COMP 304 PROJECT 2

Bora Berke Şahin - bsahin17@ku.edu.tr - 64060
Büşra Işık - bisik18@ku.edu.tr - 69016

Link to Repository: https://github.com/boraberke/comp304-fall22-project2

ALL OF OUR PROJECT PARTS ARE WORKING PROPERLY.

Part 1

Our implementation consists of 5 queues and 1 linkedlist. For each task, we have a queue. For each queue, we also have created a mutex. Whenever we check anything related to that queue, mutex is locked and after our job is done, it is unlocked. We used a control thread to add new gift types and place those gifts and their corresponding tasks to the corresponding queues.

To handle gift type 1,2,3; elfA, elfB and Santa check their queues and dequeue from their prioritized queues. For instance, elfA and elfB first check the packaging queue, then check either painting or assembly. Similarly for Santa, Santa first checks if there is any delivery job, then checks QA.

To handle gift type 4 and 5 we created a new data structure - linked list. The first two tasks needed for the gift type 4 or 5 are added to their corresponding queues just like every other task. After a task is completed for a gift type 4 or 5, the thread function that completed it checks whether that gift with that ID exists in the list. If it doesn't exist, it creates a gift struct and initializes that task's value as 1 indicating as it is completed. If it does exist, the gift struct gets updated with that task's value being 1. Thus, when elf A or B is going to do packaging, it first checks the linked list and looks for ready gifts to be packaged (both required tasks completed). This assumes that gift type 4 and 5 will have been waiting for a longer time and will get their packaging done first. If no such gift ID exists, the thread will continue looking through the queues for a task to do.

Part 2

We updated the probabilities of the gifts, type 1 having a 50% probability. We added a new function to linked list called WaitingQA, it returns the number of gifts that are waiting for QA. So, santa thread checks whether there are more than 3 QA tasks waiting for gift types 4 and 5 and whether the delivery queue is empty before deciding which task to do.

Part 3

We have added two new fields to the queue struct. These are as follows:

```
int isGiftFromNewZealand;  // Boolean to check if there is a gift for NZ
Task giftFromNewZealand;
```

With the help of these two new fields, we always check if any prioritized gifts are ready in the queue and put them here. Our workers will first check these fields and start with this gift if there is any.

We also updated linked list in a way that if gives priority to the gifts from New Zealand. We added fields to task and gift structs indicating whether the gift and their tasks are form New Zealand. So when a thread asks for a task from the linked list (for packaging type 4 and 5 gifts), it returns the one that is from New Zealand. If no such gift exists, it returns a ready gift that is not from New Zealand.

Lastly in the main project file, before doing any task the threads check whether there is a gift waiting from New Zealand in other queues. For example elf A checks whether painting queue has any waiting gifts from New Zealand. If there is, it does that task without going into packaging. This works because realistically there won't be a packaging task waiting in the queue or the linked list if there is a task from New Zealand in painting. Packaging will always need to be done after the previous tasks are done and a gift from New Zealand will always complete its tasks before another gift from New Zealand arrives.

Keeping Logs:

We have added three log files in our zip file as well as to our github page for each part.