

Title:

# **An Introduction to Simulation and Modeling (HW4)**

**Author :**

Adib Rastegarnia

**Email:**

arastega@purdue.edu

**Fall 2014**

## 1- Design and Implementation of a Basic CSIM

In this homework, I have tried to implement a basic version of CSIM simulator. In order to implement this simulator, pthread library and C programming language is used. A short description about my design is provided as follows:

A main function in the library is defined in order to initialize and create a thread for Sim and then yields control to the Sim thread.



After the Sim thread is created, main function returns back to sim function. In this situation, sim continues running. If you call create function for the customer (after the sim thread is created), the create function for the customer is called in the library and then a thread will be created for that customer, and after that this customer push into the calendar (with current time as a parameter) and then it yields to the Sim function again to continue. In this situation customer thread is suspended (by using mutex) till the scheduler take it out from the calendar and runs it again.

A diagram is drawn in order to show above description that you see as follows:

**Create function is called from Sim for a customer. Customer is created and push into the calendar with current time as a variable.**



**Customer pushed into the calendar and suspended till scheduler decides to remove it from calendar.**

In order to implement the calendar, heap data structure is used. In addition, a scheduler is implemented in order to pop each of the threads (customers or sim) that we pushed them into the calendar. The scheduler checks the calendar frequently in order to see there is anything in the calendar to execute or not. If there are more than one entity in the calendar (customers or

sim ), the next minimum time between entities is selected to execute (When an entity is pushed into the calendar a time variable is added as well which shows current time ). Suppose that scheduler checks the calendar and find a customer that should be executed. In this case, scheduler pop it from calendar, then scheduler yields control to the customer in order to execute cust() function in user program. In other words, scheduler is used in order to check the calendar frequently in order to find the entity which has the minimum time value and pop it from calendar for running.

In order to implement hold function, the current thread that is running is determined by using pthread\_self function and then push into calendar till scheduler removed it again from the calendar and execute it again. In order to suspend thread execution in the hold function, mutex is used.

## **2- A short description about source codes:**

There are 3 files in the folder of Asim which are described as follows:

- 1- Main.cpp ( User Interface part in order to define simulation problem)
- 2- Asim.cpp (Body of functions which we have defined in Asim.h)
- 3- Asim.h ( defining function protoptypes)