Project Data Structures

<table>
<tr><th>List Name</th><th>Chosen Data Structure</th><th>Justification</th></tr>
<tr><td>Event List</td><td>Queue</td><td>Events are executed in their order of insertion (first in first out) so to be able to access the first element fast a Queue is used.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert an Event</td><td>O(1)</td></tr>
<tr><td>Remove Front Event</td><td>O(1)</td></tr>
<tr><td>Access Front Event</td><td>O(1)</td></tr>
<tr><td>Get Count of Events</td><td>O(1)</td></tr>
</table></td></tr>
<tr><td>Ready Truck Normal
Ready Truck Special
Ready Truck VIP</td><td>Queue</td><td>Three lists are used each one for a type (Normal, Special, VIP), as the concept of first truck ready will be first used (first in first out) a Queue is used.
Accessing the front element in O(1) is good as the front element of each type will be checked repeatedly each hour.
Three Lists are used to easily check whether there are available trucks each hour of specific type by using Get Size Function to know the available types.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Truck</td><td>O(1)</td></tr>
<tr><td>Remove Front Truck</td><td>O(1)</td></tr>
<tr><td>Access Front Truck</td><td>O(1)</td></tr>
<tr><td>Get Count of Trucks</td><td>O(1)</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td>Maintained Truck Normal
Maintained Truck Special
Maintained Truck VIP</td>
<td>Queue</td>
<td>We will need to access the front element of each type of trucks being maintained each hour to know whether it finished its maintenance duration or not, so accessing the first element that comes to maintenance is important as it will be the first element to finish its maintenance (each type of trucks are grouped together because they have same maintenance time), consequently O(1) complexity for accessing and removing front Truck is required. In addition, the respect of order of insertion is needed (first in first out), so a queue is used.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Truck</td><td>O(1)</td></tr>
<tr><td>Remove Front Truck</td><td>O(1)</td></tr>
<tr><td>Access Front Truck</td><td>O(1)</td></tr>
<tr><td>Get Count of Trucks</td><td>O(1)</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td>Loading Truck Normal
Loading Truck Special
Loading Truck VIP</td>
<td>Priority Queue</td>
<td>Trucks being loaded list need to sort trucks according to the maximum load time of cargos each truck carries, so that the first truck finishes loading its cargos will be removed from loading truck list to in trip truck list, so most suitable data structure is priority queue to access top element(first one to finish) in O(1) and insert element being sorted in O($log\ (N)$) and Overall insertion of N elements with sorting in O($Nlog\ (N)$)
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Truck</td><td>O($log\ (N)$)</td></tr>
<tr><td>Remove Front Truck</td><td>O(1)</td></tr>
<tr><td>Access Front Truck</td><td>O(1)</td></tr>
<tr><td>Get Count of Trucks</td><td>O(1)</td></tr>
</table>
</td>
</tr>
<tr>
<td>In Trip Normal Truck
In Trip Special Truck
In Trip VIP Truck</td>
<td>Priority Queue</td>
<td>Those are lists for moving trucks, each list for each type of trucks need to be sorted according to truck trip finish time so that the first one finishes will be the first one to be removed from this list, So top element with least trip time will be accessed to check whether the truck finished or not, thus a priority queue will be used to access top Truck (with least time) in O(1) and Insert a Truck in O($log\ (N)$)
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Truck</td><td>O($log\ (N)$)</td></tr>
<tr><td>Remove Front Truck</td><td>O(1)</td></tr>
<tr><td>Access Front Truck</td><td>O(1)</td></tr>
<tr><td>Get Count of Trucks</td><td>O(1)</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td>Normal Waiting Cargo</td>
<td>General linked list</td>
<td>It was used to add or remove a cargo from a certain position.
A derived specialization from a general linked list used to add a function to return the cargo and remove from list without deleting it.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Cargo</td><td>O(1)</td></tr>
<tr><td>Remove Cargo</td><td>O(1)</td></tr>
<tr><td>Insert end</td><td>O(1)</td></tr>
<tr><td>Clear the list</td><td>O(n)</td></tr>
<tr><td>Get a cargo from id</td><td>O(n)</td></tr>
<tr><td>Remove from list then return cargo</td><td>O(n)</td></tr>
</table>
</td>
</tr>
<tr>
<td>Special waiting cargo</td>
<td>Queue</td>
<td>list is used as the concept of first Cargo ready will be first used (first in first out) a Queue is used.
Accessing the front element in O(1) is good as the front element of each type will be checked repeatedly each hour.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Cargo</td><td>O(1)</td></tr>
<tr><td>Remove Front Cargo</td><td>O(1)</td></tr>
<tr><td>Access Front Cargo</td><td>O(1)</td></tr>
<tr><td>Get Count of Cargo</td><td>O(1)</td></tr>
</table>
</td>
</tr>
<tr>
<td>VIP waiting list</td>
<td>Priority Queue</td>
<td>Each cargo will have a priority to get loaded using a priority equation.
The first Cargo to be used will be the one with highest Weight (calculated from priority equation).
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Cargo</td><td>O($log\ (N)$)</td></tr>
<tr><td>Remove Front Cargo</td><td>O(1)</td></tr>
<tr><td>Access Front Cargo</td><td>O(1)</td></tr>
<tr><td>Get Count of Cargo</td><td>O(1)</td></tr>
</table>
</td>
</tr>
</table>

<table>
<tr><td></td><td></td><td></td></tr>
<tr><td>Delivered Cargo Normal
Delivered Cargo Special
Delivered Cargo VIP</td><td>Queue</td><td>Separate lists are used to ease calculation in output file and easy access for each type of cargos separately.
Queue is chosen as the complexity of each operation is low, thus better performance.
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Cargo</td><td>O(1)</td></tr>
<tr><td>Remove Front Cargo</td><td>O(1)</td></tr>
<tr><td>Access Front Cargo</td><td>O(1)</td></tr>
<tr><td>Get Count of Cargo</td><td>O(1)</td></tr>
</table></td></tr>
<tr><td>Moving cargo</td><td>Priority Queue</td><td>This list will be sorted by CDT(cargo delivery time) as first cargo to be delivered will be at top to be dequeued
<table>
<tr><td>Operation</td><td>Complexity</td></tr>
<tr><td>Insert a Cargo</td><td>O($log\ (N)$)</td></tr>
<tr><td>Remove Front Cargo</td><td>O(1)</td></tr>
<tr><td>Access Front Cargo</td><td>O(1)</td></tr>
<tr><td>Get Count of Cargo</td><td>O(1)</td></tr>
</table></td></tr>
</table>