GIT Department of Computer Engineering
CSE 344 - Spring 2020
MIDTERM Report

Esra EMİRLİ
151044069

1 Synchronization Problems And Solutions

The problems encountered were explained by the function of semaphores.

- Actors / sem_supplier : We have one supplier and our kitchen, which has as much space as K, can come back to the supplier one after another because the processes do not have any working order, and in this case the kitchen could be exceeded. Therefore, the supplier semaphore was used and the kitchen size was given as the first value. Each time the supplier enters the kitchen, the semaphore is given a wait order, so that the supplier's entrance to the kitchen is blocked when the kitchen is full. As the cook came to the kitchen and bought a plate, she posted it to the supplier and the access of the supplier to the kitchen was provided in this way
- Actors / sem_cook : Even though the supplier should work first, any process will want to process. If there is a cook who wants to make the first process, it will try to buy a plate from the empty kitchen. To prevent this situation, the cook was given 0 as the first value. As Supplier put plates in the kitchen, it posted all the cooks. But this time, if the same cooks entered more than one kitchen, it was possible to buy as many dishes as there were not in the kitchen. As a solution to this, a wait for cooks was placed at the entrance of the kitchen, and each time the cook entered, it was prevented from entering unless the post came again from the supplier.
- Kitchen / mutex_kitchen : While the supplier put plates in the kitchen, the cook buying plates at the same time caused confusion. For this reason, who first entered the kitchen, blocked the kitchen entrance with the mutex, and then when it was done, it opened it.
- Counter / sem_desert , sem_soup , sem_main_course : The plates available to students at the counter are limited. If the cook does not perform some checks while buying these plates, deadlock may occur.
    - Can these controls be sweet on the counter? If it is put, it is checked whether there is dessert in the kitchen and dessert is brought to the counter according to the situation.
    - Can these controls be placed on the counter with soup? If it is placed, it is checked whether there is soup in the kitchen and dessert is brought to the counter according to the situation.

- Can these controls be added to the counter as a main course? If it is put, it is checked whether there is a main course in the kitchen and dessert is brought to the counter according to the situation.

Semaphores were also used to understand whether these plates could be placed on the counter. These semaphores were assigned a value according to a rule, and then the corresponding semaphore was reduced for each plate placed on the counter. A 0 means that it cannot be placed on the counter. Therefore, the value will remain 0 until the student gets that plate from the counter.

Rules

!! The value of the plates can be at least S / 3.

!! An extra 1 is added to S%3 plates.

S : 3 -> P: 1 C: 1 D: 1 So here is 0

S : 4 -> P: 1 C: 1 D: 2 So here is 1

S : 5 -> P: 1 C: 2 D: 2 So here is 2

S : 6 -> P: 2 C: 2 D: 2 So here is 0

- Counter / sem_cook : Since the counter is empty, people with kitchen and counter access should be aware of it. When the student gets his food from the counter, he informs that 3 plates are open and ensures that the kitchen comes to the counter. The maximum value is set to counter size, so that the counter size does not exceed.
- When the cook puts the ingredients evenly on the counter, students can get three kinds of food at the same time. When they take 3 meals, they send their semaphores to their food and come back again.
- Counter / mutex_counter: While the cooks put plates in the counter, the student buying plates at the same time caused confusion. For this reason, who first entered the counter, blocked the counter entrance with the mutex, and then when it was done, it opened it.

- Counter / sem_table : When a student sits at a table, the empty table should decrease, so that there is no intersection with other students. In this case, it was solved with semaphore that first value is table size. When the table is empty, the empty table value is increased by posting a post to the semaphore.
- Actor / sem_students : Since the program cannot get anything from the counter as it works, its first value is assigned to 0. When the ingredients were put on the counter by the cooks, all students were posted and with the wait at the entrance of the bench, M students were provided inside.
- Actor / supplier_finish, flagSup : It means that all the plates that the Supplier has to end are in the kitchen. It was the supplier that warned the cooks to get ingredients from the kitchen. If it ends, the cooks warns themself when he leaves the plates on the counter and goes back to the kitchen and takes the ingredients.