Project description

First: project Modules:-

1. Data structures: (discussed in phase 1)
   1. Linked List
   2. Priority queue
   3. Queue.
2. Load class: (discussed in phase 1)
   1. Has all the input file data as data members
   2. Includes functions that:
      1. take the file name as a string and processes it to fill the data members.
      2. Instantiate objects of all the events in the simulation.
      3. Take a pointer to restaurant class and add all events in the events’ queue in restaurant class.
3. Events:
   1. Event (Parent):
      1. Event time, order Id 🡺 data members
      2. Execute 🡺 virtual function
   2. Arrival event(child):
      1. Distance, type, region, money,.. 🡺 data members
      2. Setters & getters
      3. Execute function: allocates new order and adds it to its proper queue in restaurant (it uses a pointer to restaurant).
   3. Cancellation event(child):
      1. Execute function calls cancellation functions from restaurant that searches for the order(Normal ones only) by its ID and deletes it from its queue.
   4. Promotion event(child):
      1. Extra money 🡺 data member
      2. Execute function calls promoteToVIP function in restaurant which takes the order ID and extra money and promotes the order to VIP by incrementing its money and changing its type then setting a priority equation to it and finally adds it to the VIP queue in its region.
4. Order class:
   1. ID, type, region, money, distance, AT, ST, FT, WT & priority 🡺 data members
   2. Setters & getters
   3. Operators overloading
5. Motorcycle class:
   1. Id,speed ,region, status, orderID
   2. Setters & getters
   3. Assign : assigns a motorcycle to a certain order.
   4. deAssign: resets the motorcycle to IDLE mode.
6. Restaurant Class: [The maestro]
   1. Data members:
      1. Pointer to GUI
      2. Enum program\_mode (interactive, step by step, silent)
      3. Events queue
      4. Number of total waiting orders
      5. Number of normal, frozen, vip waiting orders in each region individually
      6. number of motorcycles in each region(using normal queues)
      7. queues for all order types in each region( using normal queues for frozen orders, priority queues for VIP orders and Linked Lists for normal orders)
      8. priority queues for occupied motorcycles
      9. autoS (time limit for auto promotion)
      10. speeds of motorcycles.
   2. Functions:
      1. Constructor: initializes all numbers with zero and opens the output file.
      2. Add event: used in load function to add events to the events queue
      3. Execute events: loops on the events queue and executes the events for the current time step
      4. Add to queue functions: adds a certain order to a certain queue
      5. Gett order functions : gets the front order from a certain queue
      6. Cancel order: loops on all normal queues of the four regions and cancels the order with the given ID
      7. Set motorCycles function: Allocates motorcycles with the given number from the load function for each region
      8. Promote to vip: promotes a certain order to VIP
      9. Draw one queue: draws a queue to screen
      10. Delete order each time step: loops on all order queues and assigns any available order to any available motorcycle
      11. Simulation function: calls load functions, calls set motorcycles, loops on events queue and executes any available event each time step and increments the time step each iteration, calls auto promotion each iteration(time step)
      12. Other auxiliary functions.

Other Ideas:

* Customized the GUI.
* Adding sound in the background.