We normally publish the post-mortem for an assignment after it has been marked and released. Here is a list of common errors provided by the graders for assignment 7.

Parts of Q01, Q02 and Q03 were chosen to be marked for different style criteria. Thus, it is possible that other questions might have style problems that we did not address. There were also design recipe elements that we chose to give feedback on and not deduct for. You should still improve on these, since we may decide to deduct for them on a later assignment. Please review the posted solutions and style guide to help resolve any questions you may have. If that is insufficient, please raise your questions in 1-1 consulting hours.

**Question 1 (bst.rkt)**

**Purpose**
- Many students did not explicitly reference the function parameters by name in the purpose. As a reminder, this means that the parameters that appear in your function header should be referenced by exact name in the purpose. For example: "...takes in a [bst] and ..."

**Question 2 (eval-apply.rkt)**

**Data Definition**
- Many students did not modify the original data definition to include constants.
- Many students did not use the correct type `sym` for modifying the data definition.

**Question 3 (trie.rkt)**

**Template**
- Many students were missing contracts for their templates.
- Many students incorrectly used a recursive list template for `trie-template`. Recall that the children of `trie` are a `(listof TNode)` rather than `(listof Trie)`.
- Many students did not include necessary references between template functions. For instance, `tnode-template` should make a call to `list-tnode-template`.
- Many students had one or more other significant errors in their templates. As a reminder, a template is a skeleton for later implementation. Since template design is data-definition-specific, your template should be similar to your real functions in both structure and usage.