

## **Report for Assignment Question 2:**

### **Automating Biryani Serving**

Problem solved using concept of multithreading and synchronization.

Each thread for table , student , robot chef is created and made to work in such a way that deadlock don't occur.

#### **Logic flow for robot chefs:**

After the threads for all the robots are created, it does the following untill all the students are successfully served biryani for the day.

- 1) It generates random numbers  $w$  = Time after which biryani will be prepared ,  $p$  = Capacity of each vessel ,  $r$  = Number of vessels.
- 2) Goes to sleep for  $w$  seconds after which it prints "biryani prepared".
- 3) Calls the function `biryani_ready()` which searches for a table for each of the  $r$  vessels made. It basically makes some flag change value to let the tables know that is ready to give a vessel and executes `pthread_cond_wait()` to wait till a table takes the vessel and signals it executing `pthread_cond_signal()`.
- 4) As soon as all the  $r$  vessels get empty , it checks if all the students have been served or not . If yes then the function exits else it continues to make a new batch of biryani by going to step 1.

#### **Logic flow for tables:**

After the thread for tables is created this is what happens:

- 1) Each table keeps on polling through all the chefs to check if it's ready to give a vessel or not. If yes then it signals that chef so tht it doesn't wait anymore.

2) Once the vessel is loaded it executes `pthread_cond_wait()` and waits untill  $x = (\text{some random number})$  students come and take biryani from that table.

3) The last person to have biryani from that table signals the table to stop waiting.

4) It then executes 1.

### **Logic flow for students:**

Once the thread for student gets created it does the following:

1) Keep polling through all the tables to check which one is in serving mode.

2) If such a table is found, print that biryani was had from that table and exit from the thread. If the person to take biryani from a table is the last person , then he/she should signal the table to stop waiting and continue.

The main function joins the threads for students and then exits .

i.e. The program exits only when all the K students have been served biryani.