Problem 8.4_____

4. What are the pros and cons of using unique closing reserved words on compound statements?

PRO: **Requires that the control structures be more apparent, which aids readability for more complex structures.**

CON: **Adds clutter to the code that, in many cases, adds little to the readability, particularly for simple structures.**

Adds to the number of keywords that have to be remembered and avoided when choosing identifier names.

Decreases the writability of the code by requiring more verbiage be included.

Problem 8.5_____

5. What are the arguments, pro and con, for Python's use of indentation to specify compound statements in control statements?

PRO: **Reduces clutter by eliminating block delimiters.**

Forces programmers to make their source code visually match its logical structure.

CON: **Creates maintenance headaches if two pieces of code use different indentation schemes.**

Makes it difficult to cut and paste code since indentation styles might differ.

Complicates compiler design since now indentation information must be tracked.

Problem 8.10

10. In Ada, the choice lists of the **`case`** statement must be exhaustive, so that there can be no unrepresented values in the control expression. In C++, unrepresented values can be caught at run time with the **`default`** selector. If there is no **`default`**, an unrepresented value causes the whole statement to be skipped. What are the pros and cons of these two designs (Ada and C++)?

PRO: **Ada makes the code more reliable since every value must be dealt with.**

C++ makes code more flexible and writable since only the code that is needed need be written.

CON: **Ada's choice is akin to requiring every if() statement to explicitly have an else() clause. But most program logic doesn't match this requirement and so the fit between the program logic and the program code is not as clean.**

Enumerating every possible value of the control statement is not practically possible unless enumerated data types are used (which is the normal case in Ada). By not requiring enumerations, C++ allows code to be more flexible by letting integer types be used for control expressions.