Student: ________________________________ AG score: ________

**General code quality**

- (2) Appropriate commenting.
- Function prototypes first, functions in readable order
- Main function is very small
- Program has well-chosen subfunctions to organize the code
- Program lacks redundant or awkward code.
- Code is not duplicated excessively.
- Code reads well, and is not excessively verbose or convoluted.
- Good variable/symbol naming
- Code follows recommended C style practices (e.g. NULL, typedefs, macro names)
- C used idiomatically (e.g. for statements follow idiomatic pattern)

**Specific code quality - differs depending on project**

- Main function built around a switch statement that calls command-specific functions
- Format for scanf of strings disallows overrun of array
- No unnecessary memory allocations (e.g. for temporary buffers - local arrays used instead)
- Memory allocated and deallocated in responsible module's functions only
- Return value from malloc checked and program terminated if 0
- Memory for strings allocated to fit the data
- p1_globals files are correct and used correctly
- Globals are incremented/decremented by responsible module
- Good choice of functions in Utility (e.g. a string allocator/deallocator)
- Standard Library facilities used appropriately (no reinvention of wheels)