**Class: Packet**

* void AddData( T data); where T is any data type; updates parameter count, type table, and puts the data into the packet
* void Finalize(); prepares the packet to be sent, MUST be called before sending via network, AddData(T data) will \*not\* add data after finalization
* unsigned char\*\_get\_packet(); returns a pointer to the final packet
* Packet(unsigned char\* head); the default constructor, the head contains 1) packet type 2) hardware ID and 3) hardware address

**Class: Network**

* Constructed directly after definition( \_network\_interface ), user should not be instantizing another network object
* Void SendPacket(Packet\* letter); sends a packet via \_network\_interface, accepts a pointer to the Packet object (used for efficiency; so long as programmer uses new and delete keywords when constructing packets)
* Void ReconstructPacked(unsigned char\* data); accepts pointer to start of packet buffer, reconstructs the buffer and displays result in console

**Class: NetObject**

* virtual void receive(unsigned char\* data); implemented by all child classes in order to receive data for network, accepts pointer to packet stream BEGINNING AT PARAMETER COUNTER

**@TODO: modify for clarity**