

Curtin University – Department of Computing

Assignment Cover Sheet / Declaration of Originality

Complete this form if/as directed by your unit coordinator, lecturer or the assignment specification.

| | | | |
|------------------------------|--|-------------------|--|
| Last name: | | Student ID: | |
| Other name(s): | | | |
| Unit name: | | Unit ID: | |
| Lecturer / unit coordinator: | | Tutor: | |
| Date of submission: | | Which assignment? | (Leave blank if the unit has only one assignment.) |

I declare that:

- The above information is complete and accurate.
- The work I am submitting is *entirely my own*, except where clearly indicated otherwise and correctly referenced.
- I have taken (and will continue to take) all reasonable steps to ensure my work is *not accessible* to any other students who may gain unfair advantage from it.
- I have *not previously submitted* this work for any other unit, whether at Curtin University or elsewhere, or for prior attempts at this unit, except where clearly indicated otherwise.

I understand that:

- Plagiarism and collusion are dishonest, and unfair to all other students.
- Detection of plagiarism and collusion may be done manually or by using tools (such as Turnitin).
- If I plagiarise or collude, I risk failing the unit with a grade of ANN ("Result Annulled due to Academic Misconduct"), which will remain permanently on my academic record. I also risk termination from my course and other penalties.
- Even with correct referencing, my submission will only be marked according to what I have done myself, specifically for this assessment. I cannot re-use the work of others, or my own previously submitted work, in order to fulfil the assessment requirements.
- It is my responsibility to ensure that my submission is complete, correct and not corrupted.

Signature:  _____ Date of signature: _____

(By submitting this form, you indicate that you agree with all the above text.)

OPERATING SYSTEMS ASSIGNMENT 1

AUTHOR

KARTHIK KARAVATT
20619965

CURTIN UNIVERSITY

Contents

| | |
|---|----|
| README | 2 |
| Structure of assignment directories | 2 |
| Running the code | 2 |
| Code | 3 |
| main.c | 3 |
| assignmentMethods.c | 5 |
| assignmentMethods.h | 10 |
| linkedList.c | 11 |
| linkedList.h | 14 |
| makefile | 15 |
| Synchronization Discussion | 16 |
| Tests and inconsistencies | 20 |
| Testing | 20 |
| Sample input and output | 21 |
| Test 1 | 23 |
| Test 2 | 45 |
| Test 3 | 67 |

README

2023 Operating systems Assignment

Author : Karthik Karavatt

StudentID: 20619965

Structure of assignment directories

- assignment
 - contains all assignment code
- misc
 - contains miscellaneous files used for the assignment

Running the code

- Enter code directory
- run ‘make’ command
- enter ‘./main [m] [t_c] [t_w] [t_d] [t_i]’
- the [] sections should be replaced with positive integers

Code

main.c

```
//Name: Karthik Karavatt
//StudentID: 20619965
#include "assignmentMethods.h"
#include "linkedList.h"
#include <pthread.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <unistd.h>

int main(int argc, char *argv[]) {
    //global variable from assignmentMethods.c
    extern pthread_mutex_t writeToLog;
    extern pthread_mutex_t listLock;
    extern pthread_mutex_t fileLock;
    extern pthread_cond_t cond;
    extern pthread_cond_t queueFull;
    extern pthread_cond_t continueOperation;
    //command line arguments
    int m = atoi(argv[1]);
    int t_c = atoi(argv[2]);
    int t_w = atoi(argv[3]);
    int t_d = atoi(argv[4]);
    int t_i = atoi(argv[5]);
    // M has to be greater than 0 for the program to work
    // Otherwise it will not make sense
    if(m <= 0){
        exit(0);
    }
    pthread_t id, t1, t2, t3, t4;
    CustomerArgs args;
    Teller teller1, teller2, teller3, teller4;
    //using linked list made in USP to act as a queueFull
    //The only methods that will be used is insert last and remove first
    //Therefore it is equivalent to a queue
    LinkedList *c_queue = createList();
    //assigning tellers variables
    teller1.id = "1";
    teller1.t_i = t_i;
    teller1.m = m;
    teller1.t_d = t_d;
    teller1.t_w = t_w;
    teller1.list = c_queue;
    teller1.served = 0;
    teller2.id = "2";
    teller2.m = m;
    teller2.t_i = t_i;
    teller2.t_d = t_d;
    teller2.t_w = t_w;
    teller2.list = c_queue;
    teller2.served = 0;
    teller3.id = "3";
```

```

teller3.m = m;
teller3.t_i = t_i;
teller3.t_d = t_d;
teller3.t_w = t_w;
teller3.list = c_queue;
teller3.served = 0;
teller4.id = "4";
teller4.m = m;
teller4.t_i = t_i;
teller4.t_d = t_d;
teller4.t_w = t_w;
teller4.list = c_queue;
teller4.served = 0;
//arguments for the customer thread
args.list = c_queue;
args.t_c = t_c;
args.m = m;
// thread initialization
pthread_mutex_init(&listLock, NULL);
pthread_mutex_init(&writeToLog, NULL);
pthread_mutex_init(&fileLock, NULL);
pthread_cond_init(&cond, NULL);
pthread_cond_init(&queueFull, NULL);
//thread creation
pthread_create(&id, NULL, customer, (void *)&args);
pthread_create(&t1, NULL, teller, (void *)&teller1);
pthread_create(&t2, NULL, teller, (void *)&teller2);
pthread_create(&t3, NULL, teller, (void *)&teller3);
pthread_create(&t4, NULL, teller, (void *)&teller4);
//thread join
pthread_join(id, NULL);
pthread_join(t1, NULL);
pthread_join(t2, NULL);
pthread_join(t3, NULL);
pthread_join(t4, NULL);
// freeing resources
freeList(c_queue);
pthread_mutex_destroy(&listLock);
pthread_mutex_destroy(&writeToLog);
pthread_cond_destroy(&cond);
pthread_cond_destroy(&queueFull);
return EXIT_SUCCESS;
}

```

assignmentMethods.c

```
//Name: Karthik Karavatt
//StudentID: 20619965
#include "assignmentMethods.h"
#include "linkedList.h"
#include <pthread.h>
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/wait.h>
#include <time.h>
#include <unistd.h>

// Globals
pthread_mutex_t writeToLog;
pthread_mutex_t listLock;
pthread_mutex_t fileLock;
pthread_cond_t cond;
pthread_cond_t queueFull;
int fileread = 0; // Indicates if all customers have been dealt with
int served[4]; // array holding info about how many customers a teller has served
int tellersLeft = 4; // how many tellers are still running (not terminated)
// logs a string to the log file
void logToFile(char *message) {
    FILE *file;
    file = fopen("r_log", "a");
    fprintf(file, "%s", message);
    fflush(file);
    fclose(file);
}
// log some common info about the customer
void logCustomer(char *customerString, char *serviceString, char *onlyTime) {
    logToFile("-----\n");
    logToFile(customerString);
    logToFile(": ");
    logToFile(serviceString);
    logToFile("\n");
    logToFile("Arrival Time: ");
    logToFile(onlyTime);
    logToFile("-----\n");
}
// add customer from the file to the queue
void addCustomer(linkedList *list, char line[], int t_c) {
    time_t curTime;
    struct tm *timeString;
    char *customerString;
    char *serviceString;
    char *splitString;
    Customer *customer = malloc(sizeof(Customer));
    int index = 0;
    // split a line from the file into sub strings delimiter is the " "
    splitString = strtok(line, " \n\0");
    // iterate through split string
    while (splitString != NULL) {
        if (index == 0) {
```

```

        customerString = splitString;
        customer->number = (char *)malloc(strlen(customerString) + 1);
        strcpy(customer->number, customerString);
    } else if (index == 1) {
        serviceString = splitString;
        serviceString[strlen(serviceString, "\n")] = 0;
        customer->service = *splitString;
    }
    index++;
    splitString = strtok(NULL, " ");
}
time(&curTime);
timeString = localtime(&curTime);
char *onlyTime = (char *)malloc((101) * sizeof(char));
sprintf(onlyTime, "%d:%d:%d\n", timeString->tm_hour, timeString->tm_min,
        timeString->tm_sec);
customer->arrivalTime = onlyTime;
insertLast(list, (void *)customer);
// lock the operation to write to the file
pthread_mutex_lock(&writeToLog);
logCustomer(customerString, serviceString, onlyTime);
// unlock the operation to write to the file
pthread_mutex_unlock(&writeToLog);
}

// customer function
void *customer(void *data) {
    CustomerArgs *args = (CustomerArgs *)data;
    LinkedList *list = args->list;
    int t_c = args->t_c;
    int m = args->m;
    FILE *fptr;
    char line[50];
    fptr = fopen("c_file", "r");
    // iterates through the whole file
    while (fgets(line, sizeof(line), fptr)) {
        // lock the access to the linked list
        pthread_mutex_lock(&listLock);
        if (list->size == m) {
            // wait until the queue is empty before adding more customers
            pthread_cond_wait(&queueFull, &listLock);
        }
        addCustomer(list, line, t_c);
        // signal that customer has been added
        pthread_cond_signal(&cond);
        // unlock the access to the queue
        pthread_mutex_unlock(&listLock);
        // sleeps
        sleep(t_c);
        // locks the list again
    }
    fclose(fptr);
    pthread_mutex_lock(&listLock);
    // if the whole file has been read, customer will signal the teller until
    // queue is empty this is so there is no dead lock between the tellers Because
    // they are waiting for the signal from the customer

```



```

while (list->size != 0) {
    pthread_cond_signal(&cond);
    pthread_cond_wait(&queueFull, &listLock);
}
pthread_mutex_unlock(&listLock);
// changing the file read value indicates to tellers they should terminate
pthread_mutex_lock(&fileLock);
fileread = 1;
pthread_mutex_unlock(&fileLock);
// broadcast signals all tellers that they should terminate
pthread_cond_broadcast(&cond);
return EXIT_SUCCESS;
}

// free all customers left in queue
void *teller(void *data) {
    time_t completeionT, responseT;
    struct tm *completeionString, *responseString;
    Teller *teller = (Teller *)data;
    LinkedList *list = teller->list;
    int m = teller->m;
    pthread_mutex_lock(&fileLock);
    while (fileread == 0) {
        pthread_mutex_unlock(&fileLock);
        pthread_mutex_lock(&listLock);
        if (list->size == 0) {
            // wait for a signal from the customer when the queue is full
            pthread_cond_wait(&cond, &listLock);
        } else {
            Customer *customer = (Customer *)removeFirst(list);
            // signal to the customer function that a customer has been removed
            pthread_cond_signal(&queueFull);
            pthread_mutex_unlock(&listLock);
            teller->served += 1;
            served[atoi(teller->id) - 1] = teller->served;
            responseT = time(&responseT);
            responseString = localtime(&responseT);
            char responseTime[100];
            sprintf(responseTime, "%d:%d:%d\n", responseString->tm_hour,
                    responseString->tm_min, responseString->tm_sec);
            pthread_mutex_lock(&writeToLog);
            logToFile("Teller: ");
            logToFile(teller->id);
            logToFile("\n");
            logToFile("Customer: ");
            logToFile(customer->number);
            logToFile("\n");
            logToFile("Arrival time: ");
            logToFile(customer->arivalTime);
            logToFile("Response time: ");
            logToFile(responseTime);
            // unlocks when customer is being serviced
            pthread_mutex_unlock(&writeToLog);
            switch (customer->service) {
                case 'W':
                    sleep(teller->t_w);

```

```

        break;
    case 'D':
        sleep(teller->t_d);
        break;
    case 'I':
        sleep(teller->t_i);
        break;
}
char completeionTime[100];
completeionT = time(&responseT);
completeionString = localtime(&responseT);
sprintf(completeionTime, "%d:%d:%d\n", completeionString->tm_hour,
        completeionString->tm_min, completeionString->tm_sec);
pthread_mutex_lock(&writeToLog);
logToFile("Teller: ");
logToFile(teller->id);
logToFile("\n");
logToFile("Customer: ");
logToFile(customer->number);
logToFile("\n");
logToFile("Arrival time: ");
logToFile(customer->arivalTime);
logToFile("Completion time: ");
logToFile(completeionTime);
pthread_mutex_unlock(&writeToLog);
// Accessing fileread vairable is locked now
pthread_mutex_lock(&fileLock);
// locks it again
pthread_mutex_lock(&listLock);
free(customer->arivalTime);
free(customer->number);
free(customer);
}
// check if the teller is the last one left
// list lock is still enabled here so checking tellers left is okay
if (tellersLeft == 1) {
    char served1[100];
    char served2[100];
    char served3[100];
    char served4[100];
    pthread_mutex_lock(&writeToLog);
    logToFile("Teller 1 serverd: ");
    sprintf(served1, "%d\n", served[0]);
    logToFile(served1);
    logToFile("Teller 2 serverd: ");
    sprintf(served2, "%d\n", served[1]);
    logToFile(served2);
    logToFile("Teller 3 serverd: ");
    sprintf(served3, "%d\n", served[2]);
    logToFile(served3);
    logToFile("Teller 4 serverd: ");
    sprintf(served4, "%d\n", served[3]);
    logToFile(served4);
    pthread_mutex_unlock(&writeToLog);
}
pthread_mutex_unlock(&listLock);

```

```
}  
// decremetns this value when a teller terminates  
tellersLeft--;  
pthread_mutex_unlock(&listLock);  
pthread_mutex_unlock(&fileLock);  
return EXIT_SUCCESS;  
}
```

assignmentMethods.h

```
#pragma once
#include "linkedList.h"

typedef struct Customer {
    char* number;
    char service;
    char* arivalTime;
} Customer;

typedef struct CustomerArgs {
    LinkedList *list;
    int t_c;
    int m;
} CustomerArgs;

typedef struct Teller{
    LinkedList* list;
    char* id;
    int m;
    int t_w;
    int t_d;
    int t_i;
    int served;
}Teller;

void logToFile(char *message);
void printCustomer(void *data);
void *customer(void *data);
void *teller(void* data);
```

linkedList.c

```
/*author Karthik Karavatt
 * StudentID 20619965 */
/* This linked list will act like a queue*/
// Created when doing a USP practical
#include "linkedList.h"
#include <stdio.h>
#include <stdlib.h>

LinkedList *createList() {
    LinkedList *list = (LinkedList *)malloc(sizeof(LinkedList));
    list->head = NULL;
    list->tail = NULL;
    list->size = 0;
    return list;
}

void insertFirst(LinkedList *list, void *data) {
    Node *node = (Node *)malloc(sizeof(Node));
    node->data = data;
    node->next = NULL;
    /* If list is empty */
    if (list->head == NULL) {
        node->next = list->tail;
        list->head = node;
    }
    /* if the list has one item*/
    else if (list->head->next == NULL) {
        Node *temp = list->head;
        temp->before = node;
        list->head = node;
        node->next = temp;
        list->tail = temp;
    }
    else {
        Node *temp = list->head;
        temp->before = node;
        list->head = node;
        node->next = temp;
    }
    list->size++;
}

void insertLast(LinkedList *list, void *data) {
    Node *currentTail;
    Node *node = (Node *)malloc(sizeof(Node));
    node->data = data;
    node->next = NULL;
    /*if list is empty */
    if (list->head == NULL) {
        list->head = node;
        list->head->next = list->tail;
    }
    else {
        currentTail = list->tail;
        /*if there is only one node */
        if (currentTail == NULL) {
```

```

        list->head->next = node;
        node->before = list->head;
        list->tail = node;
    } else {
        currentTail->next = node;
        node->before = currentTail;
        list->tail = node;
    }
}
list->size++;
}

void *removeFirst(LinkedList *list) {
    void *data;
    Node *node;
    /*check if list is empty*/
    if (list->head != NULL) {
        node = list->head;
        data = node->data;
        /*check if there is only one node*/
        if (list->tail == NULL) {
            list->head = NULL;
        }
        /*check if there is only two node*/
        else if (list->head->next == list->tail) {
            list->head = list->tail;
            list->head->before = NULL;
            list->tail = NULL;
        } else {
            data = list->head->data;
            list->head = list->head->next;
            list->head->before = NULL;
        }
        free(node);
        list->size--;
    }
    return data;
}

void *removeLast(LinkedList *list) {
    void *data;
    Node *node;
    /* check if list is empty */
    if (list->head != NULL) {
        /* check if there is more than one node */
        if (list->tail != NULL) {
            node = list->tail;
            data = node->data;
            node->before->next = NULL;
            /* check if there are only two node */
            if (node->before == list->head) {
                list->tail = NULL;
            } else {
                list->tail = node->before;
            }
            node = NULL;
        }
        /* If there is only one node */

```

```

        else {
            node = list->head;
            data = node->data;
            list->head = NULL;
        }
        free(node);
        list->size--;
    }
    return data;
}

void printList(LinkedList *list, listFunc func) {
    Node *node = list->head;
    if (node != NULL) {
        while (node != NULL) {
            func(node->data);
            node = node->next;
        }
        node = NULL;
    }
}

void freeNode(LinkedList *list) {
    Node *curNode = list->head;
    Node *nextNode;
    while (curNode != NULL) {
        nextNode = curNode->next;
        free(curNode);
        curNode = nextNode;
        list->size--;
    }
}

void freeList(LinkedList *list) {
    freeNode(list);
    free(list);
}

```

linkedList.h

```
#pragma once
typedef struct Node
{
    void* data;
    struct Node* before;
    struct Node* next;
}Node;
typedef struct LinkedList
{
    Node* head;
    Node* tail;
    int size;
}LinkedList;
typedef void(*listFunc)(void* data);
LinkedList* createList(void);
void insertFirst(LinkedList* list, void* data);
void insertLast(LinkedList* list, void* data);
void* removeFirst(LinkedList* list);
void* removeLast(LinkedList* list);
void printList(LinkedList* list, listFunc func);
void freeList(LinkedList* list);
```


makefile

```
CC = gcc
LD = gcc
CFLAGS = -g -pthread
LFLAGS = -lm -s
OBJ = main.o linkedList.o assignmentMethods.o
EXEC = main
$(EXEC): $(OBJ)
    $(LD) $(CFLAGS) $(OBJ) -o $(EXEC)

main.o: main.c linkedList.h assignmentMethods.h
    $(CC) -c main.c $(CFLAGS)

linkedList.o: linkedList.c
    $(CC) -c linkedList.c $(CFLAGS)

assignmentMethods.o: assignmentMethods.c linkedList.h
    $(CC) -c assignmentMethods.c $(CFLAGS)

clean:
    $(RM) $(EXEC) $(OBJ)

val:
    valgrind --leak-check=full -s ./${EXEC} 3 1 1 1 1
run:
    ./main 3 1 1 1 1
gdb:
    gdb --args main 2 1 1 1 1

# DO NOT DELETE
```

Synchronization Discussion

To understand how the synchronization works, let's see what variables are being shared. The `c_queue` is the main-shared variable in the program. It is shared by the following functions:

- customer
- teller

The customer runs on one thread, whereas, the teller runs on 4 threads. So 5 threads need access to the `c_queue`.

The `r_log` file is also a shared variable in the program. It is represented as the file variable in the `logToFile` function. This is also shared by the customer and teller functions. Therefore, 5 threads require access to the `r_log` file.

The `fileread` variable located in `assignmentMethods.c` is also shared among all 5 threads. Although, only the customer thread can modify it, teller threads can only read it. This variable indicates when the `c_file` has been fully read.

The `tellersLeft` integer, is another shared variable only accessed and modified by teller threads. It indicates how many tellers are still executing.

The `served` array, is shared by all teller threads. This indicates the number of customers served by each teller thread.

To achieve synchronization for the `c_queue`, the mutex lock `listLock` was used. If a thread needed to access or modify the `c_queue`, the lock must be enabled first. Then after it is used, the lock must be released. For example in the customer function

`assignmentMethods.c`, customer

```
pthread_mutex_lock(&listLock);
if (list->size == m) {
    // wait until the queue is empty before adding more customers
    pthread_cond_wait(&queueFull, &listLock);
}
addCustomer(list, line, t_c);
// signal that a customer has been added
pthread_cond_signal(&cond);
// unlock the access to the queue
pthread_mutex_unlock(&listLock);
```

When the customer thread wants to check the size of the `c_queue` (called `list` in this function), it enables the `listLock` by calling `pthread_mutex_lock(listLock)`. This lock is also enabled when a customer is added to the `c_queue`, using the `addCustomer()` function. After which it is unlocked using the `pthread_mutex_unlock(&listLock)` method.

This example also highlights another way synchronization is achieved, through the use of `pthread_cond_signal()` and `pthread_cond_wait()`. When `list->size == m` it indicates that the `c_queue` is full. When this situation occurs, the customer thread must wait until it gets the `queueFull` signal. The `queueFull` signal is emitted by the teller thread when a customer is removed from the list.

assignmentMethods.c, teller

```
Customer *customer = (Customer *)removeFirst(list);  
// signal to the customer function that a customer has been removed  
pthread_cond_signal(&queueFull);
```

When a customer is removed from the `c_queue`, the signal will be emitted, this stops the customer thread from waiting, and it continues execution. Similarly, the teller thread also will be put into a waiting state when the `c_queue` is full.

assignmentMethods.c, teller

```
if (list->size == 0) {  
// wait for a signal from the customer when the queue is full  
pthread_cond_wait(&cond, &listLock);  
}
```

The teller thread can break out of the wait when the customer thread adds a customer to the list. It is important to note that the `listLock` is still locked before the `pthread_cond_wait()` is called. But after it is called, the mutex lock is unlocked until the signal is received from the customer thread.

assignmentMethods.c, customer

```
addCustomer(list, line, t_c);  
//signal that customer has been added  
pthread_cond_signal(&cond);
```

When a customer is added to the `c_queue`, the `cond` signal will be emitted, which breaks the wait condition of the teller thread, as the queue is not empty. This prevents deadlocks from occurring as there can never be situation where the `c_queue` is both empty and full at the same time. And the size value read by each thread is always accurate as a mutex lock is enabled before accessing them. The list lock is also disabled before the teller operations begin so, when a teller thread is sleeping, other threads can access the `c_queue`. This improves the concurrency of the program.

The `fileLock` mutex lock is used to protect the `fileread` variable. This variable is used to terminate all tellers when all customers have been served. When the customer thread has finished reading the `c_file`, it changes the `fileread` variable to 1.

assignmentMethods.c, customer

```
pthread_mutex_lock(&fileLock);  
fileread = 1;  
pthread_mutex_unlock(&fileLock);
```

This signifies to the tellers, that they should terminate. In the teller thread, the while loop relies on the `fileread` variable being 0. Now that the while loop is broken, the teller will terminate.

assignmentMethods.c, teller

```
pthread_mutex_lock(&fileLock);
while (fileread == 0) {
    pthread_mutex_unlock(&fileLock);
    ...
    ...
    ...
    pthread_mutex_lock(&fileLock);
    // locks it again
    pthread_mutex_lock(&listLock);
    free(customer->arrivalTime);
    free(customer->number);
    free(customer);
}
```

After the fileread has been checked, fileLock is opened, at the end of the while loop, inside it, it is locked again as fileread variable must be accessed again. But this creates a problem, where the file has been read, but there are still customers in the queue. To solve this, conditionals are used in the customer thread, before the fileread variable is changed.

assignmentMethods.c, customer

```
while (list->size != 0) {
    pthread_cond_signal(&cond);
    pthread_cond_wait(&queueFull, &listLock);
}
pthread_mutex_unlock(&listLock);
// changing the file read value indicates to tellers they should terminate
pthread_mutex_lock(&fileLock);
fileread = 1;
pthread_mutex_unlock(&fileLock);
// broadcast signals all tellers that they should terminate
pthread_cond_broadcast(&cond);
```

When the list is not empty, the customer thread will signal to the tellers, to continue operations and wait for the queueFull signal from the teller. This will continue until the list is empty, but no customers will be added. When the list is empty, the file read variable is set to 1. At this point in time a teller thread may have broken out of the loop and terminated or is waiting for the queue to be filled. If it has terminated, everything is fine, but if it is waiting, that's a problem as the customer thread has already terminated. To make tellers to stop waiting pthread_cond_broadcast() is used. Broadcast is used as there may be multiple threads waiting, this will signal to all of them unlike if pthread_cond_signal() is used, where it only wakes one of the teller threads. After broadcast is called, all the tellers terminate.

The last teller has the unique responsibility of printing the number of customers served by each teller. Because of the nature of threads, any of the four teller threads can be the last teller to terminate. The tellersLeft variable is initialized to 4 and will be decremented by one each time a teller terminates. This operation is protected by the listLock and fileLock and is only done after the tellersLeft variable has been checked. This means there will always be a situation where there is a last teller. When it is equal to 1, the last teller is found and, the final statement for the log can be written.

```
// check if the teller is the last one left
// list lock is still enabled here so checking tellers left is okay
if (tellersLeft == 1) {
    ...
    ...
    ...
}
// decrements this value when a teller terminates
tellersLeft--;
```

Writing to the log file is also a protected function. Each time a thread wants to write to the `r_log` file, it must enable the `writeToLog` mutex. After which, it must unlock it. For example:

```
pthread_mutex_lock(&writeToLog);
logToFile("Teller 1 serverd: ");
sprintf(served1, "%d\n", served[0]);
logToFile(served1);
logToFile("Teller 2 serverd: ");
sprintf(served2, "%d\n", served[1]);
logToFile(served2);
logToFile("Teller 3 serverd: ");
sprintf(served3, "%d\n", served[2]);
logToFile(served3);
logToFile("Teller 4 serverd: ");
sprintf(served4, "%d\n", served[3]);
logToFile(served4);
pthread_mutex_unlock(&writeToLog);
```

The last shared variable is the `served` array. This is an integer array, that holds four values. Each index of the array belongs to one of the tellers. This variable is not protected by any locks when it is modified as each teller has their own index in their array that they can modify. For example:

```
teller->served += 1;
served[atoi(teller->id) - 1] = teller->served;
```

Because each teller has their own unique ID, each teller will only ever access their index of the array, therefore, there cannot be a situation where a teller accesses another teller's index on the `served` array.

Tests and inconsistencies

When $m = 0$, the program is stuck in an infinite loop, this makes sense as there is no way for the customers to be added to the `c_queue`. To fix this issue the code exits at the start of execution if `m` is set to 0.

```
if(m <= 0){  
    exit(0);  
}
```

Testing

There are no other cases where the program does not run as intend. To test the program there are two ways. The first is to test it with a low m value, high t_c value and low t_d, t_i, t_w values. This will be a representation of the bounded buffer problem where the producer cannot add to the buffer fast enough to keep up with the consumer. In this case the producer is the customer thread and the consumers are the teller threads. The second way of testing is to have a high m value, low t_c value and high t_d, t_i, t_w values. This will represent the bounded buffer problem where, the consumers cannot keep up with the producer. If the program works under these edge cases, then it indicates that it is correct. Also test it with other values to be sure that it is correct.

Sample input and output

All these test were done with the following c_file.

```
1 I
2 W
3 I
4 W
5 I
6 I
7 I
8 I
9 I
10 D
11 I
12 I
13 D
14 W
15 I
16 I
17 D
18 I
19 D
20 W
21 I
22 D
23 D
24 I
25 D
26 W
27 W
28 I
29 D
30 D
31 I
32 I
33 D
34 I
35 W
36 D
37 I
38 I
39 W
40 I
41 I
42 D
43 W
44 D
45 W
46 D
47 I
48 D
49 W
50 W
51 D
52 W
53 D
```

54 W
55 W
56 I
57 W
58 D
59 W
60 I
61 D
62 D
63 D
64 D
65 I
66 I
67 D
68 D
69 I
70 I
71 D
72 W
73 D
74 I
75 I
76 I
77 D
78 W
79 I
80 D
81 D
82 W
83 D
84 W
85 D
86 W
87 I
88 I
89 W
90 D
91 W
92 D
93 I
94 D
95 D
96 I
97 W
98 I
99 I
100 I

Test 1

This represents the first test mentioned in the Testing and inconsistencies section of the report. Where the producer (customer thread) cannot add to the `c_queue` frequently enough to keep up with the consumer (teller thread). The program completes this with an even distribution of customers served by each teller. Because $m = 1$ only one teller can every access and respond to a customer at any given time. This can be seen in the results as soon as a customer is added, a teller thread gets it and responds to it. The results also show the teller threads waiting for the `c_queue` to be filled before, servicing a customer, and as soon as a customer is added one of the teller threads service it.

Input

$m = 1$

$t_c = 5$

$t_i = 1$

$t_w = 1$

$t_d = 1$

Output

```
-----
1: I
Arrival Time: 22:34:16
-----
Teller: 4
Customer: 1
Arrival time: 22:34:16
Response time: 22:34:16
Teller: 4
Customer: 1
Arrival time: 22:34:16
Completion time: 22:34:17
-----
2: W
Arrival Time: 22:34:21
-----
Teller: 1
Customer: 2
Arrival time: 22:34:21
Response time: 22:34:21
Teller: 1
Customer: 2
Arrival time: 22:34:21
Completion time: 22:34:22
-----
3: I
Arrival Time: 22:34:26
-----
Teller: 2
Customer: 3
Arrival time: 22:34:26
Response time: 22:34:26
Teller: 2
Customer: 3
Arrival time: 22:34:26
Completion time: 22:34:27
-----
4: W
```

Arival Time: 22:34:31

Teller: 3
Customer: 4
Arrival time: 22:34:31
Response time: 22:34:31
Teller: 3
Customer: 4
Arrival time: 22:34:31
Completion time: 22:34:32

5: I
Arival Time: 22:34:36

Teller: 4
Customer: 5
Arrival time: 22:34:36
Response time: 22:34:36
Teller: 4
Customer: 5
Arrival time: 22:34:36
Completion time: 22:34:37

6: I
Arival Time: 22:34:41

Teller: 1
Customer: 6
Arrival time: 22:34:41
Response time: 22:34:41
Teller: 1
Customer: 6
Arrival time: 22:34:41
Completion time: 22:34:42

7: I
Arival Time: 22:34:46

Teller: 2
Customer: 7
Arrival time: 22:34:46
Response time: 22:34:46
Teller: 2
Customer: 7
Arrival time: 22:34:46
Completion time: 22:34:47

8: I
Arival Time: 22:34:51

Teller: 3
Customer: 8
Arrival time: 22:34:51
Response time: 22:34:51
Teller: 3
Customer: 8

Arrival time: 22:34:51
Completion time: 22:34:52

9: I
Arrival Time: 22:34:56

Teller: 4
Customer: 9
Arrival time: 22:34:56
Response time: 22:34:56
Teller: 4
Customer: 9
Arrival time: 22:34:56
Completion time: 22:34:57

10: D
Arrival Time: 22:35:1

Teller: 1
Customer: 10
Arrival time: 22:35:1
Response time: 22:35:1
Teller: 1
Customer: 10
Arrival time: 22:35:1
Completion time: 22:35:2

11: I
Arrival Time: 22:35:6

Teller: 2
Customer: 11
Arrival time: 22:35:6
Response time: 22:35:6
Teller: 2
Customer: 11
Arrival time: 22:35:6
Completion time: 22:35:7

12: I
Arrival Time: 22:35:11

Teller: 3
Customer: 12
Arrival time: 22:35:11
Response time: 22:35:11
Teller: 3
Customer: 12
Arrival time: 22:35:11
Completion time: 22:35:12

13: D
Arrival Time: 22:35:16

Teller: 4
Customer: 13

Arrival time: 22:35:16
Response time: 22:35:16
Teller: 4
Customer: 13
Arrival time: 22:35:16
Completion time: 22:35:17

14: W
Arrival Time: 22:35:21

Teller: 1
Customer: 14
Arrival time: 22:35:21
Response time: 22:35:21
Teller: 1
Customer: 14
Arrival time: 22:35:21
Completion time: 22:35:22

15: I
Arrival Time: 22:35:26

Teller: 2
Customer: 15
Arrival time: 22:35:26
Response time: 22:35:26
Teller: 2
Customer: 15
Arrival time: 22:35:26
Completion time: 22:35:27

16: I
Arrival Time: 22:35:31

Teller: 3
Customer: 16
Arrival time: 22:35:31
Response time: 22:35:31
Teller: 3
Customer: 16
Arrival time: 22:35:31
Completion time: 22:35:32

17: D
Arrival Time: 22:35:36

Teller: 4
Customer: 17
Arrival time: 22:35:36
Response time: 22:35:36
Teller: 4
Customer: 17
Arrival time: 22:35:36
Completion time: 22:35:37

18: I

Arival Time: 22:35:41

Teller: 1

Customer: 18

Arrival time: 22:35:41

Response time: 22:35:41

Teller: 1

Customer: 18

Arrival time: 22:35:41

Completion time: 22:35:42

19: D

Arival Time: 22:35:46

Teller: 2

Customer: 19

Arrival time: 22:35:46

Response time: 22:35:46

Teller: 2

Customer: 19

Arrival time: 22:35:46

Completion time: 22:35:47

20: W

Arival Time: 22:35:51

Teller: 3

Customer: 20

Arrival time: 22:35:51

Response time: 22:35:51

Teller: 3

Customer: 20

Arrival time: 22:35:51

Completion time: 22:35:52

21: I

Arival Time: 22:35:56

Teller: 4

Customer: 21

Arrival time: 22:35:56

Response time: 22:35:56

Teller: 4

Customer: 21

Arrival time: 22:35:56

Completion time: 22:35:57

22: D

Arival Time: 22:36:1

Teller: 1

Customer: 22

Arrival time: 22:36:1

Response time: 22:36:1

Teller: 1

Customer: 22

Arrival time: 22:36:1
Completion time: 22:36:2

23: D
Arrival Time: 22:36:6

Teller: 2
Customer: 23
Arrival time: 22:36:6
Response time: 22:36:6
Teller: 2
Customer: 23
Arrival time: 22:36:6
Completion time: 22:36:7

24: I
Arrival Time: 22:36:11

Teller: 3
Customer: 24
Arrival time: 22:36:11
Response time: 22:36:11
Teller: 3
Customer: 24
Arrival time: 22:36:11
Completion time: 22:36:12

25: D
Arrival Time: 22:36:16

Teller: 4
Customer: 25
Arrival time: 22:36:16
Response time: 22:36:16
Teller: 4
Customer: 25
Arrival time: 22:36:16
Completion time: 22:36:17

26: W
Arrival Time: 22:36:21

Teller: 1
Customer: 26
Arrival time: 22:36:21
Response time: 22:36:21
Teller: 1
Customer: 26
Arrival time: 22:36:21
Completion time: 22:36:22

27: W
Arrival Time: 22:36:26

Teller: 2
Customer: 27

Arrival time: 22:36:26
Response time: 22:36:26
Teller: 2
Customer: 27
Arrival time: 22:36:26
Completion time: 22:36:27

28: I
Arrival Time: 22:36:31

Teller: 3
Customer: 28
Arrival time: 22:36:31
Response time: 22:36:31
Teller: 3
Customer: 28
Arrival time: 22:36:31
Completion time: 22:36:32

29: D
Arrival Time: 22:36:36

Teller: 4
Customer: 29
Arrival time: 22:36:36
Response time: 22:36:36
Teller: 4
Customer: 29
Arrival time: 22:36:36
Completion time: 22:36:37

30: D
Arrival Time: 22:36:41

Teller: 1
Customer: 30
Arrival time: 22:36:41
Response time: 22:36:41
Teller: 1
Customer: 30
Arrival time: 22:36:41
Completion time: 22:36:42

31: I
Arrival Time: 22:36:46

Teller: 2
Customer: 31
Arrival time: 22:36:46
Response time: 22:36:46
Teller: 2
Customer: 31
Arrival time: 22:36:46
Completion time: 22:36:47

32: I

Arival Time: 22:36:51

Teller: 3

Customer: 32

Arrival time: 22:36:51

Response time: 22:36:51

Teller: 3

Customer: 32

Arrival time: 22:36:51

Completion time: 22:36:52

33: D

Arival Time: 22:36:56

Teller: 4

Customer: 33

Arrival time: 22:36:56

Response time: 22:36:56

Teller: 4

Customer: 33

Arrival time: 22:36:56

Completion time: 22:36:57

34: I

Arival Time: 22:37:1

Teller: 1

Customer: 34

Arrival time: 22:37:1

Response time: 22:37:1

Teller: 1

Customer: 34

Arrival time: 22:37:1

Completion time: 22:37:2

35: W

Arival Time: 22:37:6

Teller: 2

Customer: 35

Arrival time: 22:37:6

Response time: 22:37:6

Teller: 2

Customer: 35

Arrival time: 22:37:6

Completion time: 22:37:7

36: D

Arival Time: 22:37:11

Teller: 3

Customer: 36

Arrival time: 22:37:11

Response time: 22:37:11

Teller: 3

Customer: 36

Arrival time: 22:37:11
Completion time: 22:37:12

37: I
Arrival Time: 22:37:16

Teller: 4
Customer: 37
Arrival time: 22:37:16
Response time: 22:37:16
Teller: 4
Customer: 37
Arrival time: 22:37:16
Completion time: 22:37:17

38: I
Arrival Time: 22:37:21

Teller: 1
Customer: 38
Arrival time: 22:37:21
Response time: 22:37:21
Teller: 1
Customer: 38
Arrival time: 22:37:21
Completion time: 22:37:22

39: W
Arrival Time: 22:37:26

Teller: 2
Customer: 39
Arrival time: 22:37:26
Response time: 22:37:26
Teller: 2
Customer: 39
Arrival time: 22:37:26
Completion time: 22:37:27

40: I
Arrival Time: 22:37:31

Teller: 3
Customer: 40
Arrival time: 22:37:31
Response time: 22:37:31
Teller: 3
Customer: 40
Arrival time: 22:37:31
Completion time: 22:37:32

41: I
Arrival Time: 22:37:36

Teller: 4
Customer: 41

Arrival time: 22:37:36
Response time: 22:37:36
Teller: 4
Customer: 41
Arrival time: 22:37:36
Completion time: 22:37:37

42: D
Arrival Time: 22:37:41

Teller: 1
Customer: 42
Arrival time: 22:37:41
Response time: 22:37:41
Teller: 1
Customer: 42
Arrival time: 22:37:41
Completion time: 22:37:42

43: W
Arrival Time: 22:37:46

Teller: 2
Customer: 43
Arrival time: 22:37:46
Response time: 22:37:46
Teller: 2
Customer: 43
Arrival time: 22:37:46
Completion time: 22:37:47

44: D
Arrival Time: 22:37:51

Teller: 3
Customer: 44
Arrival time: 22:37:51
Response time: 22:37:51
Teller: 3
Customer: 44
Arrival time: 22:37:51
Completion time: 22:37:52

45: W
Arrival Time: 22:37:56

Teller: 4
Customer: 45
Arrival time: 22:37:56
Response time: 22:37:56
Teller: 4
Customer: 45
Arrival time: 22:37:56
Completion time: 22:37:57

46: D

Arival Time: 22:38:1

Teller: 1

Customer: 46

Arrival time: 22:38:1

Response time: 22:38:1

Teller: 1

Customer: 46

Arrival time: 22:38:1

Completion time: 22:38:2

47: I

Arival Time: 22:38:6

Teller: 2

Customer: 47

Arrival time: 22:38:6

Response time: 22:38:6

Teller: 2

Customer: 47

Arrival time: 22:38:6

Completion time: 22:38:7

48: D

Arival Time: 22:38:11

Teller: 3

Customer: 48

Arrival time: 22:38:11

Response time: 22:38:11

Teller: 3

Customer: 48

Arrival time: 22:38:11

Completion time: 22:38:12

49: W

Arival Time: 22:38:16

Teller: 4

Customer: 49

Arrival time: 22:38:16

Response time: 22:38:16

Teller: 4

Customer: 49

Arrival time: 22:38:16

Completion time: 22:38:17

50: W

Arival Time: 22:38:21

Teller: 1

Customer: 50

Arrival time: 22:38:21

Response time: 22:38:21

Teller: 1

Customer: 50

Arrival time: 22:38:21
Completion time: 22:38:22

51: D
Arrival Time: 22:38:26

Teller: 2
Customer: 51
Arrival time: 22:38:26
Response time: 22:38:26
Teller: 2
Customer: 51
Arrival time: 22:38:26
Completion time: 22:38:27

52: W
Arrival Time: 22:38:31

Teller: 3
Customer: 52
Arrival time: 22:38:31
Response time: 22:38:31
Teller: 3
Customer: 52
Arrival time: 22:38:31
Completion time: 22:38:32

53: D
Arrival Time: 22:38:36

Teller: 4
Customer: 53
Arrival time: 22:38:36
Response time: 22:38:36
Teller: 4
Customer: 53
Arrival time: 22:38:36
Completion time: 22:38:37

54: W
Arrival Time: 22:38:41

Teller: 1
Customer: 54
Arrival time: 22:38:41
Response time: 22:38:41
Teller: 1
Customer: 54
Arrival time: 22:38:41
Completion time: 22:38:42

55: W
Arrival Time: 22:38:46

Teller: 2
Customer: 55

Arrival time: 22:38:46
Response time: 22:38:46
Teller: 2
Customer: 55
Arrival time: 22:38:46
Completion time: 22:38:47

56: I
Arrival Time: 22:38:51

Teller: 3
Customer: 56
Arrival time: 22:38:51
Response time: 22:38:51
Teller: 3
Customer: 56
Arrival time: 22:38:51
Completion time: 22:38:52

57: W
Arrival Time: 22:38:56

Teller: 4
Customer: 57
Arrival time: 22:38:56
Response time: 22:38:56
Teller: 4
Customer: 57
Arrival time: 22:38:56
Completion time: 22:38:57

58: D
Arrival Time: 22:39:1

Teller: 1
Customer: 58
Arrival time: 22:39:1
Response time: 22:39:1
Teller: 1
Customer: 58
Arrival time: 22:39:1
Completion time: 22:39:2

59: W
Arrival Time: 22:39:6

Teller: 2
Customer: 59
Arrival time: 22:39:6
Response time: 22:39:6
Teller: 2
Customer: 59
Arrival time: 22:39:6
Completion time: 22:39:7

60: I

```

Arival Time: 22:39:11
-----
Teller: 3
Customer: 60
Arrival time: 22:39:11
Response time: 22:39:11
Teller: 3
Customer: 60
Arrival time: 22:39:11
Completion time: 22:39:12
-----
61: D
Arival Time: 22:39:16
-----
Teller: 4
Customer: 61
Arrival time: 22:39:16
Response time: 22:39:16
Teller: 4
Customer: 61
Arrival time: 22:39:16
Completion time: 22:39:17
-----
62: D
Arival Time: 22:39:21
-----
Teller: 1
Customer: 62
Arrival time: 22:39:21
Response time: 22:39:21
Teller: 1
Customer: 62
Arrival time: 22:39:21
Completion time: 22:39:22
-----
63: D
Arival Time: 22:39:26
-----
Teller: 2
Customer: 63
Arrival time: 22:39:26
Response time: 22:39:26
Teller: 2
Customer: 63
Arrival time: 22:39:26
Completion time: 22:39:27
-----
64: D
Arival Time: 22:39:31
-----
Teller: 3
Customer: 64
Arrival time: 22:39:31
Response time: 22:39:31
Teller: 3
Customer: 64

```

Arrival time: 22:39:31
Completion time: 22:39:32

65: I
Arrival Time: 22:39:36

Teller: 4
Customer: 65
Arrival time: 22:39:36
Response time: 22:39:36
Teller: 4
Customer: 65
Arrival time: 22:39:36
Completion time: 22:39:37

66: I
Arrival Time: 22:39:41

Teller: 1
Customer: 66
Arrival time: 22:39:41
Response time: 22:39:41
Teller: 1
Customer: 66
Arrival time: 22:39:41
Completion time: 22:39:42

67: D
Arrival Time: 22:39:46

Teller: 2
Customer: 67
Arrival time: 22:39:46
Response time: 22:39:46
Teller: 2
Customer: 67
Arrival time: 22:39:46
Completion time: 22:39:47

68: D
Arrival Time: 22:39:51

Teller: 3
Customer: 68
Arrival time: 22:39:51
Response time: 22:39:51
Teller: 3
Customer: 68
Arrival time: 22:39:51
Completion time: 22:39:52

69: I
Arrival Time: 22:39:56

Teller: 4
Customer: 69

Arrival time: 22:39:56
Response time: 22:39:56
Teller: 4
Customer: 69
Arrival time: 22:39:56
Completion time: 22:39:57

70: I
Arrival Time: 22:40:1

Teller: 1
Customer: 70
Arrival time: 22:40:1
Response time: 22:40:1
Teller: 1
Customer: 70
Arrival time: 22:40:1
Completion time: 22:40:2

71: D
Arrival Time: 22:40:6

Teller: 2
Customer: 71
Arrival time: 22:40:6
Response time: 22:40:6
Teller: 2
Customer: 71
Arrival time: 22:40:6
Completion time: 22:40:7

72: W
Arrival Time: 22:40:11

Teller: 3
Customer: 72
Arrival time: 22:40:11
Response time: 22:40:11
Teller: 3
Customer: 72
Arrival time: 22:40:11
Completion time: 22:40:12

73: D
Arrival Time: 22:40:16

Teller: 4
Customer: 73
Arrival time: 22:40:16
Response time: 22:40:16
Teller: 4
Customer: 73
Arrival time: 22:40:16
Completion time: 22:40:17

74: I

Arival Time: 22:40:21

Teller: 1

Customer: 74

Arrival time: 22:40:21

Response time: 22:40:21

Teller: 1

Customer: 74

Arrival time: 22:40:21

Completion time: 22:40:22

75: I

Arival Time: 22:40:26

Teller: 2

Customer: 75

Arrival time: 22:40:26

Response time: 22:40:26

Teller: 2

Customer: 75

Arrival time: 22:40:26

Completion time: 22:40:27

76: I

Arival Time: 22:40:31

Teller: 3

Customer: 76

Arrival time: 22:40:31

Response time: 22:40:31

Teller: 3

Customer: 76

Arrival time: 22:40:31

Completion time: 22:40:32

77: D

Arival Time: 22:40:36

Teller: 4

Customer: 77

Arrival time: 22:40:36

Response time: 22:40:36

Teller: 4

Customer: 77

Arrival time: 22:40:36

Completion time: 22:40:37

78: W

Arival Time: 22:40:41

Teller: 1

Customer: 78

Arrival time: 22:40:41

Response time: 22:40:41

Teller: 1

Customer: 78

Arrival time: 22:40:41
Completion time: 22:40:42

79: I
Arrival Time: 22:40:46

Teller: 2
Customer: 79
Arrival time: 22:40:46
Response time: 22:40:46
Teller: 2
Customer: 79
Arrival time: 22:40:46
Completion time: 22:40:47

80: D
Arrival Time: 22:40:51

Teller: 3
Customer: 80
Arrival time: 22:40:51
Response time: 22:40:51
Teller: 3
Customer: 80
Arrival time: 22:40:51
Completion time: 22:40:52

81: D
Arrival Time: 22:40:56

Teller: 4
Customer: 81
Arrival time: 22:40:56
Response time: 22:40:56
Teller: 4
Customer: 81
Arrival time: 22:40:56
Completion time: 22:40:57

82: W
Arrival Time: 22:41:1

Teller: 1
Customer: 82
Arrival time: 22:41:1
Response time: 22:41:1
Teller: 1
Customer: 82
Arrival time: 22:41:1
Completion time: 22:41:2

83: D
Arrival Time: 22:41:6

Teller: 2
Customer: 83

Arrival time: 22:41:6
Response time: 22:41:6
Teller: 2
Customer: 83
Arrival time: 22:41:6
Completion time: 22:41:7

84: W
Arrival Time: 22:41:11

Teller: 3
Customer: 84
Arrival time: 22:41:11
Response time: 22:41:11
Teller: 3
Customer: 84
Arrival time: 22:41:11
Completion time: 22:41:12

85: D
Arrival Time: 22:41:16

Teller: 4
Customer: 85
Arrival time: 22:41:16
Response time: 22:41:16
Teller: 4
Customer: 85
Arrival time: 22:41:16
Completion time: 22:41:17

86: W
Arrival Time: 22:41:21

Teller: 1
Customer: 86
Arrival time: 22:41:21
Response time: 22:41:21
Teller: 1
Customer: 86
Arrival time: 22:41:21
Completion time: 22:41:22

87: I
Arrival Time: 22:41:26

Teller: 2
Customer: 87
Arrival time: 22:41:26
Response time: 22:41:26
Teller: 2
Customer: 87
Arrival time: 22:41:26
Completion time: 22:41:27

88: I

Arival Time: 22:41:31

Teller: 3

Customer: 88

Arrival time: 22:41:31

Response time: 22:41:31

Teller: 3

Customer: 88

Arrival time: 22:41:31

Completion time: 22:41:32

89: W

Arival Time: 22:41:36

Teller: 4

Customer: 89

Arrival time: 22:41:36

Response time: 22:41:36

Teller: 4

Customer: 89

Arrival time: 22:41:36

Completion time: 22:41:37

90: D

Arival Time: 22:41:41

Teller: 1

Customer: 90

Arrival time: 22:41:41

Response time: 22:41:41

Teller: 1

Customer: 90

Arrival time: 22:41:41

Completion time: 22:41:42

91: W

Arival Time: 22:41:46

Teller: 2

Customer: 91

Arrival time: 22:41:46

Response time: 22:41:46

Teller: 2

Customer: 91

Arrival time: 22:41:46

Completion time: 22:41:47

92: D

Arival Time: 22:41:51

Teller: 3

Customer: 92

Arrival time: 22:41:51

Response time: 22:41:51

Teller: 3

Customer: 92

Arrival time: 22:41:51
Completion time: 22:41:52

93: I
Arrival Time: 22:41:56

Teller: 4
Customer: 93
Arrival time: 22:41:56
Response time: 22:41:56
Teller: 4
Customer: 93
Arrival time: 22:41:56
Completion time: 22:41:57

94: D
Arrival Time: 22:42:1

Teller: 1
Customer: 94
Arrival time: 22:42:1
Response time: 22:42:1
Teller: 1
Customer: 94
Arrival time: 22:42:1
Completion time: 22:42:2

95: D
Arrival Time: 22:42:6

Teller: 2
Customer: 95
Arrival time: 22:42:6
Response time: 22:42:6
Teller: 2
Customer: 95
Arrival time: 22:42:6
Completion time: 22:42:7

96: I
Arrival Time: 22:42:11

Teller: 3
Customer: 96
Arrival time: 22:42:11
Response time: 22:42:11
Teller: 3
Customer: 96
Arrival time: 22:42:11
Completion time: 22:42:12

97: W
Arrival Time: 22:42:16

Teller: 4
Customer: 97

Arrival time: 22:42:16
Response time: 22:42:16
Teller: 4
Customer: 97
Arrival time: 22:42:16
Completion time: 22:42:17

98: I
Arrival Time: 22:42:21

Teller: 1
Customer: 98
Arrival time: 22:42:21
Response time: 22:42:21
Teller: 1
Customer: 98
Arrival time: 22:42:21
Completion time: 22:42:22

99: I
Arrival Time: 22:42:26

Teller: 2
Customer: 99
Arrival time: 22:42:26
Response time: 22:42:26
Teller: 2
Customer: 99
Arrival time: 22:42:26
Completion time: 22:42:27

100: I
Arrival Time: 22:42:31

Teller: 3
Customer: 100
Arrival time: 22:42:31
Response time: 22:42:31
Teller: 3
Customer: 100
Arrival time: 22:42:31
Completion time: 22:42:32
Teller 1 serverd: 25
Teller 2 serverd: 25
Teller 3 serverd: 25
Teller 4 serverd: 25

Test 2

This simulates the bounded buffer problem where the consumers(teller threads) cannot keep up with the producer(customer thread). In the results, it can be seen that there is a line of arrival of customers as the teller threads server the customer they currently have. This shows the customer thread waiting for the c_queue to be empty before adding more customers, thus the program is working as intended. And also, as soon as there is room in the c_queue, a customer is added.

Input

$m = 8$

$t_c = 1$

$t_i = 15$

$t_w = 15$

$t_d = 15$

Output

1: I

Arival Time: 23:7:7

Teller: 1

Customer: 1

Arrival time: 23:7:7

Response time: 23:7:7

2: W

Arival Time: 23:7:8

Teller: 2

Customer: 2

Arrival time: 23:7:8

Response time: 23:7:8

3: I

Arival Time: 23:7:9

Teller: 4

Customer: 3

Arrival time: 23:7:9

Response time: 23:7:9

4: W

Arival Time: 23:7:10

Teller: 3

Customer: 4

Arrival time: 23:7:10

Response time: 23:7:10

5: I

Arival Time: 23:7:11

6: I

Arival Time: 23:7:12

7: I
Arival Time: 23:7:13

8: I
Arival Time: 23:7:14

9: I
Arival Time: 23:7:15

10: D
Arival Time: 23:7:16

11: I
Arival Time: 23:7:17

12: I
Arival Time: 23:7:18

Teller: 1
Customer: 1
Arrival time: 23:7:7
Completion time: 23:7:22
Teller: 1
Customer: 5
Arrival time: 23:7:11
Response time: 23:7:22

13: D
Arival Time: 23:7:22

Teller: 2
Customer: 2
Arrival time: 23:7:8
Completion time: 23:7:23
Teller: 2
Customer: 6
Arrival time: 23:7:12
Response time: 23:7:23

14: W
Arival Time: 23:7:23

Teller: 4
Customer: 3
Arrival time: 23:7:9
Completion time: 23:7:24
Teller: 4
Customer: 7
Arrival time: 23:7:13

Response time: 23:7:24

15: I

Arrival Time: 23:7:24

Teller: 3

Customer: 4

Arrival time: 23:7:10

Completion time: 23:7:25

Teller: 3

Customer: 8

Arrival time: 23:7:14

Response time: 23:7:25

16: I

Arrival Time: 23:7:25

Teller: 1

Customer: 5

Arrival time: 23:7:11

Completion time: 23:7:37

Teller: 1

Customer: 9

Arrival time: 23:7:15

Response time: 23:7:37

17: D

Arrival Time: 23:7:37

Teller: 2

Customer: 6

Arrival time: 23:7:12

Completion time: 23:7:38

Teller: 2

Customer: 10

Arrival time: 23:7:16

Response time: 23:7:38

18: I

Arrival Time: 23:7:38

Teller: 4

Customer: 7

Arrival time: 23:7:13

Completion time: 23:7:39

Teller: 4

Customer: 11

Arrival time: 23:7:17

Response time: 23:7:39

19: D

Arrival Time: 23:7:39

Teller: 3

Customer: 8

Arrival time: 23:7:14

Completion time: 23:7:40
Teller: 3
Customer: 12
Arrival time: 23:7:18
Response time: 23:7:40

20: W
Arival Time: 23:7:40

Teller: 1
Customer: 9
Arrival time: 23:7:15
Completion time: 23:7:52
Teller: 1
Customer: 13
Arrival time: 23:7:22
Response time: 23:7:52

21: I
Arival Time: 23:7:52

Teller: 2
Customer: 10
Arrival time: 23:7:16
Completion time: 23:7:53
Teller: 2
Customer: 14
Arrival time: 23:7:23
Response time: 23:7:53

22: D
Arival Time: 23:7:53

Teller: 4
Customer: 11
Arrival time: 23:7:17
Completion time: 23:7:54
Teller: 4
Customer: 15
Arrival time: 23:7:24
Response time: 23:7:54

23: D
Arival Time: 23:7:54

Teller: 3
Customer: 12
Arrival time: 23:7:18
Completion time: 23:7:55
Teller: 3
Customer: 16
Arrival time: 23:7:25
Response time: 23:7:55

24: I
Arival Time: 23:7:55

```

-----
Teller: 1
Customer: 13
Arrival time: 23:7:22
Completion time: 23:8:7
Teller: 1
Customer: 17
Arrival time: 23:7:37
Response time: 23:8:7
-----

25: D
Arrival Time: 23:8:7
-----

Teller: 2
Customer: 14
Arrival time: 23:7:23
Completion time: 23:8:8
Teller: 2
Customer: 18
Arrival time: 23:7:38
Response time: 23:8:8
-----

26: W
Arrival Time: 23:8:8
-----

Teller: 4
Customer: 15
Arrival time: 23:7:24
Completion time: 23:8:9
Teller: 4
Customer: 19
Arrival time: 23:7:39
Response time: 23:8:9
-----

27: W
Arrival Time: 23:8:9
-----

Teller: 3
Customer: 16
Arrival time: 23:7:25
Completion time: 23:8:10
Teller: 3
Customer: 20
Arrival time: 23:7:40
Response time: 23:8:10
-----

28: I
Arrival Time: 23:8:10
-----

Teller: 1
Customer: 17
Arrival time: 23:7:37
Completion time: 23:8:22
Teller: 1
Customer: 21
Arrival time: 23:7:52

```

Response time: 23:8:22

29: D

Arrival Time: 23:8:22

Teller: 2

Customer: 18

Arrival time: 23:7:38

Completion time: 23:8:23

Teller: 2

Customer: 22

Arrival time: 23:7:53

Response time: 23:8:23

30: D

Arrival Time: 23:8:23

Teller: 4

Customer: 19

Arrival time: 23:7:39

Completion time: 23:8:24

Teller: 4

Customer: 23

Arrival time: 23:7:54

Response time: 23:8:24

31: I

Arrival Time: 23:8:24

Teller: 3

Customer: 20

Arrival time: 23:7:40

Completion time: 23:8:25

Teller: 3

Customer: 24

Arrival time: 23:7:55

Response time: 23:8:25

32: I

Arrival Time: 23:8:25

Teller: 1

Customer: 21

Arrival time: 23:7:52

Completion time: 23:8:37

Teller: 1

Customer: 25

Arrival time: 23:8:7

Response time: 23:8:37

33: D

Arrival Time: 23:8:37

Teller: 2

Customer: 22

Arrival time: 23:7:53

Completion time: 23:8:38
Teller: 2
Customer: 26
Arrival time: 23:8:8
Response time: 23:8:38

34: I
Arival Time: 23:8:38

Teller: 4
Customer: 23
Arrival time: 23:7:54
Completion time: 23:8:39
Teller: 4
Customer: 27
Arrival time: 23:8:9
Response time: 23:8:39

35: W
Arival Time: 23:8:39

Teller: 3
Customer: 24
Arrival time: 23:7:55
Completion time: 23:8:40
Teller: 3
Customer: 28
Arrival time: 23:8:10
Response time: 23:8:40

36: D
Arival Time: 23:8:40

Teller: 1
Customer: 25
Arrival time: 23:8:7
Completion time: 23:8:52
Teller: 1
Customer: 29
Arrival time: 23:8:22
Response time: 23:8:52

37: I
Arival Time: 23:8:52

Teller: 2
Customer: 26
Arrival time: 23:8:8
Completion time: 23:8:53
Teller: 2
Customer: 30
Arrival time: 23:8:23
Response time: 23:8:53

38: I
Arival Time: 23:8:53

```

-----
Teller: 4
Customer: 27
Arrival time: 23:8:9
Completion time: 23:8:54
Teller: 4
Customer: 31
Arrival time: 23:8:24
Response time: 23:8:54
-----
39: W
Arival Time: 23:8:54
-----
Teller: 3
Customer: 28
Arrival time: 23:8:10
Completion time: 23:8:55
Teller: 3
Customer: 32
Arrival time: 23:8:25
Response time: 23:8:55
-----
40: I
Arival Time: 23:8:55
-----
Teller: 1
Customer: 29
Arrival time: 23:8:22
Completion time: 23:9:7
Teller: 1
Customer: 33
Arrival time: 23:8:37
Response time: 23:9:7
-----
41: I
Arival Time: 23:9:7
-----
Teller: 2
Customer: 30
Arrival time: 23:8:23
Completion time: 23:9:8
Teller: 2
Customer: 34
Arrival time: 23:8:38
Response time: 23:9:8
-----
42: D
Arival Time: 23:9:8
-----
Teller: 4
Customer: 31
Arrival time: 23:8:24
Completion time: 23:9:9
Teller: 4
Customer: 35
Arrival time: 23:8:39

```

Response time: 23:9:9

43: W

Arrival Time: 23:9:9

Teller: 3

Customer: 32

Arrival time: 23:8:25

Completion time: 23:9:10

Teller: 3

Customer: 36

Arrival time: 23:8:40

Response time: 23:9:10

44: D

Arrival Time: 23:9:10

Teller: 1

Customer: 33

Arrival time: 23:8:37

Completion time: 23:9:22

Teller: 1

Customer: 37

Arrival time: 23:8:52

Response time: 23:9:22

45: W

Arrival Time: 23:9:22

Teller: 2

Customer: 34

Arrival time: 23:8:38

Completion time: 23:9:23

Teller: 2

Customer: 38

Arrival time: 23:8:53

Response time: 23:9:23

46: D

Arrival Time: 23:9:23

Teller: 4

Customer: 35

Arrival time: 23:8:39

Completion time: 23:9:24

Teller: 4

Customer: 39

Arrival time: 23:8:54

Response time: 23:9:24

47: I

Arrival Time: 23:9:24

Teller: 3

Customer: 36

Arrival time: 23:8:40

Completion time: 23:9:25
Teller: 3
Customer: 40
Arrival time: 23:8:55
Response time: 23:9:25

48: D
Arrival Time: 23:9:25

Teller: 1
Customer: 37
Arrival time: 23:8:52
Completion time: 23:9:37
Teller: 1
Customer: 41
Arrival time: 23:9:7
Response time: 23:9:37

49: W
Arrival Time: 23:9:37

Teller: 2
Customer: 38
Arrival time: 23:8:53
Completion time: 23:9:38
Teller: 2
Customer: 42
Arrival time: 23:9:8
Response time: 23:9:38

50: W
Arrival Time: 23:9:38

Teller: 4
Customer: 39
Arrival time: 23:8:54
Completion time: 23:9:39
Teller: 4
Customer: 43
Arrival time: 23:9:9
Response time: 23:9:39

51: D
Arrival Time: 23:9:39

Teller: 3
Customer: 40
Arrival time: 23:8:55
Completion time: 23:9:40
Teller: 3
Customer: 44
Arrival time: 23:9:10
Response time: 23:9:40

52: W
Arrival Time: 23:9:40

Teller: 1
Customer: 41
Arrival time: 23:9:7
Completion time: 23:9:52
Teller: 1
Customer: 45
Arrival time: 23:9:22
Response time: 23:9:52

53: D
Arival Time: 23:9:52

Teller: 2
Customer: 42
Arrival time: 23:9:8
Completion time: 23:9:53
Teller: 2
Customer: 46
Arrival time: 23:9:23
Response time: 23:9:53

54: W
Arival Time: 23:9:53

Teller: 4
Customer: 43
Arrival time: 23:9:9
Completion time: 23:9:54
Teller: 4
Customer: 47
Arrival time: 23:9:24
Response time: 23:9:54

55: W
Arival Time: 23:9:54

Teller: 3
Customer: 44
Arrival time: 23:9:10
Completion time: 23:9:55
Teller: 3
Customer: 48
Arrival time: 23:9:25
Response time: 23:9:55

56: I
Arival Time: 23:9:55

Teller: 1
Customer: 45
Arrival time: 23:9:22
Completion time: 23:10:7
Teller: 1
Customer: 49
Arrival time: 23:9:37

Response time: 23:10:7

57: W

Arrival Time: 23:10:7

Teller: 2

Customer: 46

Arrival time: 23:9:23

Completion time: 23:10:8

Teller: 2

Customer: 50

Arrival time: 23:9:38

Response time: 23:10:8

58: D

Arrival Time: 23:10:8

Teller: 4

Customer: 47

Arrival time: 23:9:24

Completion time: 23:10:9

Teller: 4

Customer: 51

Arrival time: 23:9:39

Response time: 23:10:9

59: W

Arrival Time: 23:10:9

Teller: 3

Customer: 48

Arrival time: 23:9:25

Completion time: 23:10:10

Teller: 3

Customer: 52

Arrival time: 23:9:40

Response time: 23:10:10

60: I

Arrival Time: 23:10:10

Teller: 1

Customer: 49

Arrival time: 23:9:37

Completion time: 23:10:22

Teller: 1

Customer: 53

Arrival time: 23:9:52

Response time: 23:10:22

61: D

Arrival Time: 23:10:22

Teller: 2

Customer: 50

Arrival time: 23:9:38

Completion time: 23:10:23
Teller: 2
Customer: 54
Arrival time: 23:9:53
Response time: 23:10:23

62: D
Arrival Time: 23:10:23

Teller: 4
Customer: 51
Arrival time: 23:9:39
Completion time: 23:10:24
Teller: 4
Customer: 55
Arrival time: 23:9:54
Response time: 23:10:24

63: D
Arrival Time: 23:10:24

Teller: 3
Customer: 52
Arrival time: 23:9:40
Completion time: 23:10:25
Teller: 3
Customer: 56
Arrival time: 23:9:55
Response time: 23:10:25

64: D
Arrival Time: 23:10:25

Teller: 1
Customer: 53
Arrival time: 23:9:52
Completion time: 23:10:37
Teller: 1
Customer: 57
Arrival time: 23:10:7
Response time: 23:10:37

65: I
Arrival Time: 23:10:37

Teller: 2
Customer: 54
Arrival time: 23:9:53
Completion time: 23:10:38
Teller: 2
Customer: 58
Arrival time: 23:10:8
Response time: 23:10:38

66: I
Arrival Time: 23:10:38

Teller: 4
Customer: 55
Arrival time: 23:9:54
Completion time: 23:10:39
Teller: 4
Customer: 59
Arrival time: 23:10:9
Response time: 23:10:39

67: D
Arival Time: 23:10:39

Teller: 3
Customer: 56
Arrival time: 23:9:55
Completion time: 23:10:40
Teller: 3
Customer: 60
Arrival time: 23:10:10
Response time: 23:10:40

68: D
Arival Time: 23:10:40

Teller: 1
Customer: 57
Arrival time: 23:10:7
Completion time: 23:10:52
Teller: 1
Customer: 61
Arrival time: 23:10:22
Response time: 23:10:52

69: I
Arival Time: 23:10:52

Teller: 2
Customer: 58
Arrival time: 23:10:8
Completion time: 23:10:53
Teller: 2
Customer: 62
Arrival time: 23:10:23
Response time: 23:10:53

70: I
Arival Time: 23:10:53

Teller: 4
Customer: 59
Arrival time: 23:10:9
Completion time: 23:10:54
Teller: 4
Customer: 63
Arrival time: 23:10:24

Response time: 23:10:54

71: D

Arrival Time: 23:10:54

Teller: 3

Customer: 60

Arrival time: 23:10:10

Completion time: 23:10:55

Teller: 3

Customer: 64

Arrival time: 23:10:25

Response time: 23:10:55

72: W

Arrival Time: 23:10:55

Teller: 1

Customer: 61

Arrival time: 23:10:22

Completion time: 23:11:7

Teller: 1

Customer: 65

Arrival time: 23:10:37

Response time: 23:11:7

73: D

Arrival Time: 23:11:7

Teller: 2

Customer: 62

Arrival time: 23:10:23

Completion time: 23:11:8

Teller: 2

Customer: 66

Arrival time: 23:10:38

Response time: 23:11:8

74: I

Arrival Time: 23:11:8

Teller: 4

Customer: 63

Arrival time: 23:10:24

Completion time: 23:11:9

Teller: 4

Customer: 67

Arrival time: 23:10:39

Response time: 23:11:9

75: I

Arrival Time: 23:11:9

Teller: 3

Customer: 64

Arrival time: 23:10:25

Completion time: 23:11:10
Teller: 3
Customer: 68
Arrival time: 23:10:40
Response time: 23:11:10

76: I
Arrival Time: 23:11:10

Teller: 1
Customer: 65
Arrival time: 23:10:37
Completion time: 23:11:22
Teller: 1
Customer: 69
Arrival time: 23:10:52
Response time: 23:11:22

77: D
Arrival Time: 23:11:22

Teller: 2
Customer: 66
Arrival time: 23:10:38
Completion time: 23:11:23
Teller: 2
Customer: 70
Arrival time: 23:10:53
Response time: 23:11:23

78: W
Arrival Time: 23:11:23

Teller: 4
Customer: 67
Arrival time: 23:10:39
Completion time: 23:11:24
Teller: 4
Customer: 71
Arrival time: 23:10:54
Response time: 23:11:24

79: I
Arrival Time: 23:11:24

Teller: 3
Customer: 68
Arrival time: 23:10:40
Completion time: 23:11:25
Teller: 3
Customer: 72
Arrival time: 23:10:55
Response time: 23:11:25

80: D
Arrival Time: 23:11:25

Teller: 1
Customer: 69
Arrival time: 23:10:52
Completion time: 23:11:37
Teller: 1
Customer: 73
Arrival time: 23:11:7
Response time: 23:11:37

81: D
Arival Time: 23:11:37

Teller: 2
Customer: 70
Arrival time: 23:10:53
Completion time: 23:11:38
Teller: 2
Customer: 74
Arrival time: 23:11:8
Response time: 23:11:38

82: W
Arival Time: 23:11:38

Teller: 4
Customer: 71
Arrival time: 23:10:54
Completion time: 23:11:39
Teller: 4
Customer: 75
Arrival time: 23:11:9
Response time: 23:11:39

83: D
Arival Time: 23:11:39

Teller: 3
Customer: 72
Arrival time: 23:10:55
Completion time: 23:11:40
Teller: 3
Customer: 76
Arrival time: 23:11:10
Response time: 23:11:40

84: W
Arival Time: 23:11:40

Teller: 1
Customer: 73
Arrival time: 23:11:7
Completion time: 23:11:52
Teller: 1
Customer: 77
Arrival time: 23:11:22

Response time: 23:11:52

85: D

Arrival Time: 23:11:52

Teller: 2

Customer: 74

Arrival time: 23:11:8

Completion time: 23:11:53

Teller: 2

Customer: 78

Arrival time: 23:11:23

Response time: 23:11:53

86: W

Arrival Time: 23:11:53

Teller: 4

Customer: 75

Arrival time: 23:11:9

Completion time: 23:11:54

Teller: 4

Customer: 79

Arrival time: 23:11:24

Response time: 23:11:54

87: I

Arrival Time: 23:11:54

Teller: 3

Customer: 76

Arrival time: 23:11:10

Completion time: 23:11:55

Teller: 3

Customer: 80

Arrival time: 23:11:25

Response time: 23:11:55

88: I

Arrival Time: 23:11:55

Teller: 1

Customer: 77

Arrival time: 23:11:22

Completion time: 23:12:7

Teller: 1

Customer: 81

Arrival time: 23:11:37

Response time: 23:12:7

89: W

Arrival Time: 23:12:7

Teller: 2

Customer: 78

Arrival time: 23:11:23

Completion time: 23:12:8
Teller: 2
Customer: 82
Arrival time: 23:11:38
Response time: 23:12:8

90: D
Arrival Time: 23:12:8

Teller: 4
Customer: 79
Arrival time: 23:11:24
Completion time: 23:12:9
Teller: 4
Customer: 83
Arrival time: 23:11:39
Response time: 23:12:9

91: W
Arrival Time: 23:12:9

Teller: 3
Customer: 80
Arrival time: 23:11:25
Completion time: 23:12:10
Teller: 3
Customer: 84
Arrival time: 23:11:40
Response time: 23:12:10

92: D
Arrival Time: 23:12:10

Teller: 1
Customer: 81
Arrival time: 23:11:37
Completion time: 23:12:22
Teller: 1
Customer: 85
Arrival time: 23:11:52
Response time: 23:12:22

93: I
Arrival Time: 23:12:22

Teller: 2
Customer: 82
Arrival time: 23:11:38
Completion time: 23:12:23
Teller: 2
Customer: 86
Arrival time: 23:11:53
Response time: 23:12:23

94: D
Arrival Time: 23:12:23

Teller: 4
Customer: 83
Arrival time: 23:11:39
Completion time: 23:12:24
Teller: 4
Customer: 87
Arrival time: 23:11:54
Response time: 23:12:24

95: D
Arival Time: 23:12:24

Teller: 3
Customer: 84
Arrival time: 23:11:40
Completion time: 23:12:25
Teller: 3
Customer: 88
Arrival time: 23:11:55
Response time: 23:12:25

96: I
Arival Time: 23:12:25

Teller: 1
Customer: 85
Arrival time: 23:11:52
Completion time: 23:12:37
Teller: 1
Customer: 89
Arrival time: 23:12:7
Response time: 23:12:37

97: W
Arival Time: 23:12:37

Teller: 2
Customer: 86
Arrival time: 23:11:53
Completion time: 23:12:38
Teller: 2
Customer: 90
Arrival time: 23:12:8
Response time: 23:12:38

98: I
Arival Time: 23:12:38

Teller: 4
Customer: 87
Arrival time: 23:11:54
Completion time: 23:12:39
Teller: 4
Customer: 91
Arrival time: 23:12:9

Response time: 23:12:39

99: I

Arrival Time: 23:12:39

Teller: 3

Customer: 88

Arrival time: 23:11:55

Completion time: 23:12:40

Teller: 3

Customer: 92

Arrival time: 23:12:10

Response time: 23:12:40

100: I

Arrival Time: 23:12:40

Teller: 1

Customer: 89

Arrival time: 23:12:7

Completion time: 23:12:52

Teller: 1

Customer: 93

Arrival time: 23:12:22

Response time: 23:12:52

Teller: 2

Customer: 90

Arrival time: 23:12:8

Completion time: 23:12:53

Teller: 2

Customer: 94

Arrival time: 23:12:23

Response time: 23:12:53

Teller: 4

Customer: 91

Arrival time: 23:12:9

Completion time: 23:12:54

Teller: 4

Customer: 95

Arrival time: 23:12:24

Response time: 23:12:54

Teller: 3

Customer: 92

Arrival time: 23:12:10

Completion time: 23:12:55

Teller: 3

Customer: 96

Arrival time: 23:12:25

Response time: 23:12:55

Teller: 1

Customer: 93

Arrival time: 23:12:22

Completion time: 23:13:7

Teller: 1

Customer: 97

Arrival time: 23:12:37

Response time: 23:13:7
Teller: 2
Customer: 94
Arrival time: 23:12:23
Completion time: 23:13:8
Teller: 2
Customer: 98
Arrival time: 23:12:38
Response time: 23:13:8
Teller: 4
Customer: 95
Arrival time: 23:12:24
Completion time: 23:13:9
Teller: 4
Customer: 99
Arrival time: 23:12:39
Response time: 23:13:9
Teller: 3
Customer: 96
Arrival time: 23:12:25
Completion time: 23:13:10
Teller: 3
Customer: 100
Arrival time: 23:12:40
Response time: 23:13:10
Teller: 1
Customer: 97
Arrival time: 23:12:37
Completion time: 23:13:22
Teller: 2
Customer: 98
Arrival time: 23:12:38
Completion time: 23:13:23
Teller: 4
Customer: 99
Arrival time: 23:12:39
Completion time: 23:13:24
Teller: 3
Customer: 100
Arrival time: 23:12:40
Completion time: 23:13:25
Teller 1 serverd: 25
Teller 2 serverd: 25
Teller 3 serverd: 25
Teller 4 serverd: 25

Test 3

The other examples were of two edge cases, here it shows varying values. Because of this the distribution of served customers per teller, is not even, unlike the earlier examples. Therefor it shows the program working correctly.

Input

$m = 4$

$t_c = 2$

$t_i = 9$

$t_w = 1$

$t_d = 4$

Output

1: I

Arival Time: 23:32:7

Teller: 3

Customer: 1

Arrival time: 23:32:7

Response time: 23:32:7

2: W

Arival Time: 23:32:9

Teller: 2

Customer: 2

Arrival time: 23:32:9

Response time: 23:32:9

3: I

Arival Time: 23:32:11

Teller: 3

Customer: 1

Arrival time: 23:32:7

Completion time: 23:32:11

Teller: 3

Customer: 3

Arrival time: 23:32:11

Response time: 23:32:11

4: W

Arival Time: 23:32:13

Teller: 4

Customer: 4

Arrival time: 23:32:13

Response time: 23:32:13

Teller: 3

Customer: 3

Arrival time: 23:32:11

Completion time: 23:32:15

5: I

```

Arival Time: 23:32:15
-----
Teller: 1
Customer: 5
Arrival time: 23:32:15
Response time: 23:32:15
-----
6: I
Arival Time: 23:32:17
-----
Teller: 3
Customer: 6
Arrival time: 23:32:17
Response time: 23:32:17
Teller: 2
Customer: 2
Arrival time: 23:32:9
Completion time: 23:32:18
Teller: 1
Customer: 5
Arrival time: 23:32:15
Completion time: 23:32:19
-----
7: I
Arival Time: 23:32:19
-----
Teller: 2
Customer: 7
Arrival time: 23:32:19
Response time: 23:32:19
Teller: 3
Customer: 6
Arrival time: 23:32:17
Completion time: 23:32:21
-----
8: I
Arival Time: 23:32:21
-----
Teller: 1
Customer: 8
Arrival time: 23:32:21
Response time: 23:32:21
Teller: 4
Customer: 4
Arrival time: 23:32:13
Completion time: 23:32:22
Teller: 2
Customer: 7
Arrival time: 23:32:19
Completion time: 23:32:23
-----
9: I
Arival Time: 23:32:23
-----
Teller: 3
Customer: 9

```

Arrival time: 23:32:23
Response time: 23:32:23
Teller: 1
Customer: 8
Arrival time: 23:32:21
Completion time: 23:32:25

10: D
Arival Time: 23:32:25

Teller: 4
Customer: 10
Arrival time: 23:32:25
Response time: 23:32:25
Teller: 4
Customer: 10
Arrival time: 23:32:25
Completion time: 23:32:26
Teller: 3
Customer: 9
Arrival time: 23:32:23
Completion time: 23:32:27

11: I
Arival Time: 23:32:27

Teller: 2
Customer: 11
Arrival time: 23:32:27
Response time: 23:32:27

12: I
Arival Time: 23:32:29

Teller: 1
Customer: 12
Arrival time: 23:32:29
Response time: 23:32:29

13: D
Arival Time: 23:32:31

Teller: 2
Customer: 11
Arrival time: 23:32:27
Completion time: 23:32:31
Teller: 4
Customer: 13
Arrival time: 23:32:31
Response time: 23:32:31
Teller: 4
Customer: 13
Arrival time: 23:32:31
Completion time: 23:32:32
Teller: 1
Customer: 12

Arrival time: 23:32:29
Completion time: 23:32:33

14: W
Arrival Time: 23:32:33

Teller: 1
Customer: 14
Arrival time: 23:32:33
Response time: 23:32:33

15: I
Arrival Time: 23:32:35

Teller: 2
Customer: 15
Arrival time: 23:32:35
Response time: 23:32:35

16: I
Arrival Time: 23:32:37

Teller: 4
Customer: 16
Arrival time: 23:32:37
Response time: 23:32:37
Teller: 2
Customer: 15
Arrival time: 23:32:35
Completion time: 23:32:39

17: D
Arrival Time: 23:32:39

Teller: 3
Customer: 17
Arrival time: 23:32:39
Response time: 23:32:39
Teller: 3
Customer: 17
Arrival time: 23:32:39
Completion time: 23:32:40
Teller: 4
Customer: 16
Arrival time: 23:32:37
Completion time: 23:32:41

18: I
Arrival Time: 23:32:41

Teller: 2
Customer: 18
Arrival time: 23:32:41
Response time: 23:32:41
Teller: 1
Customer: 14

Arrival time: 23:32:33
Completion time: 23:32:42

19: D
Arrival Time: 23:32:43

Teller: 3
Customer: 19
Arrival time: 23:32:43
Response time: 23:32:43
Teller: 3
Customer: 19
Arrival time: 23:32:43
Completion time: 23:32:44
Teller: 2
Customer: 18
Arrival time: 23:32:41
Completion time: 23:32:45

20: W
Arrival Time: 23:32:45

Teller: 4
Customer: 20
Arrival time: 23:32:45
Response time: 23:32:45

21: I
Arrival Time: 23:32:47

Teller: 1
Customer: 21
Arrival time: 23:32:47
Response time: 23:32:47

22: D
Arrival Time: 23:32:49

Teller: 3
Customer: 22
Arrival time: 23:32:49
Response time: 23:32:49
Teller: 3
Customer: 22
Arrival time: 23:32:49
Completion time: 23:32:50
Teller: 1
Customer: 21
Arrival time: 23:32:47
Completion time: 23:32:51

23: D
Arrival Time: 23:32:51

Teller: 2
Customer: 23

Arrival time: 23:32:51
Response time: 23:32:51
Teller: 2
Customer: 23
Arrival time: 23:32:51
Completion time: 23:32:52

24: I
Arival Time: 23:32:53

Teller: 3
Customer: 24
Arrival time: 23:32:53
Response time: 23:32:53
Teller: 4
Customer: 20
Arrival time: 23:32:45
Completion time: 23:32:54

25: D
Arival Time: 23:32:55

Teller: 1
Customer: 25
Arrival time: 23:32:55
Response time: 23:32:55
Teller: 1
Customer: 25
Arrival time: 23:32:55
Completion time: 23:32:56
Teller: 3
Customer: 24
Arrival time: 23:32:53
Completion time: 23:32:57

26: W
Arival Time: 23:32:57

Teller: 2
Customer: 26
Arrival time: 23:32:57
Response time: 23:32:57

27: W
Arival Time: 23:32:59

Teller: 4
Customer: 27
Arrival time: 23:32:59
Response time: 23:32:59

28: I
Arival Time: 23:33:1

Teller: 1
Customer: 28

```

Arrival time: 23:33:1
Response time: 23:33:1
-----
29: D
Arival Time: 23:33:3
-----
Teller: 3
Customer: 29
Arrival time: 23:33:3
Response time: 23:33:3
Teller: 3
Customer: 29
Arrival time: 23:33:3
Completion time: 23:33:4
Teller: 1
Customer: 28
Arrival time: 23:33:1
Completion time: 23:33:5
-----
30: D
Arival Time: 23:33:5
-----
Teller: 3
Customer: 30
Arrival time: 23:33:5
Response time: 23:33:5
Teller: 2
Customer: 26
Arrival time: 23:32:57
Completion time: 23:33:6
Teller: 3
Customer: 30
Arrival time: 23:33:5
Completion time: 23:33:6
-----
31: I
Arival Time: 23:33:7
-----
Teller: 1
Customer: 31
Arrival time: 23:33:7
Response time: 23:33:7
Teller: 4
Customer: 27
Arrival time: 23:32:59
Completion time: 23:33:8
-----
32: I
Arival Time: 23:33:9
-----
Teller: 2
Customer: 32
Arrival time: 23:33:9
Response time: 23:33:9
Teller: 1
Customer: 31

```

Arrival time: 23:33:7
Completion time: 23:33:11

33: D
Arrival Time: 23:33:11

Teller: 3
Customer: 33
Arrival time: 23:33:11
Response time: 23:33:11
Teller: 3
Customer: 33
Arrival time: 23:33:11
Completion time: 23:33:12

34: I
Arrival Time: 23:33:13

Teller: 4
Customer: 34
Arrival time: 23:33:13
Response time: 23:33:13
Teller: 2
Customer: 32
Arrival time: 23:33:9
Completion time: 23:33:13

35: W
Arrival Time: 23:33:15

Teller: 1
Customer: 35
Arrival time: 23:33:15
Response time: 23:33:15
Teller: 4
Customer: 34
Arrival time: 23:33:13
Completion time: 23:33:17

36: D
Arrival Time: 23:33:17

Teller: 3
Customer: 36
Arrival time: 23:33:17
Response time: 23:33:17
Teller: 3
Customer: 36
Arrival time: 23:33:17
Completion time: 23:33:18

37: I
Arrival Time: 23:33:19

Teller: 2
Customer: 37

Arrival time: 23:33:19
Response time: 23:33:19

38: I
Arival Time: 23:33:21

Teller: 4
Customer: 38
Arrival time: 23:33:21
Response time: 23:33:21
Teller: 2
Customer: 37
Arrival time: 23:33:19
Completion time: 23:33:23

39: W
Arival Time: 23:33:23

Teller: 3
Customer: 39
Arrival time: 23:33:23
Response time: 23:33:23
Teller: 1
Customer: 35
Arrival time: 23:33:15
Completion time: 23:33:24
Teller: 4
Customer: 38
Arrival time: 23:33:21
Completion time: 23:33:25

40: I
Arival Time: 23:33:25

Teller: 2
Customer: 40
Arrival time: 23:33:25
Response time: 23:33:25

41: I
Arival Time: 23:33:27

Teller: 1
Customer: 41
Arrival time: 23:33:27
Response time: 23:33:27
Teller: 2
Customer: 40
Arrival time: 23:33:25
Completion time: 23:33:29

42: D
Arival Time: 23:33:29

Teller: 2
Customer: 42

Arrival time: 23:33:29
Response time: 23:33:29
Teller: 2
Customer: 42
Arrival time: 23:33:29
Completion time: 23:33:30
Teller: 1
Customer: 41
Arrival time: 23:33:27
Completion time: 23:33:31

43: W
Arrival Time: 23:33:31

Teller: 4
Customer: 43
Arrival time: 23:33:31
Response time: 23:33:31
Teller: 3
Customer: 39
Arrival time: 23:33:23
Completion time: 23:33:32

44: D
Arrival Time: 23:33:33

Teller: 2
Customer: 44
Arrival time: 23:33:33
Response time: 23:33:33
Teller: 2
Customer: 44
Arrival time: 23:33:33
Completion time: 23:33:34

45: W
Arrival Time: 23:33:35

Teller: 1
Customer: 45
Arrival time: 23:33:35
Response time: 23:33:35

46: D
Arrival Time: 23:33:37

Teller: 3
Customer: 46
Arrival time: 23:33:37
Response time: 23:33:37
Teller: 3
Customer: 46
Arrival time: 23:33:37
Completion time: 23:33:38

47: I

Arival Time: 23:33:39

Teller: 2

Customer: 47

Arrival time: 23:33:39

Response time: 23:33:39

Teller: 4

Customer: 43

Arrival time: 23:33:31

Completion time: 23:33:40

48: D

Arival Time: 23:33:41

Teller: 3

Customer: 48

Arrival time: 23:33:41

Response time: 23:33:41

Teller: 3

Customer: 48

Arrival time: 23:33:41

Completion time: 23:33:42

Teller: 2

Customer: 47

Arrival time: 23:33:39

Completion time: 23:33:43

49: W

Arival Time: 23:33:43

Teller: 4

Customer: 49

Arrival time: 23:33:43

Response time: 23:33:43

Teller: 1

Customer: 45

Arrival time: 23:33:35

Completion time: 23:33:44

50: W

Arival Time: 23:33:45

Teller: 3

Customer: 50

Arrival time: 23:33:45

Response time: 23:33:45

51: D

Arival Time: 23:33:47

Teller: 2

Customer: 51

Arrival time: 23:33:47

Response time: 23:33:47

Teller: 2

Customer: 51

Arrival time: 23:33:47
Completion time: 23:33:48

52: W
Arrival Time: 23:33:49

Teller: 1
Customer: 52
Arrival time: 23:33:49
Response time: 23:33:49

53: D
Arrival Time: 23:33:51

Teller: 2
Customer: 53
Arrival time: 23:33:51
Response time: 23:33:51
Teller: 4
Customer: 49
Arrival time: 23:33:43
Completion time: 23:33:52
Teller: 2
Customer: 53
Arrival time: 23:33:51
Completion time: 23:33:52

54: W
Arrival Time: 23:33:53

Teller: 4
Customer: 54
Arrival time: 23:33:53
Response time: 23:33:53
Teller: 3
Customer: 50
Arrival time: 23:33:45
Completion time: 23:33:54

55: W
Arrival Time: 23:33:55

Teller: 2
Customer: 55
Arrival time: 23:33:55
Response time: 23:33:55

56: I
Arrival Time: 23:33:57

Teller: 3
Customer: 56
Arrival time: 23:33:57
Response time: 23:33:57
Teller: 1
Customer: 52

Arrival time: 23:33:49
Completion time: 23:33:58

57: W
Arrival Time: 23:33:59

Teller: 1
Customer: 57
Arrival time: 23:33:59
Response time: 23:33:59
Teller: 3
Customer: 56
Arrival time: 23:33:57
Completion time: 23:34:1

58: D
Arrival Time: 23:34:1

Teller: 3
Customer: 58
Arrival time: 23:34:1
Response time: 23:34:1
Teller: 4
Customer: 54
Arrival time: 23:33:53
Completion time: 23:34:2
Teller: 3
Customer: 58
Arrival time: 23:34:1
Completion time: 23:34:2

59: W
Arrival Time: 23:34:3

Teller: 4
Customer: 59
Arrival time: 23:34:3
Response time: 23:34:3
Teller: 2
Customer: 55
Arrival time: 23:33:55
Completion time: 23:34:4

60: I
Arrival Time: 23:34:5

Teller: 3
Customer: 60
Arrival time: 23:34:5
Response time: 23:34:5

61: D
Arrival Time: 23:34:7

Teller: 2
Customer: 61

Arrival time: 23:34:7
Response time: 23:34:7
Teller: 1
Customer: 57
Arrival time: 23:33:59
Completion time: 23:34:8
Teller: 2
Customer: 61
Arrival time: 23:34:7
Completion time: 23:34:8
Teller: 3
Customer: 60
Arrival time: 23:34:5
Completion time: 23:34:9

62: D
Arrival Time: 23:34:9

Teller: 1
Customer: 62
Arrival time: 23:34:9
Response time: 23:34:9
Teller: 1
Customer: 62
Arrival time: 23:34:9
Completion time: 23:34:10

63: D
Arrival Time: 23:34:11

Teller: 2
Customer: 63
Arrival time: 23:34:11
Response time: 23:34:11
Teller: 4
Customer: 59
Arrival time: 23:34:3
Completion time: 23:34:12
Teller: 2
Customer: 63
Arrival time: 23:34:11
Completion time: 23:34:12

64: D
Arrival Time: 23:34:13

Teller: 3
Customer: 64
Arrival time: 23:34:13
Response time: 23:34:13
Teller: 3
Customer: 64
Arrival time: 23:34:13
Completion time: 23:34:14

65: I

```

Arival Time: 23:34:15
-----
Teller: 1
Customer: 65
Arrival time: 23:34:15
Response time: 23:34:15
-----
66: I
Arival Time: 23:34:17
-----
Teller: 4
Customer: 66
Arrival time: 23:34:17
Response time: 23:34:17
-----
67: D
Arival Time: 23:34:19
-----
Teller: 1
Customer: 65
Arrival time: 23:34:15
Completion time: 23:34:19
Teller: 2
Customer: 67
Arrival time: 23:34:19
Response time: 23:34:19
Teller: 2
Customer: 67
Arrival time: 23:34:19
Completion time: 23:34:20
Teller: 4
Customer: 66
Arrival time: 23:34:17
Completion time: 23:34:21
-----
68: D
Arival Time: 23:34:21
-----
Teller: 3
Customer: 68
Arrival time: 23:34:21
Response time: 23:34:21
Teller: 3
Customer: 68
Arrival time: 23:34:21
Completion time: 23:34:22
-----
69: I
Arival Time: 23:34:23
-----
Teller: 1
Customer: 69
Arrival time: 23:34:23
Response time: 23:34:23
-----
70: I

```

```

Arival Time: 23:34:25
-----
Teller: 2
Customer: 70
Arrival time: 23:34:25
Response time: 23:34:25
Teller: 1
Customer: 69
Arrival time: 23:34:23
Completion time: 23:34:27
-----
71: D
Arival Time: 23:34:27
-----
Teller: 4
Customer: 71
Arrival time: 23:34:27
Response time: 23:34:27
Teller: 4
Customer: 71
Arrival time: 23:34:27
Completion time: 23:34:28
Teller: 2
Customer: 70
Arrival time: 23:34:25
Completion time: 23:34:29
-----
72: W
Arival Time: 23:34:29
-----
Teller: 2
Customer: 72
Arrival time: 23:34:29
Response time: 23:34:29
-----
73: D
Arival Time: 23:34:31
-----
Teller: 1
Customer: 73
Arrival time: 23:34:31
Response time: 23:34:31
Teller: 1
Customer: 73
Arrival time: 23:34:31
Completion time: 23:34:32
-----
74: I
Arival Time: 23:34:33
-----
Teller: 4
Customer: 74
Arrival time: 23:34:33
Response time: 23:34:33
-----
75: I

```

Arival Time: 23:34:35

Teller: 3

Customer: 75

Arrival time: 23:34:35

Response time: 23:34:35

Teller: 4

Customer: 74

Arrival time: 23:34:33

Completion time: 23:34:37

76: I

Arival Time: 23:34:37

Teller: 1

Customer: 76

Arrival time: 23:34:37

Response time: 23:34:37

Teller: 2

Customer: 72

Arrival time: 23:34:29

Completion time: 23:34:38

Teller: 3

Customer: 75

Arrival time: 23:34:35

Completion time: 23:34:39

77: D

Arival Time: 23:34:39

Teller: 4

Customer: 77

Arrival time: 23:34:39

Response time: 23:34:39

Teller: 4

Customer: 77

Arrival time: 23:34:39

Completion time: 23:34:40

Teller: 1

Customer: 76

Arrival time: 23:34:37

Completion time: 23:34:41

78: W

Arival Time: 23:34:41

Teller: 2

Customer: 78

Arrival time: 23:34:41

Response time: 23:34:41

79: I

Arival Time: 23:34:43

Teller: 3

Customer: 79

```

Arrival time: 23:34:43
Response time: 23:34:43
-----
80: D
Arival Time: 23:34:45
-----
Teller: 4
Customer: 80
Arrival time: 23:34:45
Response time: 23:34:45
Teller: 4
Customer: 80
Arrival time: 23:34:45
Completion time: 23:34:46
Teller: 3
Customer: 79
Arrival time: 23:34:43
Completion time: 23:34:47
-----
81: D
Arival Time: 23:34:47
-----
Teller: 1
Customer: 81
Arrival time: 23:34:47
Response time: 23:34:47
Teller: 1
Customer: 81
Arrival time: 23:34:47
Completion time: 23:34:48
-----
82: W
Arival Time: 23:34:49
-----
Teller: 4
Customer: 82
Arrival time: 23:34:49
Response time: 23:34:49
Teller: 2
Customer: 78
Arrival time: 23:34:41
Completion time: 23:34:50
-----
83: D
Arival Time: 23:34:51
-----
Teller: 3
Customer: 83
Arrival time: 23:34:51
Response time: 23:34:51
Teller: 3
Customer: 83
Arrival time: 23:34:51
Completion time: 23:34:52
-----
84: W

```

Arival Time: 23:34:53

Teller: 1

Customer: 84

Arrival time: 23:34:53

Response time: 23:34:53

85: D

Arival Time: 23:34:55

Teller: 2

Customer: 85

Arrival time: 23:34:55

Response time: 23:34:55

Teller: 2

Customer: 85

Arrival time: 23:34:55

Completion time: 23:34:56

86: W

Arival Time: 23:34:57

Teller: 3

Customer: 86

Arrival time: 23:34:57

Response time: 23:34:57

Teller: 4

Customer: 82

Arrival time: 23:34:49

Completion time: 23:34:58

87: I

Arival Time: 23:34:59

Teller: 2

Customer: 87

Arrival time: 23:34:59

Response time: 23:34:59

88: I

Arival Time: 23:35:1

Teller: 4

Customer: 88

Arrival time: 23:35:1

Response time: 23:35:1

Teller: 1

Customer: 84

Arrival time: 23:34:53

Completion time: 23:35:2

Teller: 2

Customer: 87

Arrival time: 23:34:59

Completion time: 23:35:3

89: W

Arival Time: 23:35:3

Teller: 1

Customer: 89

Arrival time: 23:35:3

Response time: 23:35:3

Teller: 4

Customer: 88

Arrival time: 23:35:1

Completion time: 23:35:5

90: D

Arival Time: 23:35:5

Teller: 2

Customer: 90

Arrival time: 23:35:5

Response time: 23:35:5

Teller: 3

Customer: 86

Arrival time: 23:34:57

Completion time: 23:35:6

Teller: 2

Customer: 90

Arrival time: 23:35:5

Completion time: 23:35:6

91: W

Arival Time: 23:35:7

Teller: 4

Customer: 91

Arrival time: 23:35:7

Response time: 23:35:7

92: D

Arival Time: 23:35:9

Teller: 3

Customer: 92

Arrival time: 23:35:9

Response time: 23:35:9

Teller: 3

Customer: 92

Arrival time: 23:35:9

Completion time: 23:35:10

93: I

Arival Time: 23:35:11

Teller: 2

Customer: 93

Arrival time: 23:35:11

Response time: 23:35:11

Teller: 1

Customer: 89

Arrival time: 23:35:3
Completion time: 23:35:12

94: D

Arrival Time: 23:35:13

Teller: 3

Customer: 94

Arrival time: 23:35:13

Response time: 23:35:13

Teller: 3

Customer: 94

Arrival time: 23:35:13

Completion time: 23:35:14

Teller: 2

Customer: 93

Arrival time: 23:35:11

Completion time: 23:35:15

95: D

Arrival Time: 23:35:15

Teller: 2

Customer: 95

Arrival time: 23:35:15

Response time: 23:35:15

Teller: 4

Customer: 91

Arrival time: 23:35:7

Completion time: 23:35:16

Teller: 2

Customer: 95

Arrival time: 23:35:15

Completion time: 23:35:16

96: I

Arrival Time: 23:35:17

Teller: 3

Customer: 96

Arrival time: 23:35:17

Response time: 23:35:17

97: W

Arrival Time: 23:35:19

Teller: 1

Customer: 97

Arrival time: 23:35:19

Response time: 23:35:19

Teller: 3

Customer: 96

Arrival time: 23:35:17

Completion time: 23:35:21

98: I

```

Arival Time: 23:35:21
-----
Teller: 4
Customer: 98
Arrival time: 23:35:21
Response time: 23:35:21
-----
99: I
Arival Time: 23:35:23
-----
Teller: 2
Customer: 99
Arrival time: 23:35:23
Response time: 23:35:23
Teller: 4
Customer: 98
Arrival time: 23:35:21
Completion time: 23:35:25
-----
100: I
Arival Time: 23:35:25
-----
Teller: 3
Customer: 100
Arrival time: 23:35:25
Response time: 23:35:25
Teller: 2
Customer: 99
Arrival time: 23:35:23
Completion time: 23:35:27
Teller: 1
Customer: 97
Arrival time: 23:35:19
Completion time: 23:35:28
Teller: 3
Customer: 100
Arrival time: 23:35:25
Completion time: 23:35:29
Teller 1 serverd: 22
Teller 2 serverd: 28
Teller 3 serverd: 29
Teller 4 serverd: 21

```