Synchronization Problem **2**

2 The components of the system **2**

2.1 Customer thread . . . 2

2.1.1 order_item() . . . 3

2.1.2 consume() . . . 3

2.1.3 yield() . . . 3

2.1.4 end_shopping() . . . 3

2.2 Producer thread . . . 3

2.2.1 take_order() . . . 3

2.2.2 calculate_loan_amount() . . . 3

2.2.3 loan_request() . . . 3

2.2.4 produce_item() . . . 4

2.2.5 serve_order() . . . 4

2.2.6 loan˙reimburse() . . . 4

2.2.7 Going home . . . 4

3 Investor **4**

3.1 Bank Records . . . 4

3.2 Statistics . . . 4

4 Testing **4**

4.1 Current Status of the Code . . . 4

5 What to Submit **5**
5.1 Q&A 6
5.2 Demonstrate your design and solution 6

1 Investor-Producer Synchronization Problem

In this assignment, the student solves an investor-producer critical section problem. The assignment gives the student an opportunity and insight view to write one fairly straightforward concurrent programs and get a more detailed understanding of how to use concurrency mechanisms to address the problems.

In this problem, a customer requests the producer to supply some items to accomplish their every day's need; by accepting the request, the producer loan money from the bank and use to produce the items. Bank finance the business and get the money back from the producer with a service charge. The producer decides the item price combining bank service charges and profit.

Customer pays before the consumption of the requested items. Then, they finish shopping and back home and sleep. On the other hand, bank and producer evaluate the business at the end of the day. However, investor (bank) and producer discovered that the balance sheet is inconsistent. Therefore, they call third-year students of CSE, the University of Dhaka to employ them to analyze the whole system to mitigate the discrepancies in the balance sheet. The students found serious discrepancies due to the mismanagement of demand, supply, and financial transactions.

The investor additionally invites students to produce a design document and resolve the puzzle if there is any. The CSE, DU students, first trying to understand the process and data-flow of the system, and comprehend that the entire process is indeed a mess. The same order is serving by multiple producers, and the bank financial records of the investment are in a horrible condition. They reviewed the business process and ascertained that the system consists of several entities such as customer and producer threads, bank records, that are enumerated underneath.

2 The components of the system

The system deploys a multi-threading mechanism in the OS161 kernel mode with numerous functionalities which are as follows.

2.1 Customer thread

The customer thread simulates customer activities which order items and waiting. As soon as the item is being available by the producer, then they pay the price and enjoy the commodities. After a while, they repeat the process for a few times and enjoy the product. At the end of

the shopping, they go back home and resting. The customer thread contains the following functions:

2.1.1 order_item()

Customer chooses different types of item, their quantities, and place the order electronically (by smart devices) and wait until the items are available. As soon as the item is available, they pay by cash or credit card.

2.1.2 consume()

As soon as the payment is clear, they enjoy the items among family and friends, release the service bucket, and record the spending. They eat, wear, play, send gifts to friends and relations.

2.1.3 yield()

Customers spend the astonishing occasion with their family and friends before willing to buy other items.

2.1.4 end_shopping()

At the end of the day, customers are tired and require to go back home for sleep. Presumably, some others day, they might back for shopping, however, for today, they finish shopping.

2.2 Producer thread

In the investor-producer system, the producer has a long term agreement with the bank for lending any amount of money because of their good behavior (loan repayment). Producer thread acquires the customer's orders from an electronic ordering system and spends money from the bank to produce the items. They serve the customers, determine the item price, and collect the money from the customers. Then, the producer repays the loan amount and service charge and waiting for the next order. The thread hosted many functions such as

2.2.1 take_order()

A produce wait in this function until any order is available. As soon as a customer gives a request for purchasing items, producers take (accept) the order for the process. Each of the producers can produce one-third of the total number of item types at a time.

2.2.2 calculate_loan_amount()

After accepting one or more orders producer calculate the expense for the production of the ordered items and proceeds for the bank loan.

2.2.3 loan_request()

The producer request for loan and bank disburse the fund according to the terms of the agreement between the bank and business. For simplicity, currently, the bank accepts any loan amount requested by the producer.

2.2.4 produce_item()

After investing, the producer uses the loan money for purchasing raw materials, staff and labor salaries for producing the item.

2.2.5 serve_order()

. As quickly as the item is ready, the producer serves the order according to the customers' demand. This time producer adds the profit and bank service charge to the product cost and collects the money from the customers.

2.2.6 loan˙reimburse()

Producer repays the loan money with service charge and updates its total profit.

2.2.7 Going home

When no more customers are available or not willing to shop anymore, than the producer clean up and close the shop. After closing the shop, the producer goes home for rest.

3 Investor

3.1 Bank Records

The investor banks maintain bank records for bookkeeping, track the producer loan amount, accumulated loan, remaining cash, and service charge.

3.2 Statistics

At the end of the day, the investor print the bank statistics or balance sheet to find the financial standing of the bank and business.

4 Testing

The investor planning to test the new design and code of the suggested solution of the investor-producer synchronization problem. To do so, the investor requested CSE faculty member to design a test suit to test the accuracy and performance of the solution. The major design must be efficient, accurate, and deploy maximum parallelism of the proposed solution.

4.1 Current Status of the Code

After a successful compilation, you can test your solution as,

```
[user@csewk112 root] sys161 kernel-ASST "1b"
```

or you can boot the kernel and in the menu prompt type *1b*. Current status of the output is,

```
sys161: System/161 release 2.0.8, compiled Aug 26 2018 15:26:35

OS/161 base system version 2.0.3
Copyright (c) 2000, 2001-2005, 2008-2011, 2013, 2014
President and Fellows of Harvard College.  All rights reserved.

Put-your-group-name-here's system version 0 (ASST1 #1)

1888k physical memory available
Device probe...
lamebus0 (system main bus)
emu0 at lamebus0
ltrace0 at lamebus0
ltimer0 at lamebus0
beep0 at ltimer0
rtclock0 at ltimer0
lrandom0 at lamebus0
random0 at lrandom0
lhd0 at lamebus0
lhd1 at lamebus0
lser0 at lamebus0
con0 at lser0

cpu0: MIPS/161 (System/161 2.x) features 0x0
OS/161 kernel: 1b
panic: You need to write some code!!!!
sys161: trace: software-requested debugger stop
sys161: Waiting for debugger connection...
```

5 What to Submit

Student must prepare and submit a design document which illustrates the design of the solution, question answer, and demonstrate their solution in front of the test team. Test team tests and

grades the solution according to the correctness and performance of the solution. Further, to help the design and coding, the investor provides a sample test suit.

5.1 Q&A

Each of the students must submit the Q&A that is given in the previous practice (asst1p) assignment doc.

5.2 Demonstrate your design and solution

All the students in a group must present the design and demonstrate the solution. The student must submit the design document and Q&A individually.