After local assignment: test spam
After nonlocal assignment: nonlocal spam
After global assignment: nonlocal spam
In global scope: global spam

EXPLANATION:
During run time, first the compiler sets up the scope_test() function, then runs the call to scope_test(). In this call to scope_test(), it sets up the do_local(), do_nonlocal(), and do_global() functions, and creates a spam="test spam" variable. When it executes do_local(), a new local spam="local spam" is created that does not modify the old one. Thus, the first printing of spam in the scope_test() scope is still "test spam." Next, when it executes do_nonlocal(), the line nonlocal spam creates a spam variable pointing to the spam from scope_test(), which is "test spam". It then changes it to "nonlocal spam." So, the second printing of spam in the scope_test() scope results in "nonlocal spam." After this, when it executes do_global(), since there is no global spam, a new global spam is created and set to "global spam." Thus, since this did not modify the existing spam in scope_test(), the third printing of it is the same as the second printing of "nonlocal spam." Finally, the scope_test() function is exited. When the final print spam is called, it prints the spam that was created at the global level outside the scope of all the functions, "global spam."