18-642 Project 3 Checklist

Name:

❏ **1) Variables and procedures have minimum scope**
It is fine to define all variables at the front of the procedure
Defining at smaller scope (such as within a "{}" block) is at your discretion

❏ **2) All variables are automatic**
Allocated on the stack unless required to be permanently allocated

❏ **3) All variables use strongest and simplest type possible, with no floats**
Add extra typedefs at your discretion

❏ **4) All base types are from stdint.h**
Variable size is at your discretion within reason

❏ **5) Geometric pairs (e.g. (X,Y)) are coupled using typedefs**

❏ **6) #define is not used**

❏ **7) "Magic numbers" are not used**
Use const for single values and enum for sets of related values

❏ **8) Switch statements are used to decide among enum values rather than if/else if**

❏ **9) Every switch statement has a default error handling clause**
Activates ROS_ERROR

❏ **10) Multiple switch statements in a single function, procedure, or method are strictly forbidden.** This also means nested switch statements are forbidden.

❏ **11) Every variable has a meaningful name**
Does not require explanation to someone else

❏ **12) Code is commented**
At a minimum, comment each function: purpose, inputs, outputs, saved internal state

❏ **13) Code conforms to Spaghetti Factor guidelines**
(number of globals: ___)*5 + SCC: ___ + (SLOC: ___)/20 ⇒ SF: ___
in single most complicated procedure/function with highest SF (NOT the whole file.)
Only count globals referenced by the procedure in question.
Switch statements count as +1 for SCC no matter how many cases are included.
Be ready to defend complexity above SF=10 as being absolutely necessary.

❏ **14) All the code is in a single .cpp file**

❏ **15) All conditionally executed statements are enclosed by "{}"**

❏ **16) All indentation and similar style is consistent**

❏ **17) No copy-pasted code is present**
Blocks of code that have the same functionality are factored out into modules