CSP2151 Assignment

This assignment is an extension of the Workshops you have been doing all semester. Where appropriate, the standards set out in previous workshops should be followed and the code must make good use of functions, structures and logical control flow.

Design and Implement a complete drone delivery simulation program.

The program should first load the `drone_t` data from a database file named "`Drones.dro`". The loaded data should be stored in a **linked list** of `drone_t` structures. The program should then load the `delivery_t` data from a database file named "`Deliveries.dro`". The loaded data should be stored in a linked list of `delivery_t` structures. These structures should follow the following specifications:

```
drone_t:
  name
  top_speed
  acceleration
  max_load
```

```
delivery_t:
  name
  quantity
  weight_per_item
  destination_latitude
  destination_longitude
```

Next, the program should display a menu system that allows the user to navigate through and execute the following options:

- Set Warehouse Location
- New Delivery
- Dispatch
- Exit
  - Save and exit
  - Exit without saving

The "Set Warehouse Location" should allow the user to set the latitude and longitude of the warehouse that the drones will be dispatched from.

The "New Delivery" option will prompt the user to enter details for a new `delivery_t` item to be added to the end of the linked list of queued deliveries.

The "Dispatch" option will process the next delivery. The user will be prompted to select an appropriate drone to make the delivery. The program will then calculate the total time taken for the delivery (as per workshop 3), output that value to the screen and save all relevant information about the delivery (drone name, delivery details, destination and time) in a file called "`Dispatch_Log.txt`". After processing a delivery, it should be removed from the list. Deliveries should be dispatched in FIFO order (First In First Out). If no drone is capable of carrying the delivery, an error message should be printed and the user returned to the menu.

The exit command should give the user the choice to save the current delivery queue back to the "`Deliveries.csv`" so it can be loaded the next time the program is executed.

Additional Details:

- When the "Save and exit" option is selected, the delivery list should be saved in the "`Deliveries.dro`" file in the correct format to be reloaded.
- Sample .dro files will be provided on blackboard. These are only samples. You may modify them or provide your own so long as they remain in the same format.
- No example executable file will be provided

Bonus Marks:

Additional work and effort to improve the program and make it more useful can be worth bonus marks. Bonus marks are not required and will not improve your grade above 100%

Examples of features that may be worth bonus marks:

- Improve the physics involved to use forces rather than raw acceleration and consider the effect of increased mass from heavier deliveries.
- Create a priority queue where higher priority deliveries are processed before lower priority ones.
- Allow drones to make multiple deliveries in one trip.
- Add colours or use more advanced graphics such as nCurses or the WinAPI.
- Ask your tutor if you have other ideas!

NOTE: The Workshops have been working up to this point to provide you with a base program of your own work that should give you the best possible opportunity for completing this assignment. You should look to how these Workshops have developed your work for a guide on how to structure your final assessment. No sample executables will be given for the final assignment.

Marking Guide:		
Design	/4	
Program performs calculations correctly	/2	
Appropriate use of functions and abstraction	/7	
Correct use of data structures (structs, linked lists, etc)	/7	
Code is written legibly (formatting, comments, etc)	/5	
Program compiles and runs correctly	/3	
Output is neat and formatted	/2	
Bonus	/5	
Inappropriate use of Global Variables or GOTO statements (Very bad don't do this)	/-10	
Total:	/30	