|  |  |
| --- | --- |
| **Item** | **Answer** |
| **Section** | G1 |
| **Name(s)** | *Adeline Chin Wen Jie*  *Gokarn Mallika Nitin* |
| **Brief description of project** | *Our project is a restaurant simulator, for the restaurant Gluttons Bay, wherein there are a minimum of 1 waiter and 2 chefs employed. Additionally, only 5 patrons can be eating in the restaurant at a time. Due to multiple constraints, the restaurant can employ a maximum of 4 waiters, and 3 chefs, and can thereafter seat only 9 patrons at a time.*  *As a high class vegan restaurant, we serve a 5 course vegan meal consisting of soup(Miso Soup), salad(Asian Pear Maple Walnut Salad), appetiser(Stuffed Asian Eggplant with Vegan Sushi), Main course(Vegan Hot Pot) and dessert(Dark Chocolate Jin Deui).* |
| **Justification for multi-threading** | *Each waiter can serve a customer only one course at a time. Once the course is completed, he or she must wait for the chef to complete cooking the next course and thereafter the waiter to serve this course. However, there are a minimum of 5 and a maximum of 10 customers. Also, each customer is tended to by one and only one waiter. This leads the need for concurrency.*  *A single threaded program would have only one waiter working at a time which would lead to hungry angry and impatient customers who might leave their meal midway.* |
| **Transactional integrity** | *Were there any potential race conditions; if so, how did you resolve them? Or are there still unresolved race conditions that you are aware of?* |
| **Performance** | *Is there anything noteworthy you did that improved performance? (include stats if any)* |
| **Evidence of exploration** | *Is there anything noteworthy that you did that required additional research (of topics not covered in class)?* |
| **Innovation** | *Why is your project innovative?* |
| **Adherence to coding conventions & good practices** | *Is there anything noteworthy that you want to mention? E.g. usage of a well-known multi-threading “best-practice”, or OO design pattern etc.* |
| **References/Acknowledgement** | *We used third party code from github user heidtJJ (Jared Heidt) as our starting point and built on it*  *The code to his project is in the folder titled reference.*  *We have also used the StopWatch* |