Robot Class

Private members;

* String, serial number, fixed length, following the format
  + DDD MM #####
  + where DDD is the department/field of the robot such as WFH for wafer handlers
  + MM is model identifier such as A1, A2, B7, BB, etc
  + and ##### is serial number assigned to the individual robot.
* String, model name
  + the name of the model identified by the model identifier in the serial number key
  + example: A1 could be Model A version 1, but only Model A is stored as model name
  + more creative example: S1, S2, etc can all be Single-Axis but different iterations
* String, alias, or alternative name
  + for some projects, certain robots can be designated a name
  + this name could shared among different robots as part of a bigger projects
  + or it could be just a code name for prototypes
  + example: project Atlas which involves several robots from different departments and models, each robot involved would have the Atlas alias
* String, comments
  + comments are used to keep record of the last know status of the robot
  + example: 12/12/2012 RMA Z-Axis motor failure
  + example: 10/11/2012 Refurbished
  + etc
* String, production date, formated
  + Month/Day/Year
  + Date robot first exited production facility.

Public members;

* Constructor(SNR, model, alias = “”, comments = “”, production date = “”)
  + defaults for last 3 strings are empty strings
* Destructor if allocating memory on heap.
* Getters for all 5 fields
* Setters for last 3 fields (primary and secondary should never change after object creation)
* PrimaryCompare(Robot lhs, Robot rhs);
  + used in BST as function to compare two robots by their primary keys
* SecondaryCompare(Robot lhs, Robot rhs);
  + used in BST as function to compare two robots by their secondary keys