CSC 3370 – Clojure Project

The purpose of this assignment is to give you experience building simple programs in Clojure.

The Pizza Palace

> The Pizza Palace is a VERY small restaurant. The restaurant has one table and a waiting room with a number of chairs in it. When the chef finishes feeding a customer, he dismisses the customer and then goes to the waiting room to see if there are other customers waiting. If there are, he brings one of them back to the table and makes him food. If there are no other customers waiting, he returns to his kitchen and takes a nap.
>
> Each customer, when he arrives, looks to see what the chef is doing. If the chef is sleeping, then the customer wakes him up and sits in the chair. If the chef is cooking food, then the customer goes to the waiting room. If there is a free chair in the waiting room, the customer sits in it and waits his turn. If there is no free chair, then the customer leaves.
>
> Based on a naïve analysis, the above description should ensure that the restaurant functions correctly, with the chef preparing food for anyone who arrives until there are no more customers, and then sleeping until the next customer arrives. In practice, there are a number of problems that can occur that are illustrative of general scheduling problems.
>
> The problems are all related to the fact that the actions by both the chef and the customer (checking the waiting room, entering the restaurant, taking a waiting room chair, etc.) all take an unknown amount of time. For example, a customer may arrive and observe that the chef is feeding a customer, so he goes to the waiting room. While he is on his way, the barber finishes the meal he is doing and goes to check the waiting room. Since there is no one there (the customer not having arrived yet), he goes back to nap. The chef is now waiting for a customer and the customer is waiting for the chef. In another example, two customers may arrive at the same time when there happens to be a single seat in the waiting room. They observe that the chef is feeding a customer, go to the waiting room, and both attempt to occupy the single chair.

Implementation

You will be creating a Clojure program to simulate the pizza palace problem.

Notes:

- You will want to make an agent representing the chef
- You will also want to make an atom presenting seats to wait in (there are 3)

- You will need to define a bunch of functions (it is functional programming after all). Suggestions include:
    - anyone-waiting: Check to see if anyone is waiting in the 3 seats or not
    - free-seat: Function to free up a seat with someone waiting when their turn comes
    - try-to-wait: Attempt to wait in a seat if one is available
    - serve-food: If someone is waiting bring them in and feed them. This takes 20 “minutes” (use 20 ms). Increase the count of number of people fed
    - continue-running: Hardcode the true atom to this for simplicity
    - move-to-kitchen: As long as continue running is true, sleep for 1 “minute” (ms) to avoid resource issues then have the chef try to feed another customer. This should be in a loop.
    - customer-arrive: Loop to have a new customer arrive every 10 to 30 “minutes” (ms).
- Your main program should start up the move-to-kitchen and customer-arrive code using futures. It should then sleep the main thread for the “10 hours” the restaurant is open (10,000 ms). Finally print out the value associated with the chef (number of people fed), reset continue-running to false, and shutdown the agent.

Grading Rubric

- Code implemented properly and working (100 points)

Submission

You must upload a copy of your work through the portal. If you have any issues or questions please let me know.