Logical Layout:

1. All OpCapability instructions.  
2. Optional OpExtension instructions (extensions to SPIR-V).  
3. Optional OpExtInstImport instructions.  
4. The single required OpMemoryModel instruction.  
5. All entry point declarations, using OpEntryPoint.  
6. All execution-mode declarations, using OpExecutionMode or OpExecutionModeId.  
7. These debug instructions, which must be grouped in the following order:  
a. all OpString, OpSourceExtension, OpSource, and OpSourceContinued, without forward references.  
b. all OpName and all OpMemberName  
c. all OpModuleProcessed instructions  
8. All annotation instructions:  
a. all decoration instructions (OpDecorate, OpMemberDecorate, OpGroupDecorate, OpGroupMemberDecorate,  
and OpDecorationGroup).  
9. All type declarations (OpTypeXXX instructions), all constant instructions, and all global variable declarations (all  
OpVariable instructions whose Storage Class is not Function). This is the preferred location for OpUndef  
instructions, though they can also appear in function bodies. All operands in all these instructions must be declared  
before being used. Otherwise, they can be in any order. This section is the first section to allow use of OpLine debug  
information.  
10. All function declarations ("declarations" are functions without a body; there is no forward declaration to a function  
with a body). A function declaration is as follows.  
a. Function declaration, using OpFunction.  
b. Function parameter declarations, using OpFunctionParameter.  
c. Function end, using OpFunctionEnd.  
11. All function definitions (functions with a body). A function definition is as follows.  
a. Function definition, using OpFunction.  
b. Function parameter declarations, using OpFunctionParameter.  
c. Block  
d. Block  
e. . . .  
f. Function end, using OpFunctionEnd