

### Questions

Here are questions about *libscheme: Scheme as a C library* by Brent W. Benson Jr.

1. Chapter 2.2 presents a function `make-counter`. How can you implement such a function in Tcl, or C?
2. What is the problem Benson tries to solve? What is the contribution of this paper?
3. For a conference, the members of the Program Committee decide about acceptance and rejection of papers. They can also demand changes for the final version that is published in the proceedings. As a member of the Program Committee, what changes (technical and presentational) would you have asked for?
4. Compare the extension mechanism for Tcl with that of libscheme. Draw a connection between the languages and the mechanisms to extend it; in other words, explain why the mechanisms to a large degree are a consequence of the language Scheme.
5. Libscheme uses the Boehm/Weiser conservative garbage collector for C and Benson recommends to use it for the application as well. Why? Think about how an application-allocated block of memory would be managed if only the Scheme interpreter used the garbage collector.
6. Ousterhout had to strike a balance between a powerful language and an implementation that is easy to extend. Benson had to strike similar balances; try to identify them.